

PRAWNOAUTORSKA OCHRONA PROGRAMÓW KOMPUTEROWYCH

Zbigniew Okoń

ZAGADNIENIA PRAWNE

Polecamy w serii:

Kulinaria w polskim prawie własności intelektualnej

Piotr Ślęzak

Prawna ochrona treści cyfrowych

Katarzyna Chałubińska-Jentkiewicz

Media w erze cyfrowej. Wyzwania i zagrożenia

redakcja naukowa

Katarzyna Chałubińska-Jentkiewicz, Monika Nowikowska, Krzysztof Wąsowski

Wpływ decyzji Urzędu Patentowego RP na postępowanie cywilne w sprawach
z zakresu prawa własności przemysłowej

Arkadiusz Sadza

Ochrona wzorów użytkowych

Marcin Balicki

Tłumaczenie w prawie autorskim

Jakub Dołęcki

Test trójstopniowy w prawie autorskim. Paradygmaty prawa autorskiego

Piotr Niezgódka

Domeny internetowe. Teoria i praktyka

redakcja naukowa *Ireneusz Matusiak*

Ustalenie kwoty stosownego wynagrodzenia za naruszenie autorskich praw majątkowych

Łukasz Maryniak

Naukowe i krytyczne wydania tekstów jako przedmiot prawa autorskiego i praw pokrewnych

Bohdan Widła

Ograniczenia ochrony patentowej a naruszenie patentu

Justyna Ożegalska-Trybalska

Pośrednicy internetowi w prawie Unii Europejskiej.

Rola i obowiązki wobec treści użytkowników

Magdalena Piech

Autorskie prawa osobiste w obrocie prawnym

Michał Wyrwiński

Swoboda wypowiedzi artystycznej. Standardy międzynarodowe i krajowe

Marcin Górski

Naruszenie prawa do znaku towarowego renomowanego. Studium prawnoporównawcze

Joanna Sitko

Zarząd wspólnym prawem autorskim

Michał Markiewicz

Korzystanie z chronionego wynalazku w celu rejestracji produktu leczniczego

Agnieszka Sztoldman

PRAWNOAUTORSKA OCHRONA PROGRAMÓW KOMPUTEROWYCH

Zbigniew Okoń

ZAGADNIENIA PRAWNE

Stan prawny na 1 maja 2022 r.

Redakcja serii

Ryszard Markiewicz, Andrzej Matlak, Alicja Pollesch

Recenzent

Dr hab. Marek Świerczyński, prof. uczelni

Wydawca

Monika Pawłowska

Redaktor prowadzący

Joanna Tchorek

Opracowanie redakcyjne

Katarzyna Świerk-Bożek

Projekt okładek serii

Wojtek Janikowski, Przemek Dębowski

prawolubni

Ta książka jest wspólnym dziełem twórcy i wydawcy. Prosimy, byś przestrzegał przysługujących im praw. Książkę możesz udostępnić osobom bliskim lub osobiście znanym, ale nie publikuj jej w internecie. Jeśli cytujesz fragmenty, nie zmieniaj ich treści i koniecznie zaznacz, czyje to dzieło. A jeśli musisz skopiować część, rób to jedynie na użytek osobisty.

Szanujemy prawo i własność

Więcej na www.legalnakultura.pl

Polska Izba Książki

© Copyright by Wolters Kluwer Polska Sp. z o.o., 2022

ISBN 978-83-8286-157-0

Wolters Kluwer Polska Sp. z o.o.

Dział Praw Autorskich

01-208 Warszawa, ul. Przyokopowa 33

tel. 728 313 462

e-mail: PL-ksiazki@wolterskluwer.com

księgarnia internetowa www.profinfo.pl

SPIS TREŚCI

Wykaz skrótów	17
Wstęp	23

Część I

OCHRONA PROGRAMÓW KOMPUTEROWYCH W PRAWIE AUTORSKIM STANÓW ZJEDNOCZONYCH

Rozdział I

Geneza i rozwój ochrony programów komputerowych w prawie autorskim

Stanów Zjednoczonych	31
1. Początki prawnautorskiej ochrony programów komputerowych	31
2. Raport CONTU i <i>Computer Software Copyright Act</i>	33
3. <i>Computer Software Rental Amendments Act</i>	38
4. <i>Digital Millennium Copyright Act</i>	39
5. Podsumowanie	40

Rozdział II

Przedmiotowy zakres ochrony	43
1. Uwagi wprowadzające	43
2. Ochrona przed literalnym kopiowaniem	45
3. Ochrona przed nieliteralnym kopiowaniem	48
3.1. Struktura, sekwencja i organizacja programu	48
3.2. Test <i>abstraction-filtration-comparison</i>	51
4. Ochrona interfejsów użytkownika	56
4.1. Interfejs użytkownika jako pozatekstowy element programu komputerowego	56
4.2. Struktura, sekwencja i organizacja interfejsu użytkownika	58
4.3. Test <i>extrinsic-intrinsic</i>	60
4.4. Sprawa <i>Apple Computer Inc. v. Microsoft Corp.</i>	61
4.5. Test <i>abstraction-filtration-comparison</i> w sporach dotyczących interfejsów użytkownika	65

4.6. Interfejs użytkownika jako „metoda działania” lub „proces”	66
4.7. Krytyka w nauce prawa	68
5. Ochrona <i>Application Program Interface</i> programu komputerowego	70
6. Podsumowanie	77

Rozdział III

Prawa wyłączne	79
1. Katalog praw wyłącznych	79
2. Reprodukacja	80
3. Utwory pochodne (opracowania)	83
4. Dystrybucja kopii i doktryna <i>first sale</i>	87
5. Podsumowanie	92

Rozdział IV

Ograniczenia praw wyłącznych	95
1. Reprodukacja i adaptacja przez właściciela kopii programu	95
2. Reprodukacja dla potrzeb serwisu i naprawy sprzętu	100
3. <i>Fair use</i>	101
3.1. Ogólna charakterystyka <i>fair use</i>	101
3.2. <i>Reverse engineering</i>	104
3.3. Sprawa Oracle America, Inc. v. Google, Inc.	110
3.4. <i>Fair use</i> jako podstawa adaptacji programu	114
4. Kontraktowe ograniczenia <i>fair use</i>	115
4.1. Dopuszczalność kontraktowego ograniczenia <i>fair use</i>	115
4.2. Skuteczność licencji <i>shrink-wrap</i>	117
4.3. Kontraktowe zakazy <i>reverse engineering</i>	119
5. Podsumowanie	121

Rozdział V

Ochrona technicznych zabezpieczeń	123
1. Uwagi wprowadzające	123
2. Zakres ochrony technicznych zabezpieczeń w <i>Digital Millennium Copyright Act</i>	124
3. Wyjątki od zakazu obchodzenia zabezpieczeń	126
4. Ochrona technicznych zabezpieczeń a <i>fair use</i>	130
5. Podsumowanie	133

Część II

OCHRONA PROGRAMÓW KOMPUTEROWYCH W POROZUMIENIACH MIĘDZYNARODOWYCH Z ZAKRESU PRAWA AUTORSKIEGO

Rozdział VI

Prace prowadzone w ramach World Intellectual Property Organization

w latach 1970–1990	137
1. Przepisy wzorcowe o ochronie oprogramowania	137
2. Próby zapewnienia międzynarodowej ochrony oprogramowania	140
3. Podsumowanie	143

Rozdział VII

Porozumienie w sprawie handlowych aspektów praw własności

intelektualnej	145
1. Ewolucja regulacji odnoszącej się do programów komputerowych podczas negocjacji porozumienia w sprawie handlowych aspektów praw własności intelektualnej	145
2. Stosunek do konwencji berneńskiej, forma wyrażenia, elementy niechronione	146
3. Uprawnienia majątkowe, najem programów komputerowych	148
4. Podsumowanie	150

Rozdział VIII

Traktat World Intellectual Property Organization o prawie autorskim	151
1. Programy komputerowe podczas negocjacji Traktatu o prawie autorskim	151
2. Postanowienia Traktatu o prawie autorskim odnoszące się do programów komputerowych	154
3. Prawa minimalne	155
3.1. Prawo reprodukcji	155
3.2. Prawo wprowadzania do obrotu	159
3.3. Prawo komunikowania utworu publiczności	161
4. Ochrona technicznych zabezpieczeń	163
5. Podsumowanie	166

Część III

OCHRONA PROGRAMÓW KOMPUTEROWYCH W PRAWIE UNII EUROPEJSKIEJ

Rozdział IX

Harmonizacja prawnoautorskiej ochrony programów komputerowych

w Unii Europejskiej	169
1. Początki harmonizacji prawa autorskiego w Unii Europejskiej	169

2. Dyrektywa w sprawie ochrony prawnej programów komputerowych	171
2.1. Zielona Księga o prawie autorskim i wyzwaniach technologii.....	171
2.2. Projekt Komisji z 1989 r.....	175
2.3. Reakcje na projekt Komisji z 1989 r.	176
2.4. Pierwsze czytanie w Parlamencie Europejskim.....	179
2.5. Zmieniony projekt Komisji z 1990 r. i przyjęcie dyrektywy	180
2.6. Dyrektywa 91/250/EWG	183
2.7. Zmiany dyrektywy 91/250/EWG, wersja skodyfikowana	184
3. Późniejsza harmonizacja prawa autorskiego w Unii Europejskiej.....	185
3.1. Harmonizacja w latach 1991–2001.....	185
3.2. Harmonizacja w latach 2002–2012	186
3.3. Harmonizacja po roku 2012	187
4. Podsumowanie.....	189

Rozdział X

Przedmiot ochrony	191
1. Specyfika programu jako przedmiotu ochrony prawnoautorskiej	191
1.1. Pojęcie programu komputerowego	191
1.2. Ochrona „jak dzieł literackich w rozumieniu konwencji berneńskiej”	193
1.3. Kod źródłowy i kod obiektowy	194
1.4. Przygotowawczy materiał projektowy.....	197
2. Przesłanki ochrony.....	199
2.1. Harmonizacja przesłanek ochrony w dyrektywie 2009/24/WE	199
2.2. „Horyzontalna” harmonizacja pojęcia utworu w orzecznictwie TSUE	202
2.3. Oryginalność w orzecznictwie TSUE.....	205
2.4. Znaczenie orzecznictwa TSUE dla interpretacji przesłanki oryginalności programów komputerowych	208
2.5. Zdarność wyrażenia programu do powielenia.....	209
3. Przedmiotowe granice ochrony	212
3.1. Uwagi wprowadzające	212
3.2. Interfejsy użytkownika.....	214
3.2.1. Orzecznictwo sądów państw członkowskich	214
3.2.2. Sprawa Bezpečnostní softwarová asociace.....	218
3.2.3. Zakres stosowania dyrektywy 2001/29/WE	222
3.3. Funkcje techniczne i zbiór funkcji programu	224
3.4. Języki programowania	228
3.5. Formaty plików.....	233
3.6. Kompilacje elementów niechronionych	237
4. Podsumowanie.....	240

Rozdział XI

Podmiot uprawniony	243
1. Zasada twórcy	243
2. Wyjątki od zasady twórcy	244
2.1. Osoby prawne „uznawane za autora”	244
2.2. Utwory kolektywne	245
3. Programy powstające w ramach stosunku pracy	246
4. Programy generowane komputerowo	249
5. Ochrona programów komputerowych autorów obcych	250
6. Podsumowanie	251

Rozdział XII

Czynności zastrzeżone	253
1. Katalog praw i metoda harmonizacji	253
2. Zwielokrotnianie	256
2.1. Ewolucja koncepcji zwielokrotniania	256
2.2. Zwielokrotnianie (powielanie) w dyrektywie 2009/24/WE	260
2.3. Czynności niestanowiące zwielokrotnienia	265
3. Wprowadzanie zmian (tłumaczenie, adaptacja, aranżacja i inne przeróbki)	267
3.1. Uwagi wprowadzające	267
3.2. Czynności objęte art. 4 ust. 1 lit. b dyrektywy 2009/24/WE	268
3.3. Zezwolenie uprawnionego	271
3.4. Prawa osoby dokonującej opracowania	273
4. Publiczna dystrybucja	274
4.1. Uwagi wprowadzające	274
4.2. Dystrybucja materialnych kopii, „publiczność” dystrybucji	276
4.3. Dystrybucja elektroniczna	279
5. Najem	282
5.1. Najem w dyrektywie 2009/24/WE	282
5.2. Stosowanie dyrektywy 2006/115/WE	283
5.3. Najem urzędzeń zawierających programy komputerowe	285
6. Użyczenie	286
7. Podsumowanie	287

Rozdział XIII

Wyczerpanie prawa	289
1. Specyfika wyczerpania prawa w obrocie programami komputerowymi	289
2. Ogólna charakterystyka wyczerpania prawa w dyrektywie 2009/24/WE	292
3. Orzecznictwo państw członkowskich Unii Europejskiej poprzedzające wyrok TSUE w sprawie UsedSoft	296
4. Wyrok TSUE w sprawie UsedSoft	299

4.1. Geneza sporu i postępowanie przed sądami krajowymi.....	299
4.2. „Sprzedaż” jako autonomiczne pojęcie prawa Unii Europejskiej	302
4.3. Ekwiwalentność dystrybucji „materialnych” i „niematerialnych” kopii.....	305
4.4. Stosowanie dyrektywy 2001/29/WE i Traktatu o prawie autorskim....	308
4.5. Brak przekazania kopii.....	310
4.6. Aktualizacje i umowa serwisowa.....	313
4.7. Rozdzielanie pakietów użytkowników	314
4.8. Ciężar dowodu	315
5. Orzecznictwo TSUE i państw członkowskich po wyroku w sprawie UsedSoft	316
5.1. Wyrok BGH w sprawie UsedSoft III.....	316
5.2. Wyrok BGH w sprawie Green-IT.....	320
5.3. Wyrok TSUE w sprawie Ranks i Vasiļevičs.....	322
5.4. Wyrok TSUE w sprawie Tom Kabinet	324
6. Skuteczność kontraktowych ograniczeń i zakazów odsprzedaży.....	328
7. Wyczerpanie w dystrybucji gier komputerowych	330
8. Podsumowanie.....	333

Rozdział XIV

Wyjątki od czynności zastrzeżonych.....	335
1. Katalog wyjątków	335
2. Korzystanie przez uprawnionego nabywcę zgodnie z zamierzonym celem.....	337
2.1. Uwagi wprowadzające	337
2.2. Pojęcie „uprawniony nabywca”	339
2.2.1. Funkcje pojęcia „uprawniony nabywca”	339
2.2.2. Uprawniony nabywca jako licencjobiorca.....	340
2.2.3. Uprawniony nabywca jako beneficjent wyjątku lub ograniczenia	341
2.2.4. Uprawniony nabywca jako osoba, która legalnie uzyskała kopię programu.....	342
2.2.5. Status członków rodziny, pracowników i zleceniobiorców uprawnionego nabywcy	344
2.3. Dozwolone czynności.....	345
2.4. Konstrukcja prawna wyjątku na rzecz uprawnionego nabywcy.....	349
2.4.1. Istota problemu	349
2.4.2. Koncepcja dorozumianej licencji	350
2.4.3. Koncepcja rozszerzenia do wyczerpania.....	352
2.4.4. Koncepcja „prawa” użytkownika	354
2.4.5. Koncepcja ograniczenia praw wyłącznych.....	357
2.4.6. „Obowiązkowy rdzeń” art. 5 ust. 1 dyrektywy 2009/24/WE.....	360

3. Sporządzenie kopii zapasowej	367
4. Obserwowanie, badanie i testowanie (analiza)	368
5. Dekompilacja.....	371
5.1. Charakterystyka ogólna	371
5.2. Test trójstopniowy	375
6. Podsumowanie.....	378

Rozdział XV

Techniczne zabezpieczenia programów	381
1. Ochrona technicznych zabezpieczeń w dyrektywie 2009/24/WE	381
1.1. Uwagi wprowadzające	381
1.2. Chronione „urządzenia techniczne”	383
1.3. Środki przeznaczone do usuwania lub omijania zabezpieczeń	384
1.4. Zakazane czynności.....	386
1.5. Właściwe środki ochrony prawnej.....	387
1.6. Usuwanie lub omijanie zabezpieczeń w celu użycia programu zgodnie z zamierzonym celem	388
2. Stosowanie dyrektywy 2001/29/WE.....	390
3. Podsumowanie.....	393

Rozdział XVI

Porównanie modelu ochrony programów komputerowych w Stanach

Zjednoczonych i Unii Europejskiej	395
1. Przedmiotowe granice ochrony	395
2. Prawa wyłączne.....	400
3. Ograniczenia praw wyłącznych.....	403
4. Techniczne zabezpieczenia.....	405
5. Podsumowanie.....	408

Część IV

OCHRONA PROGRAMÓW KOMPUTEROWYCH W POLSKIM PRAWIE AUTORSKIM

Rozdział XVII

Objęcie programów komputerowych ochroną prawa autorskiego	413
1. Status programów komputerowych w ustawie o prawie autorskim z 1952 r.	413
2. Przebieg prac nad ustawą o prawie autorskim i prawach pokrewnych z 1994 r.	416
3. Programy komputerowe w obowiązującej ustawie o prawie autorskim i prawach pokrewnych	419
4. „Abolicja” komputerowa w latach 1994–2000.....	421

5. Zmiany przepisów szczególnych dotyczących programów komputerowych	422
6. Podsumowanie.....	423

Rozdział XVIII

Przedmiot ochrony prawnoautorskiej	425
1. Zakres zastosowania przepisów rozdziału 7 pr. aut.	425
1.1. Definicja programu komputerowego.....	425
1.2. Forma wyrażenia programu komputerowego, kod źródłowy i obiektowy	426
1.3. Gry komputerowe i multimedia	430
2. Przesłanki ochrony.....	435
2.1. Twórczość i indywidualność.....	435
2.1.1. Uwagi wprowadzające.....	435
2.1.2. Poglądy doktryny.....	437
2.1.3. Wykładnia prounijna	440
2.2. Ustalenie	446
2.3. Okoliczności irrelewantne.....	448
3. Przedmiotowe granice ochrony	450
3.1. Uwagi wprowadzające	450
3.2. Brak ochrony idei i zasad	451
3.3. Znaczenie pojęcia „formy wyrażenia programu”	455
3.4. Ochrona struktury wewnętrznej i algorytmów programu	455
4. Podsumowanie.....	457

Rozdział XIX

Podmiot prawa autorskiego.....	459
1. Twórca jako podmiot prawa autorskiego	459
2. Programy współautorskie.....	460
3. Twórczość pracownicza	466
3.1. Pierwotny charakter nabycia autorskich praw majątkowych przez pracodawcę	466
3.2. Pojęcie „stosunek pracy”	468
3.3. Powstanie programu w wyniku wykonywania obowiązków pracowniczych	469
3.4. Zakres praw nabywanych przez pracodawcę	471
4. Utwory zbiorowe.....	472
5. Podsumowanie.....	474

Rozdział XX

Autorskie prawa osobiste	475
1. Uwagi wprowadzające	475

2. Zakres autorskich praw osobistych programisty w ustawie o prawie autorskim i prawach pokrewnych.....	477
3. Ograniczenie katalogu autorskich praw osobistych a konwencja berneńska.....	478
4. Prawo do autorstwa utworu oraz decydowania o jego oznaczeniu.....	482
4.1. Treść i podmioty uprawnione.....	482
4.2. Skutki nieujawnienia autorstwa programu	483
5. Publiczne udostępnienie utworu w nieodpowiedniej formie albo ze zmianami	484
6. Podsumowanie.....	485

Rozdział XXI

Autorskie prawa majątkowe	487
1. Konstrukcja autorskich praw majątkowych do programu komputerowego.....	487
2. Zwiłokrotnianie	491
2.1. Ewolucja pojęcia zwiłokrotniania w polskim prawie autorskim	491
2.2. Zwiłokrotnianie programu komputerowego	494
2.3. Zwiłokrotnianie w związku z transmisją programu w sieci.....	498
3. Modyfikacja i opracowanie	500
3.1. Uprawnienia przysługujące twórcy pierwotnego programu komputerowego	500
3.2. Wprowadzanie zmian do programu.....	502
3.3. Opracowanie programu komputerowego	506
3.4. Prawo zezwalania na wykonywanie praw zależnych	507
4. Rozpowszechnianie.....	510
4.1. Rozpowszechnianie w ogólności.....	510
4.2. Wprowadzanie do obrotu.....	514
4.3. Najem	518
4.4. Użyczenie.....	520
5. Podsumowanie.....	521

Rozdział XXII

Wyczerpanie prawa	523
1. Przesłanki wyczerpania prawa w obrocie programami komputerowymi	523
1.1. Przesłanki wyczerpania prawa w ustawie o prawie autorskim i prawach pokrewnych	523
1.2. Ekonomiczno-funkcjonalne znaczenie pojęcia „przeniesienie własności”	526
1.3. Licencja na oprogramowanie jako „przeniesienie własności”	529
1.4. Wprowadzenie programu do obrotu w dystrybucji „fizycznych” egzemplarzy.....	533

1.5. Wprowadzenie programu do obrotu w dystrybucji elektronicznej	536
1.6. Jednolitość czy wielość modeli wyczerpania prawa?	539
2. Odsprzedaż kopii programu komputerowego.....	543
2.1. Prawa ulegające wyczerpaniu	543
2.2. Przekazanie kopii programu.....	544
2.3. Sytuacja prawna dalszego nabywcy kopii.....	546
2.4. Sytuacja prawna zbywcy „wyczerpanej” kopii.....	550
2.5. Obowiązki dowodowe.....	552
3. Bezwzględnie obowiązujący charakter wyczerpania.....	553
4. Podsumowanie.....	554

Rozdział XXIII

Ograniczenia autorskich praw majątkowych	557
1. Specyfika ograniczeń autorskich praw majątkowych dotyczących programów komputerowych.....	557
2. Korzystanie zgodnie z przeznaczeniem	559
2.1. Uwagi wprowadzające	559
2.2. Pojęcie „legalny posiadacz”	561
2.3. Konstrukcja prawna zezwolenia na korzystanie zgodnie z przeznaczeniem	564
2.3.1. Poglądy doktryny.....	564
2.3.2. Artykuł 75 ust. 1 pr. aut. jako ograniczenie autorskich praw majątkowych.....	567
2.3.3. Artykuł 75 ust. 1 pr. aut. jako reguła interpretacyjna lub norma dyspozytywna	570
2.3.4. Zabezpieczenie minimum swobód legalnego posiadacza	575
2.3.5. Konkluzje.....	579
2.4. Dozwolone czynności.....	579
2.4.1. „Przeznaczenie” programu.....	579
2.4.2. „Niezbędność” czynności.....	583
2.4.3. Dekompilacja w celu użycia programu zgodnie z przeznaczeniem.....	585
2.4.4. Opracowania dokonane przez legalnego posiadacza	588
3. Pozostałe ograniczenia przewidziane w przepisach szczególnych dotyczących programów komputerowych.....	589
3.1. Sporządzanie kopii zapasowych.....	589
3.1.1. Różnice w stosunku do dyrektywy 2009/24/WE.....	589
3.1.2. Przesłanka niezbędności sporządzenia kopii zapasowej	590
3.1.3. Liczba wykonywanych kopii zapasowych	593
3.2. Obserwowanie, badanie i testowanie programu.....	594
3.3. Dekompilacja	596
3.3.1. Uwagi wprowadzające.....	596

3.3.2. Przesłanki dopuszczalności dekompilacji.....	598
3.3.3. Wykorzystanie informacji uzyskanych z dekompilacji.....	602
3.3.4. Test trójstopniowy i stosowanie art. 35 pr. aut.	605
3.4. Nieważność odmiennych postanowień umownych.....	606
4. Stosowanie innych przepisów o dozwolonym użytku	608
5. Podsumowanie.....	609
Rozdział XXIV	
Ochrona technicznych zabezpieczeń	611
1. Uwagi wprowadzające	611
2. Techniczne zabezpieczenia i skuteczne techniczne zabezpieczenia.....	613
3. Roszczenie o zniszczenie niedozwolonych środków technicznych.....	615
4. Roszczenia przysługujące w razie usuwania lub obchodzenia zabezpieczeń.....	616
5. Ochrona przed podejmowaniem „czynności przygotowawczych”	618
6. Podsumowanie.....	621
Wnioski końcowe	623
1. Ogólna ocena prawnoautorskiego modelu ochrony programów komputerowych	623
2. Aktualny stan i perspektywy rozwoju ochrony programów w prawie Unii Europejskiej.....	625
3. Aktualny stan ochrony programów w prawie polskim i postulaty <i>de lege ferenda</i>	629
Tabela korelacji przepisów dyrektywy 91/250/EWG i dyrektywy 2009/24/WE	633
Literatura.....	635
Raporty i dokumenty	657
Orzecznictwo	659

WYKAZ SKRÓTÓW

Prawo krajowe

austriackie prawo autorskie	– Bundesgesetz über das Urheberrecht an Werken der Literatur und der Kunst und über verwandte Schutzrechte (Urheberrechtsgesetz)
BGB	– Bürgerliches Gesetzbuch – niemiecki kodeks cywilny
CDPA	– Copyright, Designs and Patents Act 1988
Copyright Act	– Copyright Law of the United States 1976 (Title 17 of the United States Code)
CPI	– Code de la Propriété Intellectuelle – francuski kodeks własności intelektualnej
czeskie prawo autorskie	– Zákon o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon)
DMCA	– Digital Millennium Copyright Act 1998
fińskie prawo autorskie k.c.	– Tekijänoikeuslaki – ustawa z 23.04.1964 r. – Kodeks cywilny (Dz.U. z 2020 r. poz. 1740 ze zm.)
Konstytucja RP	– Konstytucja Rzeczypospolitej Polskiej z 2.04.1997 r. (Dz.U. Nr 78, poz. 483 ze zm.)
k.p.	– ustawa z 26.06.1974 r. – Kodeks pracy (Dz.U. z 2020 r. poz. 1320 ze zm.)
LPI	– Ley de Propiedad Intelectual – hiszpańskie prawo własności intelektualnej
pr. aut.	– ustawa z 4.02.1994 r. o prawie autorskim i prawach pokrewnych (Dz.U. z 2021 r. poz. 1062 ze zm.)
pr. aut. z 1952 r.	– ustawa z 10.07.1952 r. o prawie autorskim (Dz.U. Nr 34, poz. 234 ze zm.)
UrhG	– Gesetz über Urheberrecht und verwandte Schutzrechte (Urheberrechtsgesetz) –niemiecka ustawa o prawie autorskim i prawach pokrewnych

Prawo Unii Europejskiej

dyrektywa 87/54/EWG	– dyrektywa Rady 87/54/EWG z 16.12.1986 r. w sprawie ochrony prawnej topografii produktów półprzewodnikowych (Dz.Urz. UE L 24, s. 36, ze zm.)
---------------------	---

-
- | | |
|---|---|
| dyrektywa 91/250/EWG | – dyrektywa Rady 91/250/EWG z 14.05.1991 r. w sprawie ochrony prawnej programów komputerowych (Dz.Urz. UE L 122, s. 42, ze zm.) |
| dyrektywa 92/100/EWG | – dyrektywa Rady 92/100/EWG z 19.11.1992 r. w sprawie prawa najmu i użyczenia oraz niektórych praw pokrewnych prawu autorskiemu w zakresie własności intelektualnej (Dz.Urz. UE L 346, s. 61, ze zm.) |
| dyrektywa 93/98/EWG | – dyrektywa Rady 93/98/EWG z 29.10.1993 r. w sprawie harmonizacji czasu ochrony prawa autorskiego i niektórych praw pokrewnych (Dz.Urz. UE L 290, s. 9, ze zm.) |
| dyrektywa 96/9/WE
(dyrektywa w sprawie
ochrony prawnej baz
danych) | – dyrektywa 96/9/WE Parlamentu Europejskiego i Rady z 11.03.1996 r. w sprawie ochrony prawnej baz danych (Dz.Urz. UE L 77, s. 20, ze zm.) |
| dyrektywa 2001/29/WE | – dyrektywa 2001/29/WE Parlamentu Europejskiego i Rady z 22.05.2001 r. w sprawie harmonizacji niektórych aspektów praw autorskich i pokrewnych w społeczeństwie informacyjnym (Dz.Urz. UE L 167, s. 10, ze zm.) |
| dyrektywa 2004/48/WE | – dyrektywa 2004/48/WE Parlamentu Europejskiego i Rady z 29.04.2004 r. w sprawie egzekwowania praw własności intelektualnej (Dz.Urz. UE L 157, s. 45) |
| dyrektywa 2006/115/WE | – dyrektywa 2006/115/WE Parlamentu Europejskiego i Rady z 12.12.2006 r. w sprawie prawa najmu i użyczenia oraz niektórych praw pokrewnych prawu autorskiemu w zakresie własności intelektualnej (Dz.Urz. UE L 376, s. 28) |
| dyrektywa 2006/116/WE | – dyrektywa 2006/116/WE Parlamentu Europejskiego i Rady z 12.12.2006 r. w sprawie czasu ochrony prawa autorskiego i niektórych praw pokrewnych (Dz.Urz. UE L 372, s. 12, ze zm.) |
| dyrektywa 2009/24/WE | – dyrektywa Parlamentu Europejskiego i Rady 2009/24/WE z 23.04.2009 r. w sprawie ochrony prawnej programów komputerowych (Dz.Urz. UE L 111, s. 16) |
| dyrektywa 2017/1564 | – dyrektywa Parlamentu Europejskiego i Rady (UE) 2017/1564 z 13.09.2017 r. w sprawie niektórych dozwolonych sposobów korzystania z określonych utworów i innych przedmiotów chronionych prawem autorskim i prawami pokrewnymi z korzyścią dla osób niewidomych, osób słabowidzących lub osób z niepełnosprawnościami uniemożliwiającymi zapoznawanie się z drukiem oraz w sprawie zmiany dyrektywy 2001/29/WE w sprawie harmonizacji niektórych aspektów praw autorskich i pokrewnych w społeczeństwie informacyjnym (Dz.Urz. UE L 242, s. 6) |
| dyrektywa 2019/790
(dyrektywa o prawie
autorskim na jednolitym
rynku cyfrowym) | – dyrektywa Parlamentu Europejskiego i Rady (UE) 2019/790 z 17.04.2019 r. w sprawie prawa autorskiego i praw pokrewnych na jednolitym rynku cyfrowym oraz zmiany dyrektyw 96/9/WE i 2001/29/WE (Dz.Urz. UE L 130, s. 92) |
| dyrektywa 2012/28/UE | – dyrektywa Parlamentu Europejskiego i Rady 2012/28/UE z 25.10.2012 r. w sprawie niektórych dozwolonych sposobów korzystania z utworów osieroconych (Dz.Urz. UE L 299, s. 5) |

rozporządzenie 6/2002	–	rozporządzenie Rady (WE) nr 6/2002 z 12.12.2001 r. w sprawie wzorów wspólnotowych (Dz.Urz. UE L 3 z 2002 r., s. 1, ze zm.)
TEWG	–	Traktat ustanawiający Europejską Wspólnotę Gospodarczą
TFUE	–	Traktat o funkcjonowaniu Unii Europejskiej (Dz.Urz. UE C 202 z 2016 r., s. 47)
TWE	–	Traktat ustanawiający Wspólnotę Europejską

Porozumienia międzynarodowe

GATT	–	General Agreement on Tariffs and Trade – układ ogólny w sprawie taryf celnych i handlu, podpisany w Genewie 30.10.1947 r.
konwencja berneńska	–	Akt paryski Konwencji berneńskiej o ochronie dzieł literackich i artystycznych sporządzony w Paryżu 24.07.1971 r. (Dz.U. z 1990 r. Nr 82, poz. 474)
konwencja wiedeńska	–	Konwencja wiedeńska o prawie traktatów sporządzona w Wiedniu 23.05.1969 r. (Dz.U. z 1990 r. Nr 74, poz. 439)
TAWF	–	Traktat WIPO sporządzony w Genewie 20.12.1996 r. o artystycznych wykonaniach i fonogramach (Dz.U. z 2004 r. Nr 41, poz. 375)
TPA	–	Traktat Światowej Organizacji Własności Intelektualnej o prawie autorskim, sporządzony w Genewie 20.12.1996 r. (Dz.U. z 2005 r. Nr 3, poz. 12)
TRIPS	–	Porozumienie w sprawie handlowych aspektów praw własności intelektualnej (Dz.U. z 1996 r. Nr 32, poz. 143)

Czasopisma i publikatory

BGBL	–	Bundesgesetzblatt
Brook.	–	Brooklyn Law Review
BTLJ	–	Berkeley Technology Law Journal
CLSR	–	Computer Law & Security Review
Colum. J.L. & Arts	–	Columbia Journal of Law & the Arts
Colum. L. Rev.	–	Columbia Law Review
CR	–	Computer und Recht
CRI	–	Computer Law Review International
CTLR	–	Computer and Telecommunications Law Review
Duke LJ	–	Duke Law Journal
ECDR	–	European Copyright and Design Reports
EIPR	–	European Intellectual Property Review
EPS	–	Europejski Przegląd Sądowy
F.2d	–	Federal Reporter, Second Series
F.3d	–	Federal Reporter, Third Series
F.S.R.	–	Fleet Street Reports
GRUR	–	Gewerblicher Rechtsschutz und Urheberrecht
GRUR Int.	–	Gewerblicher Rechtsschutz und Urheberrecht Internationaler Teil

GRUR-Prax	– Gewerblicher Rechtsschutz und Urheberrecht, Praxis im Immaterialgüter und Wettbewerbsrecht
GRUR-RR	– Gewerblicher Rechtsschutz und Urheberrecht Rechtsprechungs-Report
GWLR	– George Washington Law Review
Harv. JL & Tech.	– Harvard Journal of Law & Technology
HC&ELJ	– Hastings Communications and Entertainment Law Journal
HTLJ	– Santa Clara Computer and High-Technology Law Journal
IIC	– International Review of Intellectual Property and Competition Law
J. Copyright Soc’y USA	– Journal of the Copyright Society of the USA
J. Intell. Prop. L.	– Journal of Intellectual Property Law
JIPITEC	– Journal of Intellectual Property, Information Technology and Electronic Commerce Law
KPP	– Kwartalnik Prawa Prywatnego
L&CP	– Law and Contemporary Problems
Mich. L. Rev.	– Michigan Law Review
MMR	– Multimedia und Recht
M. Praw.	– Monitor Prawniczy
NJW	– Neue Juristische Wochenschrift
NJW-RR	– Neue Juristische Wochenschrift Rechtsprechungs-Report Zivilrecht
OSN	– Orzecznictwo Sądu Najwyższego
OSNC	– Orzecznictwo Sądu Najwyższego. Izba Cywilna
OSNCP	– Orzecznictwo Sądu Najwyższego. Izba Cywilna, Pracy i Ubezpieczeń Społecznych
OSNKW	– Orzecznictwo Sądu Najwyższego. Izba Karna i Wojskowa
OSNP	– Orzecznictwo Sądu Najwyższego. Izba Pracy, Ubezpieczeń Społecznych i Spraw Publicznych
OSP	– Orzecznictwo Sądów Polskich
OSPika	– Orzecznictwo Sądów Polskich i Komisji Arbitrażowych
PiP	– Państwo i Prawo
PPH	– Przegląd Prawa Handlowego
PUG	– Przegląd Ustawodawstwa Gospodarczego
RC&TLJ	– Rutgers Computer and Technology Law Journal
RIDA	– Revue Internationale du Droit d’Auteur
RPC	– Reports of Patent, Design and Trade Mark Cases
RPEiS	– Ruch Prawniczy, Ekonomiczny i Socjologiczny
Stan. L. Rev.	– Stanford Law Review
U. Balt. Intell. Prop. L.J.	– University of Baltimore Intellectual Property Law Journal
U. Cin. L. Rev.	– University of Cincinnati Law Review
U.S.P.Q.	– United States Patents Quarterly
U.S.P.Q. 2d (BNA)	– United States Patents Quarterly, Second Series
ZNUJ	– Zeszyty Naukowe Uniwersytetu Jagiellońskiego
ZNUJ. PIPWI	– Zeszyty Naukowe Uniwersytetu Jagiellońskiego. Prace Instytutu Prawa Własności Intelektualnej
ZNUJ. PPWI	– Zeszyty Naukowe Uniwersytetu Jagiellońskiego. Prace z Prawa Własności Intelektualnej
ZNUJ. PWiOWI	– Zeszyty Naukowe Uniwersytetu Jagiellońskiego. Prace z Wynalazczości i Ochrony Własności Intelektualnej

ZUM	- Zeitschrift für Urheber- und Medienrecht
ZUM-RD	- Zeitschrift für Urheber- und Medienrecht – Rechtsprechungs- dienst

Instytucje i sądy

BGH	- Bundesgerichtshof
CAFC	- United States Court of Appeals for the Federal Circuit
CCPA	- United States Court of Customs and Patent Appeals
CONTU	- Commission on New Technological Uses of Copyrighted Works
EFTA	- European Free Trade Association
EOG	- Europejski Obszar Gospodarczy
EWCA	- England and Wales Court of Appeal
EWG	- Europejska Wspólnota Gospodarcza
EWHC	- England and Wales High Court.
LG	- Landgericht
NSA	- Naczelny Sąd Administracyjny
OGH	- Oberste Gerichtshof
OLG	- Oberlandesgericht
SA	- Sąd Apelacyjny
SN	- Sąd Najwyższy
SO	- Sąd Okręgowy
TSUE/TS	- Trybunał Sprawiedliwości Unii Europejskiej (do 1.12.2009 r. jako Trybunał Sprawiedliwości)
USPTO	- United States Patent and Trademark Office
WIPO	- World Intellectual Property Organization
WSA	- Wojewódzki Sąd Administracyjny
WTO	- World Trade Organization

Inne

API	- Application Programming Interface
CAPTCHA	- Completely Automated Public Turing test to tell Computers and Humans Apart
EEPROM	- Electrically Erasable Programmable Read-Only Memory
EPROM	- Erasable Programmable Read-Only Memory
EULA	- End-User License Agreement
FTP	- File Transfer Protocol
OEM	- Original Equipment Manufacturer
PROM	- Programmable Read-Only Memory
RAM	- Random Access Memory
ROM	- Read-Only Memory

WSTĘP

Programy komputerowe są obecne w ustawach autorskich od 50 lat¹. W tym czasie z innowacji o marginalnym znaczeniu dla prawa autorskiego stały się głównym produktem przemysłu kreatywnego, którego gospodarcza wartość przewyższa film, reklamę, prasę i literaturę czy przemysł muzyczny i nagraniowy razem wzięte². Zarazem, choć są chronione jako utwory literackie i dla prawa autorskiego stanowią utwór wyrażony słowami, symbolami i znakami matematycznymi, to owe słowa, symbole i znaki matematyczne nie są adresowane do człowieka, a do komputera. Oznacza to, że choć program komputerowy może być przedmiotem percepcji przez człowieka, to jednak jego zasadniczą funkcją jest sterowanie pracą maszyny (komputera) w celu realizacji określonego zadania lub funkcji. Użytkownik programu jest zaś zazwyczaj zainteresowany nie lekturą programu (tym, jak program jest napisany), lecz jego aspektem funkcjonalnym (tym, jak program działa). To właśnie ów funkcjonalny charakter programów, będących „maszynami skonstruowanymi ze słów”, utrudnia stosowanie tradycyjnych koncepcji prawa autorskiego. Trudności te obrazowo opisał sędzia M. Boudin w zdaniu odrębnym do wyroku w sprawie *Lotus Development v. Borland International*³, wskazując, że „stosowanie prawa autorskiego do programów komputerowych jest jak układanie puzzli, których elementy nie do końca do siebie pasują”. To właśnie niedopasowanie prawa autorskiego do specyfiki programów komputerowych jako przedmiotu ochrony motywuje ustawodawców do wprowadzania regulacji szczególnych i wpływa na kształtowanie się praktyki w sposób odbiegający od innych kategorii twórczości.

¹ Pierwszą ustawą autorską, która *expressis verbis* wymieniła programy komputerowe wśród wytworów mogących podlegać ochronie prawnoautorskiej był dekret prezydenta Filipin nr 49 o ochronie własności intelektualnej z 14.11.1972 r. Programy komputerowe zostały wymienione wśród innych kategorii twórczości w § 2 (N). Zob. T. Dreier, *The International Development of Copyright Protection for Computer Programs* [w:] *A Handbook of European Software Law*, red. M. Lehmann, C.F. Tapper, Oxford 1993, s. 217.

² Przykładowo w 2013 r. wartość amerykańskiego eksportu oprogramowania przekroczyła 120 mld USD, w porównaniu z 25 mld USD uzyskanych z eksportu filmów i produkcji audiowizualnych czy 6 mld USD z nagrań muzycznych. Zob. *Copyright Industries in the U.S. Economy: The 2014 Report*, Washington 2014, s. 20, <https://www.wipo.int/copyright/en/performance/> (dostęp: 1.09.2021 r.). Z kolei wedle danych za 2012 r. we Francji na branżę oprogramowania przypadało 58% wkładu do PKB całego przemysłu autorskiego. Zob. *The Economic Contribution of Copyright Industries in France*, Geneva 2016, s. 52, <https://www.wipo.int/copyright/en/performance/> (dostęp: 1.09.2021 r.). Podobne dane dotyczą innych państw rozwiniętych.

³ *Lotus Development Corp. v. Borland International Inc.*, 49 F.3d 807 (1st Cir. 1995).

Mimo gospodarczego znaczenia i wielu interesujących cech, problematyka prawnoautorskiej ochrony programów komputerowych nie cieszy się dużym zainteresowaniem doktryny. Mające fundamentalne znaczenie monografie J. Barty i R. Markiewicza⁴ oraz A. Nowickiej⁵ zostały opublikowane w pierwszej połowie lat 90. XX w. Późniejsze opracowania miały natomiast charakter komentarzowy lub stanowiły skrótową prezentację problematyki ochrony programów komputerowych, jedynie w niewielkim stopniu przyczyniając się do wzbogacenia stanu badań⁶. Niniejsza monografia ma wypełnić tę lukę. Jej celem jest analiza i ocena prawnoautorskiego modelu ochrony programów komputerowych w tych aspektach, w których różni się on od modelu ochrony utworów należących do innych kategorii twórczości. Oznacza to konieczność zbadania zagadnień związanych z przedmiotem ochrony, podmiotem uprawnionym, treścią praw autorskich i ich ograniczeniami, w tym wyczerpaniem prawa, oraz ochroną technicznych zabezpieczeń. Badania uwzględniają wielopoziomowy charakter współczesnego prawa autorskiego i objęły szczebel krajowy, unijny i międzynarodowy. Nacisk został położony na zagadnienia dotychczas nieporuszane, jak międzynarodowa ochrona programów w TRIPS i TPA, czy niezbadane w dostatecznym stopniu, jak ujęcie przedmiotu ochrony w orzecznictwie TSUE, konstrukcja autorskich praw majątkowych do programu komputerowego, ich ograniczeń i wyczerpania prawa. Ze względu na brak specyfiki zostały natomiast pominięte zagadnienia związane z innymi środkami ochrony, przede wszystkim uregulowanymi w dyrektywie 2004/48/WE Parlamentu Europejskiego i Rady z 29.04.2004 r. w sprawie egzekwowania praw własności intelektualnej, ochroną informacji o prawach, czasem ochrony autorskich praw majątkowych, a także pewnymi innymi unormowaniami wspólnymi dla wszystkich utworów, niemającymi przy tym zasadniczego znaczenia dla ochrony programów, takimi jak wyjątki i ograniczenia przewidziane w dyrektywie Parlamentu Europejskiego i Rady (UE) 2019/790 z 17.04.2019 r. w sprawie prawa autorskiego i praw pokrewnych na jednolitym rynku cyfrowym oraz zmiany dyrektyw 96/9/WE i 2001/29/WE, a także dyrektywie 2006/115/WE Parlamentu Europejskiego i Rady z 12.12.2006 r. w sprawie prawa najmu i użyczenia oraz niektórych praw pokrewnych prawu autorskiemu w zakresie własności intelektualnej.

⁴ J. Barta, R. Markiewicz, *Główne problemy prawa komputerowego*, Warszawa 1993.

⁵ A. Nowicka, *Prawnoautorska i prawnopatentowa ochrona programów komputerowych*, Warszawa 1995.

⁶ Zob. J. Barta, R. Markiewicz [w:] *Prawo autorskie i prawa pokrewne. Komentarz*, red. J. Barta, R. Markiewicz, Warszawa 2011, s. 455–471; J. Barta, R. Markiewicz, A. Matlak [w:] *System Prawa Prywatnego*, t. 13. *Prawo autorskie*, red. J. Barta, Warszawa 2017, s. 1269–1279; M. Byrska, *Ochrona programów komputerowych w świetle prawa polskiego*, Warszawa 1994; Z. Okoń [w:] *Prawo Internetu*, red. P. Podrecki, Warszawa 2007, s. 430–477; Z. Okoń [w:] *Prawo autorskie i prawa pokrewne. Komentarz*, red. D. Flisak, Warszawa 2014, s. 940–1022; K. Sztobryn, *Ochrona programów komputerowych w prawie własności intelektualnej w Unii Europejskiej*, Warszawa 2015, s. 118–182; K. Gienas [w:] *Ustawa o prawie autorskim i prawach pokrewnych. Komentarz*, red. E. Ferenc-Szydełko, Warszawa 2011, s. 416–452; I. Matusiak [w:] *Ustawa o prawie autorskim i prawach pokrewnych. Komentarz*, red. P. Ślęzak, Warszawa 2017, art. 74–77; J. Krysińska [w:] *Prawo autorskie i prawa pokrewne. Komentarz*, red. W. Machała, R.M. Sarbiński, Warszawa 2019, s. 1123–1210; M. Balicki [w:] *Ustawa o prawie autorskim i prawach pokrewnych. Komentarz*, red. A. Michalak, Warszawa 2019, art. 74–77; A. Nowicka [w:] *Ustawy autorskie. Komentarze*, red. R. Markiewicz, Warszawa 2020, s. 1619–1680.

Dobór metod badawczych zastosowanych w monografii jest typowy dla nauk prawnych i obejmuje metodę dogmatyczną, prawnoporównawczą i historyczną. W oparciu o metodę dogmatyczną poddano analizie polskie, unijne i międzynarodowe akty prawne, orzecznictwo TSUE oraz sądów krajowych, a także poglądy polskiej i zagranicznej nauki prawa. Szczególny nacisk został położony na orzecznictwo TSUE, które w ostatnich kilkunastu latach stanowiło główny czynnik stymulujący rozwój prawa autorskiego, w tym w zakresie ochrony programów komputerowych. Ponieważ orzecznictwo Trybunału stanowi formę dialogu z sądami krajowymi, w kluczowych sprawach uwzględniono również kontekst skierowania pytań prejudycjalnych do TSUE oraz sposób, w jaki odpowiedź Trybunału została odczytana przez sąd krajowy i w jaki sąd ten zastosował prawo krajowe zgodnie ze wskazówkami udzielonymi przez TSUE.

Badania prawnoporównawcze objęły przede wszystkim porównanie modelu ochrony programów w prawie autorskim w UE i USA. Wybór prawa USA jako punktu odniesienia dla regulacji unijnej uzasadnia najdłuższa – spośród państw rozwiniętych – historia ochrony programów w ramach prawa autorskiego, siła amerykańskiego przemysłu oprogramowania⁷ i najbogatsze orzecznictwo, stanowiące czynniki sprzyjające oddziaływaniu prawa amerykańskiego na inne systemy prawne. Jednocześnie w europejskiej, w tym polskiej, literaturze poświęca się mu niewiele uwagi i jest przedmiotem nielicznych opracowań⁸. Również w tym zakresie szerzej przedstawiono zagadnienia dotychczas jedynie wzmiankowane w piśmiennictwie, jak treść praw wyłącznych, doktryna *first sale* czy ograniczenia praw wyłącznych dedykowane programom komputerowym. Podobieństwa i różnice między europejskim a amerykańskim modelem ochrony zostały przedstawione w rozdziale XVI. Prawnoporównawczy charakter ma też prezentacja kluczowych rozwiązań służących wdrożeniu dyrektywy Parlamentu Europejskiego i Rady 2009/24/WE z 23.04.2009 r. w sprawie ochrony prawnej programów komputerowych w państwach członkowskich UE i Wielkiej Brytanii. Szczególną uwagę zwrócono przy tym na prawo i orzecznictwo niemieckie. Ustawa niemiecka stanowi przykład wdrożenia dyrektywy Rady 91/250/EWG z 14.05.1991 r. w sprawie ochrony prawnej programów komputerowych, polegającego na dosłownym lub bliskim dosłownemu włączeniu tekstu dyrektywy do ustawy krajowej. Stanowi więc dogodną płaszczyznę do prowadzenia porównań z dyrektywą 91/250/EWG (dyrektywą 2009/24/WE) i polską ustawą, w której posłużono się tą samą techniką legislacyjną.

⁷ Według danych PwC za 2016 r. wśród 10 największych pod względem przychodu dostawców oprogramowania znajduje się tylko jedna firma spoza USA (SAP AG). Wśród pierwszych 50 firm spoza USA pochodziło 13 podmiotów. Zob. *Global 100 Software Leaders by revenue (2016)*, <https://www.pwc.com/gx/en/industries/technology/publications/global-100-software-leaders/explore-the-data.html> (dostęp: 1.09.2021 r.).

⁸ W polskiej literaturze zob. przede wszystkim A. Nowicka, *Prawnoautorska...*, s. 23–59; A. Matlak, *Charakter prawny regulacji dotyczących zabezpieczeń technicznych utworów*, Warszawa 2007, s. 99–117. Ważny wkład w badania nad amerykańskim prawem autorskim, choć już niezwiązany z opisywaną problematyką, stanowi też praca J. Sieńczyło-Chlabicz, *Amerykańskie prawo autorskie. Sytuacja prawna producenta dzieła audiowizualnego według amerykańskiego prawa autorskiego*, Białystok 1998.

Badania historyczne objęły natomiast kształtowanie się prawnoautorskiej ochrony programów komputerowych w USA, TRIPS i TPA, prawie unijnym i prawie polskim, szczególnie względy, które zadecydowały o wyborze prawa autorskiego jako podstawowego reżimu ochrony programów komputerowych oraz przyjęciu rozwiązań, które zróżnicowały ochronę programów komputerowych i innych utworów. Szczególnie ta druga kwestia jedynie w niewielkim stopniu została przedstawiona w piśmiennictwie, również pochodzącym z okresu, kiedy zapadały kluczowe decyzje dotyczące kształtu prawnoautorskiej ochrony programów. Wówczas uwagę doktryny pochłaniało bowiem głównie zagadnienie, „czy”, a nie „jak” programy komputerowe powinny podlegać ochronie prawnoautorskiej. Z tego względu podstawowym źródłem wiedzy są tu dokumenty, analizy studialne i projekty aktów prawnych związane z procesem legislacyjnym mającym na celu uwzględnienie ochrony programów w ramach prawa autorskiego.

Przeprowadzone badania pozwalają na ogólną ocenę adekwatności prawnoautorskiej ochrony programów komputerowych w jej aktualnym kształcie oraz przedstawienie nowych propozycji interpretacyjnych, szczególnie w tych obszarach, w których dotychczasowe ujęcie doktrynalne danego zagadnienia nie pozwala na zachowanie wewnętrznej spójności systemu prawa autorskiego lub zgodności prawa krajowego z prawem unijnym. Obejmuje to przede wszystkim identyfikację koncepcji przedmiotowych granic ochrony prawnoautorskiej w orzecznictwie TSUE, powiązanie orzecznictwa TSUE dotyczącego unijnej przesłanki oryginalności z syntetyczną definicją utworu w Prawie autorskim i siatką pojęciową przyjętą w polskiej doktrynie, propozycję odmiennego ujęcia konstrukcji majątkowych praw autorskich do programu komputerowego, oznaczającą odrzucenie w tym zakresie tzw. modelu własnościowego, zmianę interpretacji pojęcia przeniesienia własności w kontekście wyczerpania prawa oraz nową propozycję konstrukcji wyjątku dotyczącego korzystania z programu zgodnie z przeznaczeniem przez jego legalnego posiadacza. W monografii sformułowano też postulaty *de lege ferenda* adresowane do polskiego ustawodawcy.

Opracowanie składa się z czterech części, poświęconych kolejno prawnoautorskiej ochronie programów komputerowych w prawie USA, porozumieniach międzynarodowych z zakresu prawa autorskiego, prawie Unii Europejskiej i prawie polskim. W każdej części starano się zachować zbliżony układ treści odpowiadający układowi artykułów i podziałowi regulowanej materii w dyrektywie 2009/24/WE i obowiązującej ustawie. Obejmuje on kolejno zagadnienia wstępne i historię regulacji, przedmiot i przesłanki ochrony, podmiot uprawniony, prawa wyłączne przysługujące uprawnionemu (na gruncie polskiej ustawy również autorskie prawa osobiste), wyczerpanie prawa, ograniczenia praw wyłącznych i zagadnienia ochrony technicznych zabezpieczeń programów. Część poświęconą prawu UE kończy rozdział, w którym porównano rozwiązania funkcjonujące w prawie amerykańskim i w prawie UE. Przyjęty układ monografii nie tylko wynika z zakresu omawianej problematyki, ale pozwala również na jej wykorzystanie w charakterze rozbudowanego komentarza do dyrektywy 2009/24/WE i przepisów

rozdziału 7 pr. aut. Waleom praktycznym pracy służy również szerokie uwzględnienie orzecznictwa sądów polskich, zagranicznych i orzecznictwa TSUE.

Względy czytelności zdecydowały o przyjęciu pewnego uproszczenia w powoływaniu dyrektywy 91/250/EWG i jej wersji skodyfikowanej, tj. dyrektywy 2009/24/WE. Ponieważ wersja skodyfikowana przyniosła jedynie zmianę numeracji motywów i artykułów bez istotnych zmian merytorycznych⁹, przy prezentowaniu stanowisk nauki prawa powoływana jest wyłącznie dyrektywa 2009/24/WE, chyba że rozróżnienie obu dyrektyw ma znaczenie merytoryczne. Dotyczy to również przypadków, w których dany pogląd został wyrażony na gruncie dyrektywy 91/250/EWG. Podobna konwencja została przyjęta w odniesieniu do innych dyrektyw posiadających wersje ujednolicone.

Monografia obejmuje stan prawny na dzień 1.05.2022 r.

⁹ Zob. szerzej rozdział X pkt 3.1.

Część I

**OCHRONA PROGRAMÓW KOMPUTEROWYCH
W PRAWIE AUTORSKIM STANÓW
ZJEDNOCZONYCH**

Rozdział I

GENEZA I ROZWÓJ OCHRONY PROGRAMÓW KOMPUTEROWYCH W PRAWIE AUTORSKIM STANÓW ZJEDNOCZONYCH

1. Początki prawnoautorskiej ochrony programów komputerowych

Początki autorskoprawnej ochrony programów komputerowych w USA można datować na rok 1964, kiedy to Copyright Office zajął oficjalne stanowisko odnośnie do dopuszczalności rejestracji programów komputerowych¹. Wskazał przy tym na dwie zasadnicze wątpliwości, towarzyszące ochronie programów komputerowych na gruncie *Copyright Act* z 1909 r. Po pierwsze nie było wówczas jasne, czy programy komputerowe mogą być uznane za „dzieła autorskie” w rozumieniu § 4 *Copyright Act* z 1909 r., a w konsekwencji, czy mogą być uznane za przedmiot prawa autorskiego. Po drugie wątpliwości Copyright Office budziła kwestia, czy program komputerowy zapisany w formie przeznaczonej do wykonania w komputerze może być uznany za „kopię”, która powinna być złożona w celu dokonania rejestracji utworu². Rozstrzygając te wątpliwości na korzyść uprawnionych, Copyright Office wskazał, że istnieje możliwość rejestracji programów komputerowych, jeżeli łącznie będą spełnione następujące warunki: 1) program odznaczać się będzie dostatecznym poziomem ochrony; 2) zostanie opublikowany w postaci opatrzonej odpowiednią notą (tzw. notą copyrightową – *copyright notice*), przy czym „kopie” udostępnione publiczności w ramach publikacji utworu będą czytelne dla człowieka; c) kopie złożone dla celów rejestracji będą zawierać reprodukcję programu w języku zrozumiałym dla człowieka.

¹ Copyright Office Circular 61, 1964, „Bulletin of the Copyright Society of the U.S.A.” 1964/11, s. 361.

² Zob. G.D. Cary, *Copyright Registration and Computer Programs*, „Bulletin of the Copyright Society of the U.S.A.” 1964/11, s. 362; J. Banzhaf, *Copyright Protection for Computer Programs*, Colum. L. Rev. 1964/64, s. 1279; R. Thrumann, *Der Rechtsschutz von Computer-Programmen in den USA*, GRUR Int. 1969/7, s. 207.

Wymogi te w dużym stopniu były zdeterminowane specyficznymi przesłankami przyznania ochrony na gruncie *Copyright Act* z 1909 r. Ustawa ta co do zasady przyznawała ochronę na podstawie ustawy (*statutory protection*) tylko utworom opublikowanym, a więc takim, których kopie zostały udostępnione publiczności, pod warunkiem że zostały one opatrzone prawidłową notą copyrightową³. Możliwość zachowania tych wymogów przez program publikowany w formie kodu źródłowego⁴ nie budziła większych wątpliwości. Inaczej było w przypadku nieprzejrzystego dla człowieka kodu wykonywalnego (maszynowego)⁵. W ocenie Copyright Office program wprowadzony na rynek w tej postaci powinien być uznawany nie tyle za „kopię”, której istnienie warunkuje dojście publikacji do skutku, a raczej – ze względu na swój funkcjonalny charakter – za część urządzenia⁶. W powiązaniu z wątpliwościami, jakie towarzyszyły wykonaniu w takim przypadku obowiązku opatrzenia programu notą copyrightową, w razie publikacji programu wyłącznie w formie kodu wykonywalnego Copyright Office wydawał certyfikaty rejestracji „z zastrzeżeniem wątpliwości” („rule of doubt”). Zastrzeżenie takie oznaczało, że Copyright Office nie przesądzał prawnautorskiej ochrony kodu maszynowego i pozostawiał tę kwestię nierozstrzygniętą do czasu ewentualnego sporu sądowego⁷.

Jeszcze w 1964 r., kiedy to zostały opublikowane wskazane zasady, Copyright Office zarejestrował trzy proste programy⁸. Jednak w rzeczywistości tak sformułowane przesłanki rejestracji programów w Copyright Office czyniły ten reżim ochrony nieatrakcyjnym dla uprawnionych, czego wyrazem była niewielka liczba zgłoszeń (do 1978 r. Copyright Office wydał certyfikaty rejestracji zaledwie dla 1200 programów⁹). Ów brak zaintereso-

³ Na gruncie ustawy z 1909 r. rejestracja w Copyright Office nie była natomiast niezbędna dla przyznania ochrony, choć warunkowała wykonywanie niektórych uprawnień, m.in. wystąpienie z powództwem o naruszenie. Zob. T.A. Reed, *The Role of the Register of Copyrights in the Registration Process: A Critical Appraisal of Certain Exclusionary Regulations*, „University of Pennsylvania Law Review” 1968/8, s. 1380.

⁴ Kod źródłowy to zapis programu w jednym z języków programowania. Język programowania jest językiem formalnym (sztucznym), w wyraźnie określonym podstawowym zbiorze wyrażań (wraz z regułami interpretacji – semantyką) oraz regułami składniowymi. W języku programowania programista tworzy program. Ta forma wyrażenia programu z zasady może być łatwo odczytana przez człowieka, ale nie zastosowana w komputerze. Uruchomienie programu w komputerze wymaga z reguły jego kompilacji, tj. automatycznego przetworzenia programu do postaci kodu wykonywalnego.

⁵ Kod wykonywalny (kod wynikowy, kod maszynowy) to postać zapisu programu przeznaczona bezpośrednio do wykonania przez komputer. Z reguły kod ten jest niemożliwy do odczytania przez człowieka bez dokonania jego dekompilacji, tj. jego „przetłumaczenia” na określony język programowania.

⁶ Rozumowanie takie stało się podstawą jedynego znanego wyroku, zapadłego na gruncie ustawy z 1909 r. Sąd pierwszej instancji w sprawie *Data Cash Sys., Inc. v. JS&A Group, Inc.* (480 F. Supp. 1063, N.D. Ill. 1979) nie uznał kodu utrwalonego w układzie półprzewodnikowym (ROM) za „kopię”, a za część urządzenia, co uzasadniało w tym wypadku odmowę ochrony prawnautorskiej. Wyrok został utrzymany (aczkolwiek z innym uzasadnieniem) w drugiej instancji (628 F.2d.1038, 7th Cir., 1980). Orzeczenie to szerzej omawia C.J. Millard, *Legal protection of Computer Programs and Data*, London 1985, s. 37–39.

⁷ M.A. Lemley, P.S. Menell, R.P. Merges, P. Samuelson, *Software And Internet Law*, New York 2006, s. 34; odnośnie do rejestracji w trybie „rule of doubt” zob. T.A. Reed, *The Role...*, s. 1389.

⁸ *Sixty-seventh Annual Report of the Register of Copyrights*, Washington 1965, s. 4.

⁹ Zob. M.W. Laun, *Improving the International Framework for the Protection of Computer Software*, „University of Pittsburgh Law Review” 1987/48, s. 1163.

wania był prawdopodobnie wynikiem nie tyle dokonywania rejestracji „z zastrzeżeniem wątpliwości”, co konieczności ujawnienia i zdeponowania w Copyright Office pełnego tekstu kodu źródłowego rejestrowanego programu, co z kolei ograniczało możliwość ochrony rozwiązań zawartych w programie jako tajemnicy handlowej zgłaszającego¹⁰. Aby prawnoautorska ochrona programów komputerowych mogła odgrywać większą rolę, niezbędne były zmiany prawa.

2. Raport CONTU i *Computer Software Copyright Act*

W 1976 r. Kongres uchwalił nową ustawę o prawie autorskim (*Copyright Act*). Nie przewidywała ona *expressis verbis* ochrony programów komputerowych, ograniczając się jedynie w § 117 do reguły interpretacyjnej, w świetle której wejście w życie ustawy nie oznaczało przyznania uprawnionym ani większych, ani mniejszych praw odnośnie do korzystania z utworów „w połączeniu z automatycznymi systemami zdolnymi do przechowywania, przetwarzania, wyszukiwania, lub przekazywania informacji”.

Momentem przełomowym dla prawnoautorskiej ochrony programów komputerowych w USA stała się dopiero publikacja w 1978 r. raportu CONTU¹¹, komisji powołanej w 1974 r. przez Kongres w celu oceny nowych zjawisk na obszarze prawa autorskiego, przede wszystkim ochrony programów komputerowych. Powołując się na opinie wyrażone w pracach prowadzonych w tym samym okresie w Wielkiej Brytanii¹² i WIPO¹³, raport wskazywał, że ochrona programów komputerowych w ramach prawa autorskiego była możliwa już na gruncie ustawy z 1976 r. bez konieczności zmiany stanu prawnego. Czynnikiem decydującym była tu interpretacja przesłanek ochrony prawnoautorskiej, która nie stanowiła bariery dla uznania programów komputerowych za utwór podlegający ochronie¹⁴. Z tego względu w raporcie nie rozważano w ogóle *sui generis* ochrony programów komputerowych. Dokonano natomiast porównania modelu prawnoautorskiego z innymi istniejącymi reżimami ochrony, przede wszystkim ochroną patentową oraz ochroną jako tajemnicy handlowej.

W odniesieniu do ochrony patentowej wskazano, że może być ona w niektórych przypadkach bardziej atrakcyjna dla uprawnionych niż ochrona prawnoautorska. Zapewnia

¹⁰ W.F. Patry, *Copyright and Computer Programs: A Failed Experiment and a Solution to a Dilemma*, „New York Law School Law Review” 2002/46, s. 202–203; M.A. Lemley, P.S. Menell, R.P. Merges, P. Samuelson, *Software...*, s. 34.

¹¹ *Final Report of the National Commission on New Technological Uses of Copyrighted Works*, Washington 1978.

¹² Chodzi tu o raport tzw. Komitetu Whitforda. Zob. G. Dworkin, *The Whitford Committee Report on Copyright and Designs Law*, „The Modern Law Review” 1977/6, s. 685.

¹³ Zob. G. Kolle, *Computer Software Protection – Present Situation and Future Prospects*, „Copyright” 1977/3, s. 72.

¹⁴ *Final Report...*, s. 11, 30.

bowiem wyłączność w zakresie używania opatentowanego urządzenia lub procesu nie tylko wtedy, gdy zostały skopiowane, ale również wówczas, gdy zostały niezależnie opracowane przez osobę trzecią. Uzyskanie patentu jest jednak kosztowne i czasochłonne, a zgłoszony wynalazek musi spełniać restrykcyjnie ujęte przesłanki zdolności patentowej i uzyskać pozytywną decyzję USPTO. W dacie przyjęcia raportu CONTU nie było też jasne, w jakim zakresie program komputerowy może w ogóle korzystać z ochrony patentowej. Orzecznictwo Sądu Najwyższego USA wskazywało, że nawet jeżeli wynalazek zakładający użycie programu komputerowego nie zostałby uznany za abstrakcyjny pomysł, które nie korzysta z ochrony patentowej, uzyskanie tej ochrony byłoby w najlepszym wypadku trudne¹⁵.

Z kolei ochrona jako tajemnicy handlowej jest uzależniona od stałego utrzymywania stanu poufności przedmiotu tajemnicy. Ochrona programów jako tajemnicy przedsiębiorstwa może więc odgrywać pewną rolę w zapewnieniu ochrony programów tworzonych na zamówienie dla jednego lub kilku odbiorców. Nie jest jednak właściwym modelem dla ochrony programów przeznaczonych do szerokiej dystrybucji wśród odbiorców masowych. Z perspektywy użytkowników oprogramowania utrzymywanie stanu tajemnicy wymaga ograniczenia przepływu informacji, m.in. na podstawie umów o zachowanie poufności, co prowadzi do zwiększenia kosztów związanych z pozyskaniem oprogramowania i utrudnia dokonywanie porównań konkurencyjnych ofert¹⁶.

Tych wad pozbawiona jest ochrona prawnoautorska. Jest odformalizowana i bezpłatna. Jej przesłanki są łatwe do spełnienia przez programy dostępne na rynku i nie koliduje ona z masową dystrybucją oprogramowania. Jej zakres umożliwia co najmniej zwalczanie piractwa, polegającego na kopiowaniu kodu wynikowego programów. Z tych względów większość członków CONTU opowiedziała się za ochroną programów komputerowych w ramach prawa autorskiego¹⁷. W celu zwiększenia pewności prawa zalecono jednak wprowadzenie do *Copyright Act* niewielkich zmian. Propozycje obejmowały wyraźne przesądzenie, że programy komputerowe w zakresie, w jakim stanowią rezultat oryginalnej twórczości autora, podlegają ochronie prawnoautorskiej, skreślenie dotychczasowego § 117 *Copyright Act* w celu zapewnienia ochrony wszystkim programom komputerowym oraz przyznanie legalnemu posiadaczowi kopii programu prawa do używania tej kopii i jej adaptacji¹⁸.

¹⁵ Szczególne znaczenie miał wyrok Sądu Najwyższego USA *Gottschalk v. Benson*, 409 U.S. 63 (1972), który został odczytany jako zakaz patentowania programów komputerowych. Zob. G.A. Stobbs, *Software patents*, Gaithersburg–New York 2000, s. 46. W raporcie CONTU przywołano również wyrok *Dann v. Johnston*, 425 U.S. 219 (1976) i wyrok *Parker v. Flook*, 437 U.S. 584 (1978), a także powołano się na wyłączenie programów „jako takich” spod ochrony w Konwencji o patencie europejskim (Dz.U. z 2004 r. Nr 79, poz. 737) i doświadczenia sądów niemieckich. Zob. *Final Report...*, s. 17.

¹⁶ *Final Report...*, s. 17.

¹⁷ Zob. R.T. Nimmer, *Law of Computer Technology*, vol. 1, Toronto 2012, § 1:3.

¹⁸ *Final Report...*, s. 1.

Zalecenia raportu CONTU zostały zrealizowane w nowelizacji *Copyright Act*, dokonanej w 1980 r.¹⁹ Nowelizacja ta polegała na uzupełnieniu § 101 *Copyright Act* o definicję programu komputerowego, wzorowaną na przepisach wzorcowych WIPO²⁰, oraz nadaniu nowej treści § 117 *Copyright Act*. Program komputerowy zdefiniowano jako zestaw poleceń lub instrukcji, przeznaczonych bezpośrednio lub pośrednio do użycia w komputerze w celu osiągnięcia określonego rezultatu. Z kolei w § 117 *Copyright Act* zamieszczono ograniczenie praw wyłącznych, zgodnie z którym właściciel kopii programu jest uprawniony do wykonania dodatkowej kopii lub adaptacji, jeżeli jest ona tworzona jako „istotny krok” (*essential step*) w korzystaniu z programu komputerowego w połączeniu z maszyną lub służą do celów archiwalnych. Nowelizacja nie zmieniła natomiast zawartego w § 102 (1) *Copyright Act* katalogu kategorii utworów podlegających ochronie prawnoautorskiej. Dodanie odrębnej kategorii programów komputerowych uznano bowiem za *superfluum* w sytuacji, w której na gruncie *Copyright Act* mogły one podlegać ochronie jako „utwory literackie”²¹. Doprecyzowanie zakresu ochrony pozostawiono sądom.

Raport CONTU nie został przyjęty jednogłośnie²². Obszerne zdanie odrębne zgłosił członek komisji J. Hersey, którego zdaniem „wtłoczenie” programów komputerowych zapisanych w formie kodu wynikowego w ramy prawa autorskiego jest niewłaściwe, niepotrzebne i stwarza ryzyko erozji tej dziedziny prawa. Stanowisko to jest warte uwagi. Nie tylko podsumowuje ono wątpliwości zgłaszane w latach 70. XX w. wobec objęcia programów komputerowych ochroną prawa autorskiego, ale stało się podstawą bez mała każdej późniejszej krytyki tego modelu ochrony.

Zasadniczy argument podniesiony przez J. Herseya był następujący. We wczesnych fazach rozwoju programu komputerowego (podczas tworzenia dokumentacji projektowej lub kodu źródłowego) podstawowe idee i metody, które mają być w nim zawarte, są określone w postaci zapisu tekstowego i w tej postaci zapewne mogą być chronione na podstawie *Copyright Act* bez konieczności wprowadzania do niego zmian. Jednak ukończony program, w swej ostatecznej i użytecznej postaci (tj. w formie kodu wynikowego), jest częścią maszyny (komputera), sterującą jej pracą i nie powinien podlegać ochronie prawnoautorskiej. Ta dwoista natura programów komputerowych sprawia, że znacznie różnią się one od dzieł podlegających ochronie prawnoautorskiej. Dzieła autorskie zawsze były przeznaczone do percepcji przez człowieka – ich przeczytania, wysłuchania lub obejrzenia. Nawet wówczas, gdy percepcja ta wymaga użycia specjalnego urządzenia w celu odtworzenia np. zapisu dźwiękowego na taśmie magnetycznej, ostatecznym rezultatem odtworzenia zapisu jest możliwość zapoznania się z nim przez człowieka. Inaczej

¹⁹ *Computer Software Copyright Act*, Law No. 96-517, 1980.

²⁰ Zob. szerzej rozdział VI pkt 1.

²¹ Zob. J. Drexler, *What is Protected in a Computer Program?: Copyright Protection in the USA and Europe*, Munich 1994, s. 16.

²² Zob. C.F.I. Koenig, *Software Copyright: The Conflict within CONTU*, „Bulletin of the Copyright Society of the U.S.A.” 1979/27 s. 340; M.D. Scott, *Information Technology Law*, vol. 1, Austin 2012, § 2.04.

jest w przypadku programów komputerowych. Zawierają one polecenia adresowane nie do człowieka, a do maszyny (komputera). Uruchomione w komputerze nie komunikują użytkownikowi treści programu, a wykonują określone działania w celu osiągnięcia założonego rezultatu. Bezpośrednim skutkiem uruchomienia programu jest więc seria impulsów elektronicznych, które sterują pracą komputera w celu wykonania określonego zadania. Ta specyfika programów powoduje, że objęcie ochroną prawnoautorską programów w formie wynikowej jest sprzeczne z celami ochrony prawnoautorskiej i zaburza jej podstawowe pojęcia. W połączeniu z siłą przemysłu komputerowego może więc prowadzić do erozji fundamentów prawa autorskiego²³.

Obawy te podzielał również inny członek komisji – M. Nimmer. Mimo poparcia dla wniosków i zaleceń CONTU zaznaczył, że niepokojące jest niewyartykułowanie przez komisję jakiegokolwiek uzasadnienia, które nie pozwalałoby na ochronę w ramach prawa autorskiego każdego innego wyrażenia jakichkolwiek oryginalnych pomysłów. Uznanie programów komputerowych za dzieła literackie prowadzi do tak szerokiej interpretacji tego pojęcia, że prawo autorskie może potencjalnie znaleźć zastosowanie w odniesieniu do wytworów intelektualnych tradycyjnie zaliczanych do domeny prawa patentowego, a nawet takich, których dotąd nie obejmowano ani ochroną prawnoautorską, ani patentową (np. metod prowadzenia działalności gospodarczej)²⁴.

Prawnoautorska ochrona programów komputerowych i założenia przyjęte w raporcie CONTU były krytycznie oceniane przez część środowisk akademickich²⁵. Na przełomie lat 80. i 90. XX w. negatywnym ocenom tego modelu ochrony sprzyjał kierunek, w jakim zmierzało orzecznictwo. Rozwinięcie, omówionej w następnym rozdziale, koncepcji ochrony struktury, sekwencji i organizacji (SSO) programu komputerowego oraz orzecznictwo dotyczące interfejsów programów silnie wypaczyły zasady ochrony prawnoautorskiej. Jednocześnie zwrot, jaki nastąpił w sprawie *Altai*²⁶, wywołał obawy o pozostawienie poza zakresem ochrony pewnych aspektów funkcjonalnych i strukturalnych oprogramowania, co postrzegano jako potencjalne zagrożenie dla interesów producentów oprogramowania. Narastający kryzys znalazł odbicie również w szeroko komentowanym²⁷ raporcie, przygotowanym w 1992 r. przez Office of Technology Assessment na zlecenie Kongresu. W raporcie stwierdzono, że choć ochrona prawnoautorska obejmuje ochronę przed literalnym kopiowaniem kodu programu, wywołuje jednak niepewność co do jej granic. W razie podjęcia dalszych prac legislacyjnych

²³ *Final Report...*, s. 29–37.

²⁴ *Final Report...*, s. 26.

²⁵ Zob. m.in. P. Samuelson, *CONTU revisited: The case against copyright protection for computer programs in machine-readable form*, Duke LJ 1984, s. 666 i n.; P.S. Menell, *Tailoring legal protection for computer software*, Stan. L. Rev. 1986/39, s. 1329.

²⁶ *Computer Associates International Inc. v. Altai* 23 U.S.P.Q. 2d 1241 (2d Cir. 1992). Zob. szerzej rozdział II pkt 3.2.

²⁷ W polskiej literaturze zob. J. Barta, R. Markiewicz, *Główne...*, s. 55–56; A. Nowicka, *Prawnoautorska...*, s. 24.

raport wskazywał na szerokie spektrum możliwych rozwiązań, od wyraźnego potwierdzenia zasad wypracowanych w orzecznictwie w ramach prawa autorskiego, przez wprowadzenie uzupełniającego reżimu ochrony *sui generis*, odnoszącego się do tych aspektów funkcjonalnych oprogramowania, które pozostają zarówno poza zakresem ochrony patentowej, jak i prawnoautorskiej, aż po wyłączenie ochrony prawnoautorskiej i wprowadzenie w to miejsce dostosowanej do specyfiki programów komputerowych ochrony *sui generis*²⁸. Odrzucenie ochrony prawnoautorskiej i zastąpienie jej ochroną *sui generis* postulowali też niektórzy przedstawiciele nauki prawa²⁹.

Krytyka modelu prawnoautorskiego i postulaty formułowane w raporcie Office of Technology Assessment i doktrynie nie doprowadziły jednak ani do sformułowania konkretnych propozycji legislacyjnych, ani do zmian ustawodawczych. Wpłynęło na to kilka czynników. Po pierwsze ochrona prawnoautorska umożliwiała uzyskanie ochrony o zasięgu międzynarodowym. Dla USA jako wiodącego eksportera oprogramowania³⁰ ten aspekt miał szczególnie istotne znaczenie. Długoletnie wysiłki Stanów Zjednoczonych na forum międzynarodowym, mające na celu promowanie tego modelu ochrony, doprowadziły w tym okresie do przełomu w czasie rundy urugwajskiej GATT i podpisania porozumienia TRIPS, które ujednoliciło interpretację konwencji berneńskiej w państwach, które przystąpiły do WTO³¹. Po drugie brak było przekonujących dowodów, że odejście od ochrony programów komputerowych w ramach prawa autorskiego rzeczywiście miałoby pozytywny wpływ na rozwój przemysłu software'owego i przyniosłoby korzyści gospodarcze. Należy zgodzić się z J.C. Ginsburg, że wady ochrony programów komputerowych w ramach prawa autorskiego były wyolbrzymiane, natomiast nie doceniano trudności, jakie wiązałyby się z wprowadzeniem nowego instrumentu prawnego³². Po trzecie poczynszy od lat 90. XX w. postępował proces liberalizacji oceny zdolności patentowej wynalazków software'owych, co pozwalało na ochronę niektórych aspektów funkcjonalnych oprogramowania w ramach prawa patentowego³³. Poszukiwanie „trzeciej drogi” pomiędzy ochroną prawnoautorską a patentową straciło więc na znaczeniu.

²⁸ *Finding a Balance: Computer Software, Intellectual Property and the Challenge of Technological Change*, Washington 1992, s. 29.

²⁹ Zob. P. Samuelson, R. Davis, M. Kapor, J.H. Reichman, *A Manifesto Concerning the Legal Protection of Computer Programs*, Colum. L. Rev. 1994/12, s. 2310; K. Nichols, *Inventing software: the rise of „computer-related” patents*, Westport, Connecticut–London 1998, s. 132.

³⁰ Udział Stanów Zjednoczonych w światowym przemyśle oprogramowania wynosił 70% w 1982 r. i 57% w 1990 r. Zob. M. Campbell-Kelly, *A History of the Software Industry*, Cambridge–London 2003, s. 22–23.

³¹ Zob. szerzej rozdział VII pkt 1.

³² J.C. Ginsburg, *The Manifest Superiority of Copyright over Sui Generis Protection of Computer Software*, Colum. L. Rev. 1994/12, s. 2259 i n. Zob. też B.A. Carlson, *On the Wrong Track: A Response to the Manifesto and a Critique of Sui Generis Software Protection*, „Jurimetrics” 1997/2, s. 187.

³³ Już po publikacji raportu CONTU został wydany wyrok Sądu Najwyższego USA *Diamond v. Diehr*, 450 U.S. 175 (1981), w którym orzeczono, że kontrolowanie wykonania procesu fizycznego przez uruchomienie programu komputerowego nie wyklucza możliwości opatentowania wynalazku jako całości. Bariery dla szerszej patentowalności wynalazków software'owych wynikały jednak z testu Freeman-Walter-Abele, sformułowanego w początku lat 80. XX w. przez CCPA w sprawach *In re Freeman* 573 F.2d 1237 (197 USPQ 464), *In re Walter* 618 F.2d 758 (205 USPQ 397) i *In re Abele* 684 F.2d 902 (214 USPQ 682) (zob.

3. *Computer Software Rental Amendments Act*

Po raz pierwszy regulacja *Copyright Act* odnosząca się do programów komputerowych została uzupełniona nowelizacją z 1990 r. (*Computer Software Rental Amendments Act*)³⁴. Nowelizacja zawierała trzy grupy przepisów: odnoszące się do najmu programów komputerowych, regulujące korzystanie z automatów do gier video oraz przewidujące prowadzenie przez Copyright Office rejestru dokumentów związanych z programami typu *shareware* i *public domain*.

Największą nowością było nadanie nowego brzmienia § 109 (b) *Copyright Act*, w wyniku czego wyłączono z zakresu wyczerpania prawa dystrybucji egzemplarza najem i użyczenie programów komputerowych. Konsekwencją tej zmiany było umożliwienie uprawnionym sprawowania kontroli nad najmem, użyczeniem i innymi podobnymi formami eksploatacji egzemplarza mimo dokonania pierwszej sprzedaży danego egzemplarza programu i wyczerpania prawa. Jednocześnie poza zakresem praw wyłącznych pozostawiono najem i użyczenie urządzeń zawierających wbudowane programy komputerowe, najem i użyczenie programów komputerowych wbudowanych lub używanych w połączeniu z konsolami do gier video, oraz użyczenie programów komputerowych przez biblioteki. Przyjęcie tej regulacji miało wpływ na stanowisko Stanów Zjednoczonych w czasie negocjacji TRIPS oraz TPA. Stała się ona pierwowzorem zawartych w tych porozumieniach rozwiązań w zakresie prawa najmu programów komputerowych³⁵.

Druga grupa przepisów miała na celu ułatwienie działania podmiotów eksploatujących automaty do gier zręcznościowych. Zgodnie z § 106 *Copyright Act* podmiotowi praw autorskich przysługuje wyłączne prawo publicznego wykonywania i wyświetlania utworów audiowizualnych. W świetle orzecznictwa sądów poprzedzającego uchwalenie omawianej nowelizacji, umieszczenie w miejscu publicznym automatu do gier video oznaczało wystawienie i wyświetlenie utworu audiowizualnego. Kwestia ta miała przede wszystkim znaczenie w przypadku obrotu automatami „z drugiej ręki”, uprawnienia przewidziane w § 106 *Copyright Act* nie ulegały bowiem wyczerpaniu. W konsekwencji ustawienie w miejscu publicznym automatu nabytego na rynku wtórnym mogło prowadzić do naruszenia prawa autorskiego³⁶. W celu zapobiegania takim sytuacjom nowelizacja uzupełniła *Copyright Act* o § 109 (e), który zezwala posiadaczowi „legalnej” kopii gry video (w tym posiadaczowi automatu zawierającego wbudowaną grę) do

A. Nowicka, *Prawnoautorska...*, s. 151, 157–170). Został on odrzucony w sprawach *In re Alappat*, 33 F.3d 1526 (Fed. Cir. 1994), *In re Lowry*, 32 F.3d 1579 (Fed. Cir. 1994) i *State Street Bank & Trust Co. v. Signature Financial Group, Inc.*, 149 F.3d 1368 (Fed. Cir. 1998) (zob. M. Porzeżyński, *Zdolność patentowa programów komputerowych*, Warszawa 2017, s. 223).

³⁴ Law No. 101-650, 1990.

³⁵ *Finding a balance...*, s. 113–114.

³⁶ Zob. np. wyrok *Red Baron-Franklin Park, Inc. v. Taito Corp.*, 883 F.2d 275, 11 U.S.P.Q.2d (BNA) 1548 (4th Cir. 1989); por. E. Finkel, *Copyright Protection for Computer Software in the Nineties*, HTLJ 1991/7, s. 283.

publicznego wykonywania i wyświetlania przedstawień audiowizualnych generowanych przez taką grę.

Wreszcie w *Computer Software Rental Amendments Act* znalazły się przepisy odnoszące się do prowadzenia przez Copyright Office rejestru dokumentów związanych z programami typu *shareware*³⁷ i *public domain*³⁸. W pierwszym przypadku rejestr obejmuje umowy licencyjne dołączane do oprogramowania, w drugim – oświadczenia o zrzeczeniu się praw autorskich, skutkujące przeniesieniem programu do domeny publicznej. Znaczenia tych przepisów upatruje się przede wszystkim w potwierdzeniu w akcie prawnym skuteczności umów stosowanych przy dystrybucji *shareware*³⁹. Poza tym wpływ istnienia tego rejestru na praktykę obrotu okazał się znikomy.

4. Digital Millennium Copyright Act

Po raz kolejny przepisy dotyczące ochrony programów komputerowych zostały zmienione przez *Digital Millennium Copyright Act* z 1998 r. Z punktu widzenia omawianej problematyki znaczenie ma tytuł I DMCA, obejmujący dostosowanie prawa USA do TPA i TAWF, oraz tytuł III DMCA, regulujący korzystanie z programów dla celów serwisu i obsługi sprzętu (*Computer Maintenance Competition Assurance Act*).

Digital Millennium Copyright Act, mimo że formalnie uchwalono w celu implementacji traktatów WIPO, zawiera rozwiązania daleko wykraczające poza traktatowe minimum ochrony. Do najbardziej kontrowersyjnych rozwiązań należy wprowadzenie ochrony skutecznych środków technicznych używanych przez uprawnionych w celu ochrony utworów przed nieautoryzowanym dostępem lub kopiowaniem. Ochrona obejmuje zakaz obchodzenia tego zabezpieczenia przed dostępem do dzieła (§ 1201 (a) (1) (A) DMCA) oraz zakaz wytwarzania lub dystrybucji środków technicznych zaprojektowanych lub wytworzonych przede wszystkim dla celów obchodzenia technicznych zabezpieczeń utworów przed nieautoryzowanym dostępem i niedozwolonym kopiowaniem utworów (odpowiednio § 1201 (a) (2) i § 1201 (b) (1) DMCA)⁴⁰. Inaczej niż w prawie europejskim regulacja ta znajduje zastosowanie w odniesieniu do wszystkich skutecznych zabezpieczeń technicz-

³⁷ *Shareware* to sposób dystrybucji oprogramowania komputerowego. Polega on na zezwoleniu każdemu na rozpowszechnianie programu komputerowego. Dzięki temu użytkownicy końcowi mają możliwość jego wypróbowania przed zakupem. Zazwyczaj po okresie próbnym lub po określonej liczbie uruchomień użytkownik powinien uzyskać płatną licencję lub zrezygnować z korzystania z programu.

³⁸ W ten sposób określa się oprogramowanie udostępnione publiczności wraz z oświadczeniem autora o zrzeczeniu się praw autorskich.

³⁹ Zob. szerzej C.T. Turkington, *A Comparison of Shareware and Other Software Licensing: Role of the New Copyright Office Shareware Registry*, U. Balt. Intell. Prop. L.J. 1992/1, s. 89; E. Samuels, *The Public Domain in Copyright Law*, J. Copyright Soc'y USA 1993/41, s. 159.

⁴⁰ Odnośnie do genezy i rozwoju tej regulacji zob. Z. Efroni, *A Momentary Lapse of Reason: Digital Copyright, the DMCA and a Dose of Common Sense*, Colum. J.L. & Arts 2005/1, s. 250. W polskiej literaturze zob. zwłaszcza A. Matlak, *Charakter...*, s. 99–106.

nych, bez względu na kategorię twórczości, do której należy zabezpieczony utwór, a więc także do programów komputerowych. Programów dotyczy jednak również szczególny wyjątek, wyraźnie wzorowany na rozwiązaniach europejskich w zakresie dekompilacji, tj. art. 6 dyrektywy 2009/24/WE, zezwalający na przełamanie technicznych zabezpieczeń, jeżeli jest to niezbędne dla zachowania interoperacyjności i osiągnięcia współdziałania niezależnie od stworzonego programu komputerowego (§ 1201 (f) DMCA)⁴¹.

Tytuł III DMCA (*Computer Maintenance Competition Assurance Act*) uzupełnił § 117 *Copyright Act* o zezwolenie na zwielokrotnienie programu komputerowego, jeśli do zwielokrotnienia takiego dochodzi w związku z uruchomieniem urządzenia, zawierającego „legalną” kopię programu komputerowego, a uruchomienie następuje w celu konserwacji lub naprawy tego urządzenia. Regulacja ta odzwierciedla wątpliwości, jakie pojawiły się po orzeczeniu *MAI Sys. Corp. v. Peak Computer, Inc.*⁴², w którym sąd uznał czasowe wprowadzenie programu do pamięci operacyjnej komputera (RAM), dokonane w czasie serwisowania komputera przez zewnętrznego dostawcę usług serwisowych, za naruszenie prawa autorskiego. Skutkiem nowelizacji jest usunięcie wątpliwości co do oceny prawnej świadczenia usług serwisu i utrzymania systemów informatycznych przez firmy zewnętrzne.

5. Podsumowanie

Wybór modelu prawnoautorskiego jako podstawowego reżimu ochrony programów komputerowych w USA podyktowany był względami praktycznymi. Wymagało to jedynie niewielkich zmian w *Copyright Act*, sprowadzających się do dodania definicji programu komputerowego i specyficznych ograniczeń praw wyłącznych dotyczących sytuacji prawnej właściciela kopii programu i sporządzania kopii archiwalnych. Jednocześnie model prawnoautorski oferował ochronę pozbawioną opłat, odformalizowaną, a tym samym dostępną dla szerokiego kręgu podmiotów i obejmującą w zasadzie wszystkie programy oferowane na rynku. Po uchwaleniu w 1980 r. zmian w *Copyright Act* dalsza ewolucja regulacji ustawowej doprowadziła do poszerzenia katalogu wyjątków o przypadki związane z najmem urządzeń zawierających program komputerowy i uruchamianiem programu dla potrzeb serwisu sprzętu. Z punktu widzenia balansu interesów uprawnionych i użytkowników programów istotne znaczenie miało też wprowadzenie przez *Digital Millennium Copyright Act* szerokiej ochrony technicznych zabezpieczeń utworów, co objęło również zabezpieczenia programów komputerowych.

⁴¹ Zob. J.L. Davis, *Is Interoperability Just for Those Who Can Hack It – The Application of the DMCA Interoperability Exceptions in the Consumer Electronics Industry*, „University of Illinois Journal of Law, Technology & Policy” 2005/1, s. 151 i n.

⁴² *MAI Sys. Corp. v. Peak Computer, Inc.*, 991 F.2d 511, 26 U.S.P.Q.2d (BNA) 1458 (9th Cir. 1993).

Pewne zainteresowanie alternatywami dla modelu prawnoautorskiego można było zaobserwować jedynie w połowie lat 90. XX w., wraz z odrzuceniem w orzecznictwie możliwości ochrony w ramach prawa autorskiego pewnych aspektów funkcjonalnych i strukturalnych oprogramowania. Ponieważ w tym okresie z jednej strony doszło do wyraźnego potwierdzenia międzynarodowej ochrony programów na podstawie konwencji berneńskiej, co zwiększało atrakcyjność modelu prawnoautorskiego, z drugiej pewnej liberalizacji praktyki organów patentowych w zakresie oceny wynalazków software'owych, postulaty utworzenia alternatywnego reżimu ochrony, dedykowanego wyłącznie programom komputerowym, nie spotkały się z szerszą reakcją i nie przełożyły na konkretne propozycje legislacyjne.

Rozdział II

PRZEDMIOTOWY ZAKRES OCHRONY

1. Uwagi wprowadzające

W świetle definicji legalnej, zawartej w § 101 *Copyright Act*, program komputerowy to zestaw poleceń lub instrukcji, przeznaczony do użycia bezpośrednio lub pośrednio w komputerze w celu uzyskania określonego rezultatu. Tak sformułowana definicja niewiele w istocie mówi o przedmiotowym zakresie ochrony. Nie przesądza nawet możliwości prawnautorskiej ochrony programów wyrażonych w formie kodu wynikowego czy utrwalonych w układzie scalonym. Dlatego w pierwszym okresie rozwoju orzecznictwa, trwającym mniej więcej do połowy lat 80. XX w., sądy musiały przede wszystkim udzielić odpowiedzi na pytanie o zakres ochrony przed „literalnym” kopiowaniem. Nie budziło przy tym wątpliwości, że „literalne” kopiowanie, tj. przejęcie fragmentu lub całości cudzego utworu, stanowi naruszenie. Kwestie te jednoznacznie rozstrzyga *Copyright Act*, stanowiąc, że każdy taki przypadek, z wyjątkiem czynności objętych ograniczeniami wynikającymi z § 117 *Copyright Act*, narusza wyłączne prawo do reprodukcji przysługujące uprawnionemu. Chodziło tu raczej o rozstrzygnięcie, w jakiej formie i jakiego rodzaju programy komputerowe mogą podlegać ochronie prawnautorskiej¹.

Drugi etap rozwoju orzecznictwa, w którym najważniejsze judykaty przypadają na lata 1985–1995, obejmował ustalenie, w jakim stopniu ochrona prawnautorska obejmuje również ochronę przed „nieliteralnym” kopiowaniem. Na problem „nieliteralnego” kopiowania składa się kilka zbliżonych, choć z europejskiego punktu widzenia odrębnych zagadnień, których wspólnym mianownikiem jest ocena, kiedy – mimo braku identyczności kodu dwóch programów – może dojść do naruszenia prawa autorskiego². Po pierwsze chodzi tu o wytyczenie granicy pomiędzy niechronioną ideą (elementami koncepcyjnymi) a chronioną formą wyrażenia (ekspresją) dzieła. Po drugie chodzi o udzielenie odpowiedzi na pytanie, czy podobieństwo pewnych „zewnętrznych” cech programu komputerowego, takich jak realizowane funkcje, używane protokoły komu-

¹ J. Drexler, *What is Protected...*, s. 16.

² Zob. R.T. Nimmer, *Law...*, § 1:5.

nikacyjne, formaty plików czy wreszcie obrazy wyświetlane na ekranie w celu komunikacji z użytkownikiem, składające się na interfejs użytkownika, mogą prowadzić do naruszenia prawa do programu komputerowego mimo braku podobieństwa kodu. W tej drugiej kwestii wątpliwości dotyczyły nie tylko rozróżnienia między wyrażeniem a ideą, ale również możliwości uznania elementów „pozatekstowych” (wykraczających poza kod programu, np. elementów interfejsu użytkownika czy obrazów wyświetlanych na ekranie) za część programu komputerowego podlegającego ochronie.

Oceniając orzecznictwo pochodzące z tego okresu, trzeba uwzględnić specyfikę prawa amerykańskiego. Rozróżnienie chronionego wyrażenia (ekspresji) i niechronionej idei wynika z § 102 (b) *Copyright Act*. Poza zakresem ochrony pozostają „idee, procedury, systemy, sposoby postępowania, koncepcje, zasady lub odkrycia, bez względu na formę, w jakiej zostały opisane, wyjaśnione, zilustrowane lub ucieleśnione”³. W celu ułatwienia identyfikacji przedmiotowych granic ochrony w orzecznictwie korzysta się również z tzw. doktryn ograniczających (*limiting doctrines*). Dla omawianej problematyki największe znaczenie mają *merger doctrine* i *scenes à faire doctrine*. Pierwsza doktryna zakłada, że wyrażenie nie podlega ochronie, jeżeli doszło do „stopienia się” wyrażenia z ideą. Wówczas bowiem przyznanie ochrony takiemu wyrażeniu nieuchronnie prowadziłoby do przyznania ochrony idei. Druga z kolei opiera się na założeniu, że poza zakresem ochrony powinny pozostawać pewne wątki, konwencje czy schematy niezbędne lub co najmniej typowe dla danej dziedziny twórczości. Obie doktryny mają podobny cel, jakim jest zapobieganie monopolizacji idei. Doktryna *merger* powinna jednak znaleźć zastosowanie tam, gdzie idea i wyrażenie są nierozłączne, tj. nie ma innego sposobu wyrażenia idei lub też wybrane wyrażenie jest jednym z niewielu możliwych. Doktryna *scenes à faire* dotyczy zaś przypadków, w których zakres dostępnych sposobów wyrażenia jest szerszy, ale czynniki zewnętrzne powodują wykorzystanie pewnych wspólnych elementów, a tym samym podobieństwo wyrażenia porównywanych utworów⁴.

Sądy amerykańskie stosują również różnego rodzaju tzw. testy abstrakcji, mające wytyczyć granicę pomiędzy chronionym wyrażeniem a niechronioną, abstrakcyjną ideą⁵. Stanowią one pomoc zwłaszcza w tych przypadkach, w których nie doszło do literalnego skopiowania cudzej pracy, jednak między porównywanymi utworami istnieją dostrzegalne podobieństwa. W sporach dotyczących programów komputerowych zastosowanie znalazły zarówno testy wypracowane wcześniej w orzecznictwie w odniesieniu do innych kategorii twórczości, jak i takie, które sformułowano przy uwzględnieniu specyfiki

³ § 102 (b) *Copyright Act* uznawany jest za kodyfikację zasady wyrażonej po raz pierwszy w orzeczeniu *Baker v. Selden*, 101 U.S. 99 (1879), w którym amerykański Sąd Najwyższy uznał, że prawo autorskie do książki opisującej metodę księgowania nie rozciąga się na metodę jako taką. Kolejnym ważnym precedensem był wyrok *Mazer v. Stein*, 347 U.S. 201, 217 (1954), zgodnie z którym ochrona przedmiotów artystycznych obejmuje tylko elementy estetyczne, a nie mechaniczne lub funkcjonalne.

⁴ Zob. wyrok *CMM Cable Rep., Inc. v. Ocean Coast Properties, Inc.*, 97 F.3d 1504 (1st Cir. 1996).

⁵ Zob. J. Velasco, *The Copyrightability of Nonliteral Elements of Computer Programs*, Colum. L. Rev. 1994 s. 254; A. Nowicka, *Prawnoautorska...*, s. 37.

programów. Dobór i sposób przeprowadzenia testu oraz interpretacja „doktryn ograniczających” bezpośrednio wpływają na przedmiotowe granice ochrony, a tym samym na zakres wyłączności przysługującej uprawnionemu. Orzecznictwo amerykańskie przeszło w tym zakresie długą ewolucję, będącą odbiciem zmieniających się potrzeb przemysłu software'owego i zmiany spojrzenia na rolę prawnoautorskiej ochrony programów komputerowych.

2. Ochrona przed literalnym kopiowaniem

Problem ochrony literalnych elementów programu komputerowego sprowadzał się do udzielenia odpowiedzi na pytanie, czy ochronie podlega program wyrażony zarówno w formie kodu źródłowego, jak i w formie kodu wykonywalnego, w tym kodu trwale połączonego ze sprzętem przez jego zapisanie w pamięci nieulotnej (ROM), oraz jakie znaczenie dla ochrony prawnoautorskiej ma kwestia, czy dany program należy uznać za system operacyjny komputera czy za program aplikacyjny.

Po wejściu w życie *Copyright Act* z 1976 r. pierwszym rozstrzygnięciem sądu apelacyjnego odnoszącym się do kwestii ochrony kodu zapisanego w pamięci ROM był wyrok w sprawie *Williams Electronics, Inc. v. Arctic International, Inc.* Sąd apelacyjny 3 okręgu jednoznacznie odrzucił w nim argument, że ochrona prawnoautorska obejmuje wyłącznie kopie utworu wykonane w sposób umożliwiający ich percepcję przez człowieka i stanowiące środek komunikacji z człowiekiem. Zdaniem sądu przyjęcie takiej interpretacji prowadziłoby do powstania luki prawnej, w wyniku której naruszenie praw do programu komputerowego ograniczałoby się wyłącznie do reprodukcji kodu źródłowego programu komputerowego, ale nie do powielania programu komputerowego utrwalonego na układzie scalonym. Byłoby to sprzeczne z wolą prawodawcy, który wyraźnie zamierzał objąć programy komputerowe ochroną prawnoautorską⁶. Z kolei w wyrokach w sprawie *GCA Corp. v. Chance*⁷ oraz *Midway Manufacturing Co. v. Strohon*⁸ wskazano, że kod wynikowy jest jedynie mechaniczną transformacją programu komputerowego i podlega ochronie, jeżeli został uzyskany z kodu źródłowego stanowiącego przedmiot prawa autorskiego⁹.

⁶ *Williams Elecs., Inc. v. Artic Int'l, Inc.*, 1981 WL 1287 (D.N.J. June 24, 1981); 685 F.2d 870 (3d Cir. 1982). Podobny wniosek wynika z wyroku *Tandy Corp. v. Personal Micro Computers, Inc.*, 524 F.Supp. 171, 175 (N.D.Cal.1981).

⁷ *GCA Corp. v. Chance* 17 U.S.P.Q. (BNA) 718 (N.D. Cal. 1982).

⁸ *Midway Manufacturing Co. v. Strohon*, 564 F.Supp. 741, 746 (N.D.Ill. 1983).

⁹ Zob. A.G. Rodau, *Protecting Computer Software: After Apple Computer, Inc. v. Franklin Computer Corp.*, 714 F.2d 1240 (3d Cir. 1983), *Does Copyright Provide the Best Protection*, „Temple Law Quarterly” 1984/57, s. 527. Podobne stanowisko znalazło wyraz również w wyroku *Stern Electronics, Inc. v. Kaufmann*, 699 F.2d 852, 855 n.3 (2d Cir. 1982) oraz wyroku *Hubco Data Prods. Corp. v. Management Assistance, Inc.*, 219 U.S.P.Q. 450 (D. Idaho 1983).

Ostatecznie sygnalizowane wątpliwości zostały rozstrzygnięte w wyroku w sprawie *Apple Computer, Inc. v. Franklin Computer*¹⁰. W stanie faktycznym, będącym przedmiotem oceny sądu, pozwany (Franklin) wytwarzał komputer kompatybilny z komputerem Apple II, pozwalający na uruchomienie programów aplikacyjnych i systemu operacyjnego Apple. Oprogramowanie dołączane do komputera przez Franklin Computer było w istotnej części skopiowane z komputera Apple i zapisane w pamięci ROM komputera Franklin w postaci kodu wykonywalnego. Różnice między programami Apple i Franklin były minimalne i dotyczyły np. usunięcia noty copyrightowej Apple. Główna linia obrony Franklina zmierzała do wykazania, że system operacyjny Apple nie podlega ochronie, gdyż dla zachowania pełnej kompatybilności komputerów niezbędne jest jego odtworzenie w identycznej lub niemal identycznej postaci. W trakcie postępowania podniesiono również wątpliwości co do zdolności ochronnej programu komputerowego zapisanego w formie kodu wynikowego oraz w pamięci ROM. Sąd jednoznacznie stwierdził, że ochrony prawnoautorskiej nie wyklucza: zapisanie programu w formie kodu wynikowego, zapisanie programu w pamięci ROM oraz kwestia, czy program stanowi system operacyjny komputera. Podobnego rozstrzygnięcia doczekała się również sprawa *Apple Computer Inc. v. Formula International, Inc.*¹¹, gdzie sąd wyraźnie wskazał, że ochroną prawnoautorską mogą być objęte wszystkie programy komputerowe utrwalone na dowolnym medium, niezależnie od swojego przeznaczenia lub funkcji. Podkreślił też, że dla ochrony prawnoautorskiej nie ma znaczenia możliwość bezpośredniego zapoznania się przez użytkownika z formą wyrażenia programu¹².

Najpóźniej w orzecznictwie rozstrzygnięto kwestie prawnoautorskiej ochrony mikro kodu (*microcode*)¹³. Podstawowe znaczenie miała tu szeroko komentowana sprawa *NEC Corp. v. Intel Corp.*¹⁴, w której NEC zmierzał do ustalenia, że mikro kod stosowany w procesorach Intela nie podlega ochronie prawnoautorskiej. W trakcie sporu przeciw ochronie prawnoautorskiej mikro kodu podniesiono m.in., że pełni on wyłącznie funkcję użytkową (jest raczej częścią maszyny, nie zaś utworem w rozumieniu prawa autorskiego), jest trwale połączony z komputerem ze względu na jego zainstalowanie w pamięci ROM, a ponadto nie może podlegać ochronie z uwagi na niewielki stopień oryginalności¹⁵. Sąd ostatecznie odrzucił argumentację przemawiającą przeciw prawnoautorskiej ochronie mikro kodu i uznał, że wskazane okoliczności nie wyłączają jego

¹⁰ *Apple Computer, Inc. v. Franklin Computer Corp.*, 714 F.2d 1240, 219 U.S.P.Q. (BNA) 113 (3d Cir. 1983); 464 U.S. 1033 (1984).

¹¹ *Apple Computer Inc. v. Formula International, Inc.*, 562 F.Supp. 775 (C.D. Cal. 1983); 725 F.2d 521 (9th Cir. 1984).

¹² J. Drexler, *What is Protected...*, s. 17.

¹³ Zadaniem mikro kodu jest bezpośrednie sterowanie pracą procesora komputera. Jego działanie polega na „tłumaczeniu” listy rozkazów procesora na wewnętrzne polecenia znajdujące odzwierciedlenie w fizycznej budowie układu półprzewodnikowego. Mikro kod działa na najniższym poziomie integracji ze sprzętem komputerowym.

¹⁴ *NEC Corp. v. Intel Corp.*, 10 U.S.P.Q. 2d (BNA) 1177 (N.D. Cal. 1989).

¹⁵ Argumenty NEC i historię sporu po wyroku sądu pierwszej instancji najpełniej prezentuje R.C. Hinckley, *NEC v. Intel: Will Hardware Be Drawn into the Black Hole of Copyright*, HTLJ 1987/3, s. 23 i n.

ochrony na gruncie *Copyright Act*¹⁶. Prawnoautorska ochrona mikro kodu została też potwierdzona w wyroku w sprawie *Allen Myland, Inc. v. IBM*¹⁷, w którym sąd uzasadnił prawnoautorską ochronę programu komputerowego dostatecznym zakresem swobody twórczej, jaką w tej sprawie dysponował programista.

Literalne przejście treści cudzego programu nie musi jeszcze implikować naruszenia. W sprawie *Sega Enterprises Ltd. v. Accolade, Inc.*¹⁸ spór dotyczył konsoli do gier Sega Genesis III. Konsola zawierała zabezpieczenie techniczne typu *lock-and-key*, które pozwalało uruchomić na niej tylko gry zapisane na kartridżach licencjonowanych producentów gier. Zasadnicza część sporu dotyczyła dekompilacji oprogramowania zabezpieczającego i znaczenie tego wyroku wynika głównie z uznania tej praktyki za dopuszczalną w ramach *fair use*. Sąd uznał jednak również, że do naruszenia autorskich praw majątkowych nie doprowadziło skopiowanie przez pozwanego krótkiego, 20-bajowego fragmentu programu powoda, którego odtworzenie w dokładnie takim kształcie było niezbędne dla obejścia zabezpieczeń konsoli. Sąd podkreślił, że programy komputerowe z reguły mają użytkowy charakter, tj. służą realizacji określonych zadań. Jako takie zawierają wiele elementów logicznych, strukturalnych i wizualnych, które są podyktowane funkcją do wykonania, względami wydajności lub czynnikami zewnętrznymi, takimi jak wymagania kompatybilności czy potrzeby biznesowe. Dlatego w niektórych okolicznościach nawet konkretny zestaw poleceń użytych przez programistę, konkretny fragment kodu może nie podlegać ochronie z uwagi na jego funkcjonalny charakter¹⁹. Zgodnie z zacytowanym przez sąd fragmentem raportu CONTU sytuacja taka wystąpi, jeżeli dany zestaw instrukcji jest „jedynym i podstawowym środkiem realizacji danego zadania”. Wówczas „jego późniejsze użycie przez inną osobę nie będzie równoznaczne z naruszeniem”²⁰. Sąd wskazał ponadto, że minimalna długość przejętego fragmentu wskazuje, iż pozostaje on poza zakresem ochrony zgodnie z przyjmowaną przez *Copyright Office* doktryną krótkich fraz (*short phrases doctrine*)²¹, na mocy której poza zakresem ochrony prawnoautorskiej pozostają słowa i krótkie frazy, takie jak nazwy, tytuły i slogany.

Podobnie w sprawie *Lexmark International, Inc. v. Static Control Components*²² sąd uznał, że program komputerowy osadzony w kartridżach do drukarek Lexmark, który

¹⁶ Sąd jednocześnie uznał, że dekompilacja mikro kodu w celu osiągnięcia kompatybilności sprzętowej nie prowadzi do naruszenia prawa autorskiego, podobnie jak niezależne stworzenie zbliżonego mikro kodu w ramach techniki określanej mianem *clean room* (przez programistów niemających styczności z oryginalnym kodem Intela). Zob. J. Contreras, L. Handley, T. Yang, *NEC v. Intel: Breaking new ground in the Law of Copyright*, Harv. JL & Tech. 1990/3, s. 209 i n. Zob. A. Nowicka, *Prawnoautorska...*, s. 26–27.

¹⁷ *Allen-Myland, Inc. v. IBM Corp.*, 746 F. Supp. 520 (E.D. Pa. 1990).

¹⁸ *Sega Enterprises Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992).

¹⁹ *Sega Enterprises Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992), pkt 54.

²⁰ *Final Report...*, s. 20.

²¹ Zob. tytuł 37 § 202.1 (a) *Code of Federal Regulations*, który wskazuje, że poza zakresem ochrony prawnoautorskiej pozostają „słowa i krótkie wyrażenia, takie jak nazwy, tytuły i slogany”.

²² *Lexmark Int'l, Inc. v. Static Control Components*, 387 F.3d 522 (6th Cir. 2004).

jego konkurenci musieli zainstalować we własnych kartridżach, aby umożliwić współdziałanie z drukarkami Lexmark, nie podlega ochronie prawnoautorskiej. Sąd zastosował tu doktrynę *merger*, uznając, że w tym przypadku idea lub funkcja programu i jego forma wyrażenia „stapiają” się w stopniu wyłączającym ochronę prawnoautorską²³.

3. Ochrona przed nieliteralnym kopiowaniem

3.1. Struktura, sekwencja i organizacja programu

Ograniczenie ochrony prawnoautorskiej tylko do przypadków dosłownego, literalnego kopiowania spowodowałoby, że wystarczyłyby proste zmiany językowe, aby uniknąć naruszenia praw autorskich. Stąd nie ulega wątpliwości, że efektywna ochrona prawnoautorska musi obejmować również przypadki „nieliteralnego” kopiowania. Również w przypadku programów komputerowych ochrona prawnoautorska wykracza poza konkretną postać kodu. Pojawia się tu jednak pytanie, jak daleko ochrona ta powinna sięgać i w jaki sposób dokonać rozróżnienia między niechronioną ideą (elementami koncepcyjnymi) a chronioną formą wyrażenia (ekspresją) programu.

Problem ten zarysował się wyraźnie w drugim etapie rozwoju orzecznictwa sądów amerykańskich. Jego początki można datować na połowę lat 80. XX w.²⁴ Wtedy to, w orzeczeniu *SAS Inst. v. S&H Computer Systems*²⁵ sąd uznał, że do naruszenia praw autorskich prowadzi nie tylko przejęcie konkretnych linii kodu cudzego programu, ale również „organizacji i detali struktury”²⁶. Podobnie w orzeczeniu *E.F. Johnson v. Uniden*²⁷ sąd dopatrył się naruszenia, mimo braku literalnej zbieżności poszczególnych linii kodu porównywanych programów przeznaczonych dla sprzętu wyposażonego w inne typy procesorów. W ocenie sądu przejęcia elementów cudzego programu nie uzasadniała konieczność zachowania kompatybilności programów, zostały zresztą przejęte elementy, które nie były decydujące dla zapewnienia komunikacji między programem powoda i pozwanego²⁸.

²³ Zob. P. Samuelson, *Functionality and Expression in Computer Programs: Refining the Tests for Software Copyright Infringement*, BTLJ 2016/31, s. 1221.

²⁴ Z okresu obowiązywania *Copyright Act* z 1909 r. znane jest tylko jedno orzeczenie dotyczące omawianej problematyki – wyrok *Synercom Technology, Inc. v. University Computing Co.*, 462 F. Supp. 1003, 199 U.S.P.Q. (BNA) 537 (N.D. Tex. 1978). Sąd za niechronioną ideę uznał tu sekwencje poleceń wprowadzanych przez operatora programu i sposób zapisu danych na kartach perforowanych.

²⁵ *SAS Inst., Inc. v. S&H Computer Systems, Inc.*, 605 F.Supp. 816 (M.D. Tenn. 1985).

²⁶ Zob. szerzej M. Radcliffe, *Recent US Developments in Copyright Law Related to Computer Software*, EIPR 1986/2, s. 43.

²⁷ *E.F. Johnson Co. v. Uniden Corp. of Am.*, 623 F.Supp. 1485 (D.C. Minn. 1985).

²⁸ Zob. A.L. Clapes, P. Lynch, M. Steinberg, *Determining the proper scope of copyright protection for computer programs*, „UCLA Law Review” 1987/34, s. 1561, 1577.

Przełomem okazało się jednak dopiero głośne orzeczenie w sprawie *Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc.*²⁹ Powód i pozwany początkowo współpracowali przy oferowaniu na rynku programu stworzonego przez powoda (Dentalab), przeznaczonego dla komputerów IBM Series/1. W pewnym momencie pozwany wprowadził na rynek produkt stworzony przez powiązany podmiot (Dentcom), przeznaczony dla mniejszych komputerów PC (Dentlab). Kod źródłowy obu programów został napisany w różnych językach programowania – program Whelan był napisany w języku EDL, program Dentlab w BASIC. Ze względu na różnice między tymi językami programowania nie ulegało wątpliwości, że kod źródłowy programu Dentcom nie został literalnie przetłumaczony z EDL na BASIC, chociaż wiele wskazywało na to, że mógł być wykorzystany przy tworzeniu programu w BASIC. Między programem Whelan i Dentcom zachodziło natomiast daleko idące podobieństwo w zakresie struktury plików danych i przedstawień ekranowych. Ponadto pięć podprogramów było niemal identycznych, istniało też ogólne podobieństwo w zakresie struktury wewnętrznej.

Sąd wyszedł z założenia, że ochrona programów komputerowych, podobnie jak innych utworów literackich, nie ogranicza się do ich literalnej postaci. Sformułował więc prosty test, pozwalający na oddzielenie idei i ekspresji. Wskazał mianowicie, że za ideę należy uznać przeznaczenie lub ogólną funkcję utworu o charakterze użytkowym, natomiast jej ekspresją jest wszystko, co nie jest niezbędne do urzeczywistnienia takiej funkcji lub przeznaczenia. Dlatego w sytuacji, w której istnieją różne sposoby urzeczywistnienia tak rozumianej idei, sposób obrany przez twórcę stanowi chronioną ekspresję. W przypadku programu Dentalab ideą jest podstawowa funkcja realizowana przez program, tj. zarządzanie laboratorium dentystycznym. Zachowanie struktury programu Dentalab nie jest konieczne dla urzeczywistnienia tak rozumianej idei, inne programy oferowane na rynku odznaczają się bowiem inną strukturą, mimo że realizują te same funkcje. W konsekwencji struktura programu Dentalab winna być uznana za chronioną formę wyrażenia, a więc jej przejście do programu Dentlab stanowi naruszenie prawa autorskiego.

Wyrok w sprawie Whelan oznaczał akceptację tezy, że prawnoautorska ochrona programu komputerowego rozciąga się poza literalną postać kodu źródłowego lub wynikowego i obejmuje również „strukturę, sekwencję i organizację” wewnętrzną programu³⁰. Należy podkreślić, że koncepcja ta, choć w omawianej sprawie po raz pierwszy zastosowana do programów komputerowych, ma swoją genezę gdzie indziej. W odniesieniu do utworów beletrystycznych i dramatycznych ochrona rozciągana jest przecież na konkretną fabułę, sposób zawiązania akcji, sekwencję scen, charakterystykę postaci czy schemat dialogów. Z kolei w przypadku utworów naukowych może ona objąć szczegółowy układ treści, a także dobór, zestawienie lub układ faktów, argumentów lub rozumowań. Wreszcie oryginalność w zakresie doboru, układu lub wzajemnego zestawienia elementów uza-

²⁹ *Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc.*, 797 F.2d 1222, 230 U.S.P.Q. (BNA) 481 (3d Cir. 1986).

³⁰ J. Velasco, *The Copyrightability...*, s. 261.

sadnia ochronę w ramach prawa autorskiego różnego rodzaju zbiorów. Dlatego też nie można *a priori* przyjąć, że struktura, sekwencja i organizacja programu komputerowego z zasady należy do sfery idei, pozostając tym samym poza zakresem ochrony.

Problemem jest natomiast – na co trafnie zwraca się uwagę w literaturze przedmiotu – sposób, w jaki sąd w sprawie *Whelan* wytyczył granicę ochrony. Posłużył się bowiem nadmiernym uproszczeniem, utożsamiając ideę programu wyłącznie z jego ogólną funkcją i przeznaczeniem, a następnie ograniczył się do badania, czy istnieje inny sposób realizacji tak rozumianej idei³¹. Nie uwzględnił to kwestii, że w przypadku technologii informatycznych pewne sposoby realizacji idei mogą być lepsze od innych. Uznanie pewnych sposobów za „lepsze” może wynikać chociażby z dążenia do zapewnienia efektywności działania programu, a więc szybkości realizacji obliczeń, zajętości pamięci czy kompatybilności. Korzystniejszym podejściem jest więc ograniczenie ochrony programów komputerowych do ekspresyjnych elementów struktury, sekwencji i organizacji wewnętrznej programów, czyli takich, które nie rozciągają się na funkcjonalne atrybuty programu³².

Podejście przyjęte w sprawie *Whelan* ma też istotny aspekt konkurencyjny. Zbyt szeroko określone granice ochrony wymuszają poszukiwanie „na siłę” rozwiązania odmiennego i często gorszego niż zastosowane w programie wcześniejszym³³. Potwierdza to wydany niedługo później wyrok w sprawie *Broderbund v. Unison World*³⁴, dotyczący interfejsu użytkownika programu *Print Shop*. Sąd, odwołując się do sprawy *Whelan*, wprost wskazał, że w celu uniknięcia naruszenia prawa autorskiego pozwany powinien posługiwać się innymi opisami poszczególnych elementów interfejsu mimo ich zasadniczo opisowego charakteru.

Test abstrakcji wypracowany w sprawie *Whelan* nigdy nie doczekał się pełnej akceptacji w orzecznictwie. Przykładem może być sprawa *Plains Cotton Co-op v. Goodpasture Computer Serv., Inc.*³⁵ Spór dotyczył oprogramowania umożliwiającego użytkownikom otrzymywanie regularnie aktualizowanych informacji dotyczących cen i dostępności bawełny, a także księgowości i realizacji zamówień. Sąd odrzucił możliwość zastosowania rozumowania leżącego u podstaw sprawy *Whelan* z uwagi na to, że wiele podobieństw między porównywanymi programami było podyktowanych zewnętrznymi uwarun-

³¹ Zob. S. Englund, *Idea, Process, or Protected Expression? Determining the Scope of Copyright Protection of the Structure of Computer Programs*, Mich. L. Rev. 1990/88, s. 896; P.S. Menell, *An Analysis of the Scope of Copyright Protection for Applications Programs*, Stan. L. Rev. 1989/41, s. 1084; J. Velasco, *The Copyrightability...*, s. 261.

³² P.S. Menell, *An Analysis...*, s. 1085.

³³ P. Spivak, *Does Form Follow Function? The Idea/Expression Dichotomy in Copyright Protection of Computer Software*, „UCLA Law Review” 1988/35, s. 747.

³⁴ *Broderbund Software, Inc. v. Unison World, Inc.*, 648 F. Supp. 1127, 231 U.S.P.Q. (BNA) 700 (N.D. Cal. 1986).

³⁵ *Plains Cotton Co-op v. Goodpasture Computer Serv., Inc.*, 807 F.2d 1256, 1262 (5th Cir. 1987).

kowaniami związanymi ze specyfiką rynku bawełny. Na przykład jeden z elementów programu został zaprojektowany w taki sposób, aby przedstawiać te same informacje, które są zawarte w „papierowym” arkuszu informacyjnym dotyczącym bawełny, z uwzględnieniem ograniczeń narzuconych przez użycie komputera. Ponieważ specyfika rynku bawełny odgrywała, zdaniem sądu, znaczącą rolę w określaniu sekwencji i organizacji porównywanych programów do sprzedaży bawełny, sąd uznał je za „pomysły” pozostające poza zakresem ochrony.

Poza wspomnianym wyrokiem w sprawie *Broderbund v. Unison World* test przyjęty w sprawie *Whelan* został jednak zastosowany w kilku wyrokach sądów³⁶. Kulminacją tej linii orzeczniczej był wyrok w sprawie *Pearl Sys. v. Competition Electronics*³⁷. W wyniku zastosowania testu abstrakcji, sformułowanego w sprawie *Whelan*, doszło tu do rozciągnięcia ochrony prawnoautorskiej na funkcje urządzenia realizowane przez program komputerowy. Sąd doszedł bowiem do wniosku, że jeżeli programy sterujące pracą urządzeń przeznaczonych do tego samego celu (pomiaru czasu na zawodach sportowych) reagują tak samo na takie same działania użytkownika (naciśnięcie przycisku uruchamiającego pomiar czasu), to zachodzi między nimi istotne podobieństwo wskazujące, że program późniejszy został skopiowany. Tymczasem o kopiowaniu można mówić tylko wówczas, gdy do późniejszego utworu zostaną przejęte elementy podlegające ochronie, nie zaś – jak w tym przypadku – elementy o charakterze funkcjonalnym³⁸.

3.2. Test *abstraction-filtration-comparison*

Odwrot od poszerzania przedmiotowych granic ochrony nastąpił w pierwszej połowie lat 90. XX w. Szczególne znaczenie miało tu szeroko komentowane i nadal aktualne orzeczenie sądu apelacyjnego 2 okręgu w sprawie *Computer Associates International Inc. v. Altai*³⁹. Stan faktyczny, będący podstawą rozstrzygnięcia, był złożony. Powód (*Computer Associates International Inc.*) oferował na rynku program *CA-Scheduler*, służący do kontroli i uruchamiania w określonym czasie zadań w systemach typu *main-frame*, wytwarzanych przez IBM. Cześć *CA-Scheduler* był moduł o nazwie *Adapter*, zapewniający kompatybilność programu głównego z wieloma systemami operacyjnymi. Pozwany (*Altai*) oferował analogiczny program o nazwie *Zeke*, wykonujący te same funkcje co *CA-Scheduler*, ale przeznaczony tylko dla jednego systemu operacyjnego. W odpowiedzi na zapotrzebowanie klientów pozwany postanowił wyposażyć *Zeke* w moduł umożliwiający jego współpracę z różnymi systemami operacyjnymi. Zatrudniony w tym celu programista, który poprzednio pracował u powoda przy tworzeniu

³⁶ Zob. *Bull HN Info. Sys., Inc. v. American Express Bank, Ltd.*, 1990 Copyright Law Dec. (CCH) P 26,555 at 23,278, 1990 WL 48098 (S.D.N.Y.1990); *Dynamic Solutions Inc. v. Planning & Control Inc.* 1987 Copyright Law Dec (CCH) P 26,061 (SDNY 1987).

³⁷ *Pearl Sys., Inc. v. Competition Electronics Inc.*, 8 U.S.P.Q.2d 1520 (S.D. Fla. 1988).

³⁸ Zob. szerzej A. Nowicka, *Prawnoautorska...*, s. 31; M.D. Scott, *Information...*, § 2.51 [E.1].

³⁹ *Computer Associates International Inc. v. Altai* 23 U.S.P.Q. 2d 1241 (2d Cir. 1992).

modułu Adapter, dysponując niezgodnie z prawem dostępem do kodu źródłowego, skopiował około 30% kodu tego modułu. W ten sposób powstał program Oscar 3.4. Po odkryciu tego faktu pozwany, wypuszczając na rynek kolejną wersję programu Oscar, napisał ją na nowo. Co istotne Oscar 3.5 powstał w tzw. *clean room*, tzn. został napisany przez programistów, którzy nie mieli dostępu ani do kodu programu Adapter, ani do kodu programu Oscar 3.4. Mimo że kod programu Oscar 3.5 ani bezpośrednio, ani pośrednio nie został skopiowany z programu Adapter, powód dopatrywał się podobieństw w zakresie struktury obu programów.

Przystępując do wytyczenia przedmiotowego zakresu ochrony, sąd przypominał, że również w odniesieniu do programów komputerowych zachowuje aktualność wyrażona w § 102 (b) *Copyright Act* podstawowa dla prawa autorskiego zasada, zgodnie z którą ochronie nie podlega idea zawarta w utworze, a sposób jej wyrażenia (ekspresja). Ze względu jednak na zasadniczo użytkowy charakter programów komputerowych odróżnienie w ich przypadku idei i ekspresji jest utrudnione. W dziełach o charakterze estetycznym wybierane przez twórcę środki wyrazu nie są ukierunkowane na praktyczne zastosowanie. Natomiast programy komputerowe odzwierciedlają w swej treści zarówno procesy obliczeniowe, jak i abstrakcyjne idee, łącząc w sobie „ekspresję twórczą i techniczną”. W konsekwencji, w porównaniu z dziełami o charakterze estetycznym, sytuują się znacznie bliżej granicznej linii wyznaczanej przez § 102 (b) *Copyright Act*.

Następnie sąd podważył główną tezę wyroku w sprawie Whelan. Podzielając pogląd, że również nieliteralne elementy programu komputerowego mogą podlegać ochronie w ramach prawa autorskiego, podkreślił zarazem, że przyjęta w niej koncepcja ochrony struktury, sekwencji i organizacji programu jest wynikiem „błédnego zrozumienia sposobu działania programu komputerowego”. U podstaw programu nie leży bowiem tylko jedna idea, sprowadzająca się do jego ogólnego przeznaczenia (funkcji) programu. Ostateczna funkcja lub przeznaczenie programu komputerowego jest natomiast złożonym wynikiem oddziaływania składających się nań poszczególnych podprogramów. Ponieważ każdy z podprogramów jest sam w sobie programem, ma swą własną ideę. W konsekwencji teza leżąca u podstaw wyroku Whelan, że ogólne przeznaczenie programu jest tożsame z ideą programu, jest „opisowo nieadekwatne”.

Wreszcie sąd apelacyjny 3 okręgu potwierdził zasadność dokonanego w wyroku sądu pierwszej instancji⁴⁰ rozróżnienia sekwencji czynności wykonywanych przez program (jak program działa, jego „struktury dynamicznej”) od konkretnego uporządkowania instrukcji w kodzie programu (jak program jest napisany, „struktury statycznej” programu). Pierwszy obszar jest wyłączony spod ochrony prawnoautorskiej zgodnie z § 102 (b) *Copyright Act* jako proces, system, metoda działania. Jedynie drugi obszar może podlegać ochronie w takim zakresie, w jakim stanowi chronioną formę wyrażenia.

⁴⁰ Computer Associates International Inc. v. Altai Inc., 775 F.Supp. 544, 20 USPQ2d 1641 (E.D.N.Y. 1991).

W celu oddzielenia niechronionej idei od chronionej formy wyrażenia sąd sformułował trójetapowy test, określany mianem testu poziomów abstrakcji (*levels of abstraction test*) lub testem AFC (*abstraction-filtration-comparison test*). W pierwszym etapie (*abstraction*) należy dokonać rozdzielenia poszczególnych poziomów abstrakcji dla poszczególnych modułów programu. Poziomy abstrakcji powinny być przy tym odpowiednikami kroków podjętych przez programistę dla stworzenia programu, a więc na najniższym poziomie abstrakcji program jest zbiorem instrukcji w konkretnym języku programowania, na najwyższym zaś pojmowany jest poprzez funkcje, jakie pełnią poszczególne jego moduły. Sąd wyróżnił następujące etapy tworzenia programu: określenie podstawowego przeznaczenia (funkcji) programu, podział tego celu na podzadania realizowane przez poszczególne podprogramy lub moduły, określenie operacji logicznych wykonywanych w poszczególnych modułach i ich wzajemnych powiązań w postaci diagramów przepływu (*flow chart*), zakodowanie programu w konkretnym języku programowania. Odpowiednio do tego procesu wyodrębnił następujące poziomy abstrakcji: ogólny cel, listy usług systemu operacyjnego, listy parametrów programu, kod źródłowy, kod wynikowy⁴¹.

W drugim etapie (*filtration*) dla każdego poziomu abstrakcji należy oddzielić ekspresję od idei, posługując się metodami wypracowanymi w prawie autorskim, przede wszystkim *merger doctrine* i *scenes à faire doctrine*. Interpretacja *merger doctrine* przyjęta w wyroku polega na uwzględnieniu efektywności działania programu. Sąd zauważył, że programiści generalnie dążą do tworzenia możliwie najefektywniejszych programów. Koncepcja efektywności jest tu zbliżona do uzyskania najbardziej zwięzłego dowodu logicznego lub sformułowania najbardziej zwięzłego obliczenia matematycznego. Dlatego im wydajniejszy jest dany element programu, w tym większym stopniu stanowi on czyste odzwierciedlenie idei lub procesu zawartego w tym konkretnym elemencie. O ile hipotetycznie może istnieć niezliczona liczba sposobów, na które programista może realizować pewne funkcje w ramach programu, tzn. wyrażać ideę ucieleśnioną w danym podprogramie, o tyle potrzeba zapewnienia wydajności może zawęzić rzeczywiście istniejący zakres wyboru do zaledwie jednej lub dwóch form ekspresji. Z kolei *scenes à faire doctrine* pozwala oddzielić elementy, których forma wyrażenia podyktowana jest czynnikami zewnętrznymi, takimi jak: techniczna specyfika komputera, w którym funkcjonować ma program, konieczność zapewnienia kompatybilności z innymi programami, standardy projektowania w przemyśle informatycznym czy szeroko stosowane rozwiązania programistyczne. Wreszcie nie podlegają również ochronie elementy zaczerpnięte z domeny publicznej jako nieodznaczające się oryginalnością (niepochodzące od twórcy).

⁴¹ Zob. A. Nowicka, *Prawnoautorska...*, s. 44. Należy pamiętać, że poziomy abstrakcji w sprawie Altai zostały sformułowane dla potrzeb oceny bardzo specyficznych programów, będących „pośrednikami” między systemem operacyjnym a programem aplikacyjnym. W sposób bardziej uniwersalny poziomy abstrakcji sformułowano w omawianym niżej orzeczeniu w sprawie Gates Rubber Co. v. Bando Chemical Industries, Ltd., 9 F.3d 823, 28 U.S.P.Q. 2d 1503 (10th Cir. 1993).

Na ostatnim etapie (*comparison*) należy porównać to, co pozostało po odrzuceniu elementów pozostających poza zakresem ochrony. Jeżeli wszystkie podobieństwa w ekspresji wynikają z wykorzystania wspólnych idei porównywanych programów, to nie zachodzi istotne podobieństwo. Nie można również mówić o istotnym podobieństwie w sytuacji, w której zostanie przejęta nieistotna część programu wcześniejszego.

Zastosowanie testu AFC doprowadziło sąd do wniosku, że nie tylko funkcje realizowane przez poszczególne programy należy pozostawić poza zakresem ochrony. Chronione prawem autorskim nie mogą być również listy usług, ponieważ ich kształt wymusza system operacyjny. Tak samo podyktowane czynnikami zewnętrznymi są listy parametrów. Z kolei na poziomie literalnym między kodem źródłowym porównywanych programów nie było w zasadzie jakiegokolwiek podobieństwa. W konsekwencji w zakresie elementów mogących podlegać ochronie tylko kilka tzw. makropoleceń (makr) w programie Oscar 3.5 wykazywało istotne podobieństwo do programu Adapter. Jednak ich wpływ na całość programu nie był znaczący. W świetle tej analizy sąd podtrzymał wyrok sądu pierwszej instancji, stwierdzając, że program Oscar 3.5 nie stanowił naruszenia praw autorskich do programu Adapter.

Krytyka orzeczenia Altai koncentrowała się przede wszystkim wokół sposobu, w jaki zastosowano *merger doctrine* oraz wskazanych przez sąd elementów zdeterminowanych wymogami zewnętrznymi, podlegających odfiltrowaniu w oparciu o *scenes à faire doctrine*. W komentarzach można spotkać opinię, że wyłączenie z zakresu ochrony prawnoautorskiej elementów podyktowanych koniecznością zapewnienia efektywności działania programu miałoby stanowić zachętę dla konkurentów do naśladowania najlepiej sprzedających się programów. Ponadto, zdaniem części autorów, na gruncie amerykańskiego prawa autorskiego brak podstawy normatywnej dla wyłączenia spod ochrony tych elementów, które odpowiedzialne są za zapewnienie interoperacyjności (kompatybilności) programu⁴². Większość autorów odrzuca jednak te zastrzeżenia. Wskazuje się, że zastosowanie *merger doctrine* do wyłączenia spod ochrony elementów optymalnych z punktu widzenia efektywności programu nie oznacza zezwolenia na kopiowanie fragmentów programu w oparciu o subiektywną ocenę konkurenta. Natomiast „odfiltrowanie” elementów odpowiedzialnych za interoperacyjność programu jest prostą konsekwencją zastosowania doktryny *scenes à faire* w kontekście programów komputerowych. W odróżnieniu od „zwyczajnych” utworów literackich, w przypadku których swobodę środków wyrazu ogranicza wyłącznie wyobraźnia autora, programy komputerowe muszą być pisane w logicznej formie, która może być wykonana przez

⁴² Tak W. Effross, *Assaying Computer Associates v. Altai: How Will the „Golden Nugget” Test Pan Out?*, RC&TLJ 1993/19, s. 52 i n.; J. Butler, *Pragmatism in Software Copyright: Computer Associates v. Altai*, Harv. JL & Tech. 1992/6, s. 186–187.

komputer. Ogranicza to zakres możliwych form wyrażenia tak w odniesieniu do literalnej treści programu, jak i jego struktury⁴³.

Test AFC, sformułowany w sprawie Altai, doczekał się rozwinięcia w orzeczeniu *Gates Rubber v. Bando Chemical Industries*. W pierwszym etapie testu sąd wyróżnił następujące „poziomy abstrakcji”: 1) główne przeznaczenie programu, 2) strukturę lub architekturę programu, 3) poszczególne moduły, 4) algorytmy i struktury danych, 5) kod źródłowy, 6) kod wykonywalny (maszynowy). W drugim wskazał na konieczność „odfiltrowania” sześciu rodzajów elementów wyłączonych spod ochrony: idei, procedur, faktów, *merger*, *scenes à faire* i elementów należących do *public domain*. Główne przeznaczenie lub funkcja programu, podobnie jak poszczególnych modułów, zawsze będą pozostawać poza zakresem ochrony. Natomiast kod źródłowy i kod maszynowy będą z reguły stanowić chronione wyrażenie, jeżeli tylko odznaczają się oryginalnością. Na pośrednich poziomach abstrakcji nie można natomiast przesądzić o braku ochrony poszczególnych elementów. W przypadku jednego programu struktura lub organizacja wewnętrzna może bowiem zostać „odfiltrowana”, a w konsekwencji pozostawać poza zakresem ochrony, natomiast w przypadku innego stanowić chronioną formę wyrażenia⁴⁴.

Mimo ostrej krytyki ze strony dużych producentów oprogramowania⁴⁵ linia wytyczona przez orzeczenia Altai i Gates Rubber szybko zdominowała orzecznictwo⁴⁶. Wynika to przede wszystkim z przejrzystości wypracowanego testu i łatwości jego stosowania do wewnętrznych elementów oprogramowania. Tym można tłumaczyć odnotowywane w literaturze zastosowanie zbliżonych testów również w orzecznictwie sądów innych państw⁴⁷. Wątpliwa jest jednak, wbrew stanowisku niektórych autorów, jego recepcja w Europie. We francuskim wyroku w sprawie *Computer Associates v. S.A.R.L. Faster*⁴⁸,

⁴³ Zob. J. Velasco, *The Copyrightability...*, s. 273; D.S. Karjala, *Recent US and International Developments in Software Protection*, part 2, EIPR 1994/2, s. 60, A. Miller, *Copyright Protection for Computer Programs, Databases and Computer-Generated Works: is Anything new Since CONTU?*, „Harvard Law Review” 1993/106, s. 1001.

⁴⁴ E. Derclaye, *Software Copyright Protection: Can Europe Learn from American Case Law*, part 2, EIPR 2000/2, s. 59.

⁴⁵ Zob. wypowiedzi cytowane przez M.A. Lemley, *Convergence in the Law of Cotware Copyright?*, BTLJ 1995/1, s. 14.

⁴⁶ Zob. m.in. *Kepner-Tregoe, Inc. v. Leadership Software, Inc.*, 12 F.3d 527 (5th Cir. 1994); *Harbor Software, Inc. v. Applied Systems, Inc.*, 887 F. Supp. 86 (S.D.N.Y. 1995), 925 F.Supp. 1042 (S.D.N.Y. 1996), 936 F.Supp. 167 (S.D.N.Y. 1996); *Control Data Systems, Inc. v. Infoware, Inc.*, 903 F. Supp. 1316 (D. Minnesota 1995). Z kolei w orzeczeniu *Autoskill, Inc. v. National Educ. Support Sys., Inc.* 994 F.2d 1476, 1495 (10th Cir. 1993), sąd pierwotnie dopatrzył się naruszenia w przejęciu systemu nauki czytania. Wyrok został jednak częściowo uchylony w oparciu o test sformułowany w sprawie Altai. Zob. D.S. Karjala, *Recent...*, s. 60.

⁴⁷ Test *abstraction-filtration-comparison* został zastosowany w orzeczeniach sądów kanadyjskich *Carolian Systems International Inc. v. Triolet Systems Inc.*, (1989), 25 C.P.R. (3d) 87 oraz *Matrox Elect. Sys. v. Gaudreau*, 1993 R.J.Q. 2449. Zob. S. Handa, J. Buchan, *Copyright as It Applies to the Protection of Computer Programs in Canada*, IIC 1995/1, s. 48 i n.

⁴⁸ Wyrok Tribunal de Commerce w Bobigny z 20.01.1995 r., *Computer Associates v. S.A.R.L. Faster*, RIDA 1995/4(166), s. 325.

podawanym w literaturze jako dowód zastosowania wypracowanego w sprawie *Computer Associates v. Altai*⁴⁹ testu AFC przez sąd europejski⁵⁰, nawiązanie do testu AFC było związane z problemem, w jakim stopniu powołany wyrok sądu amerykańskiego powoduje, że powództwo przeciw francuskiemu dystrybutorowi Altai stanowi *res iudicata*. Sąd ostatecznie nie dopatrzył się naruszenia, jednak podstawą był brak oryginalności przejętych przez Altai elementów⁵¹. Z kolei w sprawie *Richardson v. Flanders*⁵² EWHC raczej próbował zastosować niż zastosował test AFC. Odmienność tradycji prawnych i niedostosowanie testu do okoliczności rozpatrywanej sprawy doprowadziło do rozciągnięcia ochrony prawnoautorskiej na cechy programu, którym bliżej do niechronionych metod działania, niż chronionej formy wyrażenia⁵³.

4. Ochrona interfejsów użytkownika

4.1. Interfejs użytkownika jako pozatekstowy element programu komputerowego

Powszechnie stosowanym sposobem prezentacji informacji i interakcji programu komputerowego z użytkownikiem jest wyświetlanie na ekranie monitora określonych obrazów, symboli czy tekstu, składających się na tzw. interfejs użytkownika (*user interface*, UI). Problem ochrony interfejsów użytkownika programów użytkowych (aplikacyjnych) pojawił się w połowie lat 80. XX w. Na bardzo konkurencyjnym wówczas rynku istotnego znaczenia nabral „wygląd” i sposób obsługi programu. Upodobnienie własnego produktu do produktu lidera rynkowego ułatwiało bowiem pozyskanie nowych użytkowników, dla których zmiana programu nie musiała wówczas oznaczać zmiany przyzwyczajeń związanych z obsługą programu danego typu⁵⁴. Zrozumiała jest więc dająca się wówczas zaobserwować presja ze strony producentów oprogramowania na objęcie ochroną wyświetlanych na ekranie obrazów (słów, symboli), składających się na interfejs użytkownika. W pierwszej kolejności sądy musiały się tu zmierzyć z problemem, czy interfejs użytkownika może być uznany za część programu komputerowego, jego „nieliteralny” aspekt, którego skopiowanie prowadzi do naruszenia prawa do programu komputerowego.

⁴⁹ *Computer Associates International Inc. v. Altai* 23 U.S.P.Q. 2d 1241 (2d Cir. 1992).

⁵⁰ M.A. Lemley, *Convergence...*, s. 15.

⁵¹ Zob. E. Derclaye, *Software...*, part. 2, s. 64; S. Mota, *Computer Associates v. Altai – French Computer Software Copyright Action Not Barred By U.S. Decision*, „Journal of Technology Law & Policy” 1997/1.

⁵² *John Richardson Computers Ltd. v. Flanders*, [1993] F.S.R. 497.

⁵³ Zob. D.S. Karjala, *Recent...*, s. 63; S. Lai, *The copyright protection of computer software in the United Kingdom*, Oxford–Portland, Oregon 2000, s. 78.

⁵⁴ Zob. T. Daubert, *Copyright, potential markets, and the user interface: defining the scope of the limited monopoly*, L&CP 1992/55, s. 368.

Po raz pierwszy problem rozróżnienia między programem a generowanymi przez program obrazami pojawił się w orzecznictwie dotyczącym gier video. W sprawach *Stern Electronics v. Kaufman*⁵⁵ i *Atari v. North American Philips Consumer Elecs.*⁵⁶ sądy oparły się na założeniu, że mimo iż nie doszło do naruszenia prawa autorskiego do programu komputerowego, możliwe jest naruszenie prawa autorskiego przez skopiowanie obrazów gry⁵⁷. Podobnie w wyrokach w sprawie *M. Kramer Manufacturing v. Andrews*⁵⁸ oraz *Williams Electronics v. Artic International*⁵⁹ przedstawienia audiowizualne generowane przez grę zostały uznane za utwór audiowizualny. W tych sprawach obrazy wyświetlane na ekranie, w tym elementy sterujące pracą gry, uznano za odrębny przedmiot ochrony od programu komputerowego, który te obrazy wygenerował.

W odniesieniu do programów użytkowych podejście zostało zastosowane w sprawie *Digital Communications v. Softklone*⁶⁰. Sąd przyjął, że podobieństwo obrazów prezentowanych na ekranie prowadzi do naruszenia prawa autorskiego do programu komputerowego jedynie wówczas, gdy wiąże się z tym podobieństwo w zakresie kodu programu. W pozostałych przypadkach obrazy generowane przez program winny być traktowane jako odrębny utwór. Niezwykle liberalnie podszedł przy tym do ustalenia, które z elementów wyświetlanych na ekranie mają charakter estetyczny. Za chronioną formę wyrażenia uznano bowiem sposób pogrupowania elementów na ekranie oraz podświetlenie i wyróżnienie dwóch wybranych liter każdego z poleceń⁶¹.

Podejście takie nie zyskało akceptacji w późniejszym orzecznictwie sądów. Począwszy od omawianego niżej wyroku w sprawie *Broderbund v. Unison World* utrzymała się tendencja do postrzegania interfejsów użytkownika jako pozatekstowych elementów programu, a podobieństwa programów w zakresie interfejsów użytkownika jako przypadków nieliteralnego kopiowania. Niewątpliwie wpłynęły na to względy praktyczne. Interfejsy i ekrany programów komputerowych z przełomu lat 80. i 90. XX w. były bowiem – z dzisiejszej perspektywy – bardzo proste. Jednocześnie na dłuższą metę nie było możliwe utrzymanie bardzo niskiego progu oryginalności, przyjętego w sprawie *Digital Communications v. Softklone*. Koncepcja nieliteralnego kopiowania pozatekstowych aspektów programu pozwalała ominąć ten problem⁶².

⁵⁵ *Stern Electronics Inc. v. Kaufman*, 669 F.2d 852 (2d Cir. 1982).

⁵⁶ *Atari, Inc. v. North American Philips Consumer Elecs. Corp.*, 672 F.2d 607 (7th Cir. 1982).

⁵⁷ Zob. S. Lai, *The copyright...*, s. 67.

⁵⁸ *M. Kramer Manufacturing Co., Inc. v. Andrews*, 783 F.2d 421, 448 (4th Cir. 1986).

⁵⁹ *Williams Electronics, Inc. v. Artic International, Inc.*, 685 F.2d 870, 215 U.S.P.Q. 405 (3d Cir. 1982).

⁶⁰ *Digital Communications Assoc. v. Softklone Distributing Corp.*, 659 F. Supp. 449 (N.D. Ga. 1987).

⁶¹ Zob. A.S. Middleton, *Thousand Clones: The Scope of Copyright Protection in the Look and Feel of Computer Programs*, „Washington Law Review” 1988/63, s. 195; M. Dailey, *The „Look and Feel” of Copyrightable Expression*, EIPR 1987/9, s. 234.

⁶² D.S. Karjala, *Copyright Protection of Computer Program Structure*, Brook. L. Rev. 1998/64, s. 55.

4.2. Struktura, sekwencja i organizacja interfejsu użytkownika

Po raz pierwszy skopiowanie interfejsu użytkownika zostało uznane za nieliteralne kopiowanie programu komputerowego w sprawie *Broderbund v. Unison World*⁶³. Powód wywodził, że pozwany naruszył prawo do programu powoda o nazwie *Print Shop*, przeznaczonego dla komputerów Apple i służącego do drukowania kartek z życzeniami i plakatów. Naruszenie miało polegać na skopiowaniu wyglądu i sekwencji przedstawień audiowizualnych do programu pozwanego, przeznaczonego dla komputerów PC. Sąd przyjął, że ochrona prawnoautorska nie ogranicza się do literalnej treści kodu programu, ale obejmuje też jego strukturę, sekwencję i organizację, a więc również strukturę, sekwencję i organizację przedstawień ekranowych. Jednocześnie pozostając pod wyraźnym wpływem niewiele wcześniejszego wyroku w sprawie *Whelan*, sąd zidentyfikował tu jako niechronioną ideę programu wyłącznie jego ogólną funkcję i przeznaczenie. Zastosowanie testu abstrakcji ze sprawy *Whelan* doprowadziło sąd do wniosku, że owa struktura, sekwencja i organizacja obrazów wyświetlanych na ekranie będzie stanowić chronioną ekspresję, jeżeli istnieją inne sposoby realizacji tej funkcji lub przeznaczenia. W uzasadnieniu wyroku sąd wytknął pozwanemu, że podobieństwo interfejsów nie ma żadnego uzasadnienia technicznego lub praktycznego. Przykładowo w ocenie sądu zamiast używać na ekranie wyboru czcionki zwrotu „Choose Font”, identycznego ze zwrotem używanym w programie powoda, pozwany winien użyć zwrotu „Select a Font”, „Indicate a Typeface Preference” czy „Which Type Style Do You Prefer”. W rezultacie podobieństwo programu pozwanego w zakresie wskazanych elementów zostało uznane za równoznaczne z naruszeniem prawa autorskiego do programu powoda⁶⁴.

Również w sprawie *Lotus Development Corp. v. Paperback Software International*⁶⁵ sąd oparł rozstrzygnięcie na rozumowaniu podobnym do przyjętego w sprawie *Broderbund*. Tu zarzuty dotyczyły m.in. przejścia do programu pozwanego (*Paperback VP-Planner*) struktury menu oraz „struktury, sekwencji i organizacji” obrazów wyświetlanych na ekranie (*screen displays*) arkusza kalkulacyjnego powoda (*Lotus 1-2-3*). Niesporne było, że kod programu pozwanego został stworzony niezależnie. Natomiast interfejs obu programów był niemalże identyczny, z wyjątkiem drobnych różnic w menu. Wśród argumentów podniesionych przez pozwanego znalazło się twierdzenie, że ochronie prawnoautorskiej mogą podlegać jedynie „tekstowe” elementy programu komputerowego (a więc kod źródłowy i kod wykonywalny programu). Ponieważ wskazane elementy interfejsu użytkownika nie należą do tak rozumianych „tekstowych” elementów programu, winny pozostawać poza zakresem ochrony prawnoautorskiej. Argument ten znacząco zawęził obszar sporu. W miejsce rozważenia, czy wskazane elementy interfejsu użytkownika mogą same w sobie podlegać ochronie w ramach prawa autorskiego, sąd

⁶³ *Broderbund Software, Inc. v. Unison World, Inc.*, 648 F. Supp. 1127, 231 U.S.P.Q. (BNA) 700 (N.D. Cal. 1986).

⁶⁴ J. Pinheiro, G. Lacroix, *Protecting the „Look and Feel” of computer software*, HTLJ 1987/2, s. 413.

⁶⁵ *Lotus Development Corp. v. Paperback Software International* 740 F Supp. 37 (D. Mass. 1990).

poddał analizie problem zdolności ochronnej „pozatekstowych” elementów programu komputerowego⁶⁶. Uznając ostatecznie, że ochrona prawnoautorska rozciąga się również na „pozatekstowe” elementy programu, sąd musiał przyjąć założenie, że prezentowane na ekranie elementy interfejsu użytkownika stanowią część programu komputerowego.

W celu porównania struktury menu arkusza kalkulacyjnego Paperback VP-Planner ze strukturą menu arkusza Lotus 1-2-3 oraz języka tzw. makropoleceń⁶⁷ sąd sformułował trójetapowy test, bazujący na zasadniczej tezie Whelan. Test zakładał kolejno: 1) określenie, w jaki sposób idea zawarta w programie może być wyrażona w ogóle i w jaki sposób została wyrażona w konkretnym przypadku; 2) badanie, które elementy danej formy wyrażenia są jedynym możliwym lub typowym sposobem wyrażenia danej idei w oparciu o *merger doctrine* i *scenes à faire doctrine*; 3) udzielenie odpowiedzi na pytanie, czy elementy, które nie są konieczne dla wyrażenia danej idei, a więc podlegają ochronie, stanowią istotną część danego dzieła. W oparciu o tak zarysowane rozumowanie sąd uznał, że jakkolwiek idea menu ma charakter funkcjonalny i jest oczywista, to już struktura menu jest elementem chronionej ekspresji. Nie musi być ona bowiem identyczna dla wyrażenia idei komunikacji programu z użytkownikiem. Jeśli chodzi o język makropoleceń, sąd wskazał, że kopiowanie struktury menu nie było jedynym sposobem na osiągnięcie zgodności makr napisanych dla arkusza 1-2-3 z arkuszem VP-Planner. W jego ocenie przykładem odmiennego rozwiązania może być funkcja konwersji makr, obecna w Microsoft Excel, „tłumacząca” makra napisane dla arkusza 1-2-3. W uzasadnieniu wyroku sąd jednoznacznie przyznał, że takie podejście oznacza rozciągnięcie ochrony na elementy funkcjonalne programu. Uznał jednak, że podejście odmienne nie zapewni odpowiedniego poziomu ochrony nieliteralnych elementów programu komputerowego.

Orzeczenie w sprawie Paperback, mimo że znacznie szerzej niż wyrok w sprawie Whelan ujmowało zakres niechronionych idei⁶⁸, spotkało się z krytyką w literaturze przedmiotu. Podkreśla się, że objęcie ochroną prawnoautorską elementów funkcjonalnych programu ogranicza innowacyjność przemysłu informatycznego oraz ignoruje potrzebę

⁶⁶ Orzeczenie to spotkało się z dosyć powszechną krytyką. Zob. P. Samuelson, *Computer Programs, User Interfaces and Section 102(b) of the Copyright Act of 1976: A Critique of Lotus v. Paperback*, L&CP 1992/55, s. 318; M.L. Pittman, *What the Judge Sees Is What You Get: The Implications of Lotus v. Paperback for Software Copyright*, „Wayne Law Review” 1991/37, s. 1533 i n.; D. Johansen, *Lotus Development v. Paperback Software: The Overextension Copyright Protection to Functional Aspects of Computer Software*, HC&ELJ 1992/14, s. 274. Odmienne A. Miller, *Copyright...*, s. 998. Zob. też polemikę D.S. Karjala, *Copyright protection of computer documents, reverse engineering and professor Miller*, „University of Dayton Law Review” 1994/19, s. 1002 i n.

⁶⁷ Makropolecenie (*macro command*) to zestaw rozkazów (poleceń menu, skrótów klawiaturowych, specyficznych poleceń języka makropoleceń) przeznaczonych do wykonywania przez określoną aplikację (edytor, arkusz kalkulacyjny, program graficzny). Makropolecenia umożliwiają automatyzację pewnych typowych czynności w czasie pracy z programem. W sprawie Lotus v. Paperback język makropoleceń był identyczny, co zapewniało możliwość uruchomienia makropoleceń napisanych dla arkusza kalkulacyjnego Lotus w programie Paperback.

⁶⁸ Zob. J. Drexler, *What is Protected...*, s. 83–84.

standaryzacji i zachowania interoperacyjności. W praktyce oznacza również objęcie ochroną prawnoautorską takich elementów programu, które co najwyżej mogłyby być przedmiotem ochrony patentowej, o ile spełniłyby surowe przesłanki zdolności ochronnej⁶⁹. Ograniczenie zakresu ochrony interfejsów użytkownika, do jakiego doszło w późniejszym orzecznictwie, wynikało właśnie z konieczności uwzględnienia wpływu ochrony interfejsów na funkcjonowanie rynku oprogramowania i jego konkurencyjność.

4.3. Test *extrinsic-intrinsic*

Sygnałem zmiany podejścia do ochrony interfejsów użytkownika stał się wyrok w sprawie *Brown Bag Software v. Symantec*⁷⁰. Spór dotyczył programu do tworzenia ustrukturyzowanych notatek (outlinera). Orzekający w drugiej instancji sąd 9 okręgu nie rozstrzygał tu kwestii, czy jego pozatekstowe elementy programu komputerowego podlegają ochronie w ramach prawa autorskiego, choć wcześniejsze wyroki tego sądu pozwalają udzielić tu odpowiedzi twierdzącej⁷¹. Zastosował natomiast wykorzystywany już w orzecznictwie dwuetapowy test (*extrinsic-intrinsic*) w celu ustalenia, czy zachodzi istotne podobieństwo uzasadniające zarzut naruszenia prawa autorskiego⁷².

Pierwsza faza testu (*extrinsic test*) polega na obiektywnym („zewnętrznym”) rozbiórce analitycznym porównywanych utworów. Rozbiór ten służy ustaleniu elementów wspólnych porównywanych utworów i oddzieleniu tych, które nie podlegają ochronie. Celem tego etapu testu nie jest ustalenie podobieństwa porównywanych utworów, a jedynie wyodrębnienie przedmiotu prawa autorskiego powoda. Na tym etapie sąd uznał za niechronioną ideę takie elementy, jak możliwość otwierania, edycji i drukowania istniejących plików jako należące do istoty programu komputerowego, ideę outlinera jako programu do tworzenia notatek i ogólny wygląd okna programu, rozwijane okna jako standard funkcjonujący w przemyśle komputerowym czy niebieskie tło jako element o charakterze wyłącznie funkcjonalnym⁷³. Pozostałe elementy należało więc zaliczyć

⁶⁹ D. Johansen, *Lotus...*, s. 286, 291; P. Samuelson, *Computer...*, s. 317 i n.; J. Velasco, *The Copyrightability...*, s. 265; A. Nowicka, *Prawnoautorska...*, s. 41.

⁷⁰ *Brown Bag Software v. Symantec Corp.*, 960 F.2d 1465 (9th Cir. 1992).

⁷¹ Między innymi w uzasadnieniu wyroku *Johnson Controls, Inc. v. Phoenix Control Sys., Inc.*, 886 F.2d 1173, 1175 (9th Cir. 1989), wskazano, że „to, czy pozatekstowe elementy programu komputerowego, w tym jego struktura, sekwencja i organizacja oraz interfejs użytkownika, są chronione, zależy od tego, czy, w zależności od okoliczności danego przypadku, dany element stanowi formę wyrażenia idei, czy też ideę jako taką”.

⁷² Po raz pierwszy zbliżony test zastosowano w sprawie *Arnstein v. Porter*, 154 F.2d 464 (2d Cir. 1946). Sąd 9 Okręgu zastosował go w nieco zmodyfikowanej postaci w sprawie *Sid & Marty Krofft Television Prods., Inc. v. McDonald's Corp.*, 562 F.2d 1157, 1162-65 (9th Cir. 1977). W sprawie *Brown Bag v. Symantec* sąd odwołał się do tego drugiego ujęcia. Zob. J. Velasco, *The Copyrightability...*, s. 243, 266; R.R. Cole, *Substantial Similarity in the Ninth Circuit*, HTLJ 1995/11, s. 421-422.

⁷³ Zob. G. Rinck, *The Maturing US Law on Copyright Protection for Computer Programs: Computer Associates v. Altai and Other Recent Case Developments*, EIPR 1992/10, s. 355.

do wyrażenia mogącego podlegać ochronie. Druga faza testu (*intrinsic test*) polega na subiektywnej („wewnętrznej”) ocenie podobieństwa formy wyrażenia, przeprowadzanej z perspektywy przeciętnego, rozsądnego odbiorcy. Obejmuje ona te elementy, które zostały zaliczane do ekspresji w pierwszej fazie testu. Ponieważ w sprawie *Brown Bag* żaden z elementów, dla których powód dochodził ochrony, nie przeszedł pierwszej fazy testu, przeprowadzenie takiego porównania nie było konieczne⁷⁴.

Przed wyrokiem w sprawie *Brown Bag* test *extrinsic-intrinsic* był kilkakrotnie stosowany w sprawach dotyczących gier komputerowych. W sprawie *Data East USA v. Epyx*⁷⁵ spór dotyczył gry *World Karate Championship*, którą powód uznał za naruszenie praw do jego gry *Karate Champ*. Sąd apelacyjny uznał, że 15 funkcji, które w pierwszej instancji zostały uznane za naruszające prawa powoda, są niezbędne dla urzeczywistnienia w grze samej idei karate. Przedstawienie w grze meczu z udziałem dwóch zawodników, ubranych zgodnie z zasadami tej dyscypliny sportu, używających tych samych ruchów, punktowanych tak samo jak w prawdziwych zawodach, jest nieodłącznie związane ze sportem karate i jako takie nie podlega ochronie, a nawet znaczne podobieństwo elementów niechronionych nie pozwala na stwierdzenie naruszenia. Podobne orzeczenie zostało wydane w sprawie *Frybarger v. IBM*⁷⁶. Sąd podkreślił w nim, że jakkolwiek w porównywanych grach istnieje wiele podobnych cech, to każda z tych cech należy do podstawowych idei gier komputerowych, użyte do realizacji tych cech środki ekspresji są zaś niezbędnymi lub przynajmniej standardowymi środkami urzeczywistnienia danej idei. W takich przypadkach ochrona prawnoautorska obejmuje tylko ochronę przed włączeniem do cudzego programu identycznych lub prawie identycznych elementów⁷⁷.

4.4. Sprawa *Apple Computer Inc. v. Microsoft Corp.*

Test *extrinsic-intrinsic* został zastosowany również w jednym z najgłośniejszych sporów dotyczących interfejsu użytkownika, tj. w sprawie *Apple Computer Inc. v. Microsoft Corp.* Spór ten zasługuje jednak na odrębne potraktowanie ze względu na swoją specyfikę oraz pewne modyfikacje wprowadzone do testu przez sąd pierwszej instancji i sąd apelacyjny 9 okręgu. Powód (Apple) wywodził w pozwie, że graficzny interfejs użytkownika (GUI) środowiska operacyjnego Microsoft (Windows) i nakładka na to środowisko Hewlett-Packard (NewWave) narusza prawa autorskie do systemów operacyjnych dla komputerów Apple Lisa i Apple Macintosh. Spółka Apple dochodziła przy tym ochrony dla przedstawień ekranowych (*visual displays*), traktowanych jako utwór audiowizualny, odrębny od generującego je programu komputerowego, a nie dla „pozatekstowych” elementów programu komputerowego. Znaczenia nie miał tu więc

⁷⁴ Zob. J. Velasco, *The Copyrightability...*, s. 267–268.

⁷⁵ *Data East USA, Inc. v. Epyx, Inc.* 862 F.2d 204 (9th Cir. 1988).

⁷⁶ *Frybarger v. IBM*, 812 F.2d 525 (9th Cir. 1987).

⁷⁷ Zob. szerzej R.R. Cole, *Substantial...*, s. 422–423.

brak jakiegokolwiek podobieństwa kodu źródłowego i wykonywalnego systemów Apple i Microsoft oraz oprogramowania Hewlett-Packard. Spośród zagadnień prawnych sądy musiały natomiast rozstrzygnąć m.in., w jaki sposób należy dokonywać oceny graficznych interfejsów użytkownika, jak rozróżnić elementy chronione i niechronione, a także jaki stopień podobieństwa decyduje o naruszeniu praw autorskich.

Początki sporu sięgają 1985 r., kiedy to Microsoft wypuścił na rynek swoje pierwsze graficzne środowisko operacyjne Windows 1.0, przeznaczone dla komputerów PC. System ten wykazywał pewne podobieństwa do GUI komputerów Apple Lisa i Macintosh. Zainicjowany przez Apple spór zakończył się podpisaniem umowy licencyjnej między Apple a Microsoft. Microsoft potwierdził w niej, że przedstawienia wizualne w Windows 1.0 stanowią opracowanie (utwór zależny) przedstawień wizualnych generowanych przez GUI Apple Lisa i Macintosh. Spółka Apple udzieliła zaś Microsoft licencji na „korzystanie z tych opracowań w obecnych i przyszłych produktach oraz na udzielanie licencji” osobom trzecim. Po upływie kilku lat Microsoft wypuścił Windows 2.03, a później 3.0. Natomiast jego licencjodawca, spółka Hewlett-Packard wprowadziła na rynek nakładkę NewWave, która miała ułatwiać korzystanie z systemu Windows. Spółka Apple uznała, że nowe wersje Windows i nakładka NewWave wykraczają poza zakres udzielonej licencji, sprawiając, że Windows jest bardziej „podobny do Maca”. W konsekwencji zarzuciła w pozwie skierowanym przeciw Microsoft i Hewlett-Packard naruszenie przysługujących jej praw autorskich. Jednym z zasadniczych argumentów Apple było przy tym twierdzenie, że podlegającą ochronie ekspresją są nie poszczególne przedstawienia ekranowe, a „ogólny wygląd i wrażenie” (*total look & feel*) interfejsu użytkownika Apple Macintosh, przede wszystkim metafora biurka (pulpitu) z oknami, ikonami i rozwijanymi menu, obsługiwanymi za pomocą myszy.

Zawarta przez Apple i Microsoft umowa licencyjna w pewnym stopniu narzuciła sposób podejścia sądów do oceny podobieństwa GUI. W serii wyroków⁷⁸ sąd pierwszej instancji uznał, że umowa zezwalała Microsoft i jego licencjodawcom na korzystanie z tych przedstawień wizualnych komputerów Apple Lisa i Macintosh, które były zawarte w Windows 1.0, a nie graficznego interfejsu Windows 1.0 jako całości. W konsekwencji przyjął, że elementy objęte licencją (a więc już obecne w systemie Windows 1.0) powinny być oddzielone przed zastosowaniem testów podobieństwa idei i ekspresji. Sąd stwierdził, że „ogólny wygląd i wrażenie” oprogramowania Apple, Microsoft i Hewlett-Packard może mieć pewne znaczenie dla sprawy, ale dopiero po zidentyfikowaniu

⁷⁸ Apple Computer, Inc. v. Microsoft Corp., 709 F. Supp. 925 (N.D. Cal. 1989) (Apple I); Apple Computer, Inc. v. Microsoft Corp., 717 F. Supp. 1428 (N.D. Cal. 1989) (Apple II); Apple Computer, Inc. v. Microsoft Corp., 759 F. Supp. 1444 (N.D. Cal. 1991) (Apple III); Apple Computer, Inc. v. Microsoft Corp., 779 F. Supp. 133 (N.D. Cal. 1991) (Apple IV); Apple Computer, Inc. v. Microsoft Corp., 799 F. Supp. 1006 (N.D. Cal. 1992) (Apple V); Apple Computer, Inc. v. Microsoft Corp., 821 F. Supp. 616 (N.D. Cal. 1993) (Apple VI).

niechronionych elementów ekspresji⁷⁹. Do ochrony prawnoautorskiej kwalifikuje się bowiem również oryginalne zestawienie elementów niechronionych⁸⁰.

Następnie sąd przeprowadził rozbiór analityczny interfejsów Macintosh, Windows 2.03 i 3.0 i NewWave w oparciu o listę podobieństw przedstawioną przez Apple, a następnie dokonał ich oceny pod kątem oryginalności oraz koncepcji, takich jak *merger doctrine*, *scenes à faire doctrine* i *short phrases doctrine*. Sąd uznał taki zabieg za niezbędny nie tylko w celu określenia przedmiotu praw autorskich powoda, ale z tego powodu, że naruszenie praw autorskich jest wykluczone, jeśli podobieństwo porównywanych utworów wynika wyłącznie z cech, które nie podlegają ochronie⁸¹. Zaznaczył również, że przy ustalaniu zakresu ochrony należy uwzględnić potrzebę standaryzacji rozwiązań, która przeważa nad indywidualnym interesem producenta zainteresowanego ich monopolizacją. Przyjęcie argumentów Apple prowadziłoby bowiem do objęcia zakresem jej praw autorskich wszystkich graficznych interfejsów użytkownika komputerów stacjonarnych, które wykorzystują standardowe funkcje takich interfejsów. Ochrona taka byłaby nadmierna i nie służyła rozwojowi rynku oprogramowania⁸².

W wyniku przeprowadzonej analizy sąd nie znalazł w interfejsie Windows 2.03 i 3.0 niemal żadnych nielicencjonowanych i podlegających ochronie elementów, skopiowanych z interfejsu komputerów Macintosh. Za potencjalnie chronione uznał zaś jedynie pewne detale interfejsu obecne w nakładce NewWave. W następnym kroku zamierzał porównać te potencjalnie chronione elementy w komputerach Macintosh z ich odpowiednikami w nakładce NewWave, żeby ustalić, czy zachodzi między nimi „istotne podobieństwo” (*substantial similarity*), uzasadniające naruszenie prawa autorskiego. Sąd chciał również porównać NewWave, Windows 2.03 i 3.0 jako całość z utworami Apple, przy czym do naruszenia prawa autorskiego miałyby doprowadzić ich „praktyczna identyczność” (*virtual identity*)⁸³. Ostatecznie jednak z przyczyn proceduralnych porównania takie nie zostały przeprowadzone i sąd orzekł na korzyść Microsoft i Hewlett-Packard.

Sąd apelacyjny zasadniczo zaakceptował ustalenia sądu pierwszej instancji⁸⁴. Uporządkował jednak zastosowany w sprawie test, opierając go na omówionych wyżej wyrokach w sprawach *Brown Bag v. Symantec*, *Data East USA v. Epyx* i *Frybarger v. IBM*. Zaznaczył przy tym, że odpowiedni test, który należy zastosować w danej sprawie oraz kolejność, w jakiej poszczególne elementy programu mają być badane mogą się różnić

⁷⁹ Apple Computer, Inc. v. Microsoft Corp., 759 F. Supp. 1444 (N.D. Cal. 1991) (Apple III), nb. 1449.

⁸⁰ Apple Computer, Inc. v. Microsoft Corp., 779 F. Supp. 133 (N.D. Cal. 1991) (Apple IV), nb. 135–136.

⁸¹ Apple Computer, Inc. v. Microsoft Corp., 799 F. Supp. 1006 (N.D. Cal. 1992) (Apple V), nb. 1020.

⁸² Zob. J. Drexler, *What is Protected...*, s. 32; T.S. Teter, *Merger and the Machines: An Analysis of the Pro-Compatibility Trend in Computer Software Copyright Cases*, Stan. L. Rev. 1993/4, s. 1096; D.L. Hayes, *A Comprehensive Current Analysis of Software „Look and Feel” Protection*, part I, CLSR 1995/6, s. 310.

⁸³ Zob. D.L. Hayes, *A Comprehensive...*, part I, s. 313–314; J. Myers, *Apple v. Microsoft Virtual Identity in the GUI Wars*, „Richmond Journal of Law and Technology” 1995/5, s. 15.

⁸⁴ Apple Computer Inc. v. Microsoft Corp., 35 F.3d 1435 (9th Cir. 1994).

w zależności od roszczeń powoda, przepisów proceduralnych oraz charakteru spornych programów komputerowych⁸⁵. W ocenianym stanie faktycznym ujął test w postaci trzech następujących po sobie kroków. W pierwszym kroku powód musi zidentyfikować źródło lub źródła rzekomego podobieństwa między jego utworem a utworem pozwanego. W drugim kroku, w oparciu o rozbiór analityczny, sąd musi ustalić, czy którakolwiek z rzekomo podobnych cech jest chroniona prawem autorskim. Jeżeli strony zawarły umowę licencyjną, sąd musi najpierw ustalić, które elementy są objęte licencją. Po określeniu zakresu licencji powinien oddzielić idee niepodlegające ochronie od ekspresji potencjalnie podlegającej ochronie. Po zidentyfikowaniu elementów ekspresji sąd musi następnie zastosować odpowiednie koncepcje ograniczające przedmiotowy zakres ochrony (*merger doctrine*, *scenes à faire doctrine*, oryginalność). W trzecim kroku, po zbadaniu domniemyanych podobieństw i rozważeniu zakresu możliwego wyrażenia, sąd musi określić zakres praw autorskich powoda, tj. zdecydować, czy dany utwór ma prawo do ochrony „szerokiej” czy „wąskiej”. W zależności od stopnia ochrony, sąd musi wyznaczyć odpowiedni standard subiektywnego porównania utworów, aby ustalić, czy jako całość są one wystarczająco podobne, aby uzasadnić stwierdzenie naruszenia prawa autorskiego. Decyzja sądu jest w znacznej części dyskrejonalna. Jeżeli uzna, że ochrona jest „szeroka”, tzn. jeżeli istnieje istotne natężenie cech podlegających ochronie, dla stwierdzenia naruszenia wystarczy ustalenie istotnego podobieństwa (*substantial similarity*) między porównywanymi utworami. Jeżeli jest on „wąski”, a więc wiele cech nie odznacza się oryginalnością, zastosowano *merger doctrine*, *scenes à faire doctrine*, dla zaistnienia naruszenia niezbędne jest, aby między porównywanymi utworami zachodziła praktyczna identyczność (*virtual identity*). Porównanie i związany z nim standard ochrony może przy tym dotyczyć utworów jako całości, jednak po wcześniejszym przeprowadzeniu rozbioru analitycznego i ocenie poszczególnych elementów⁸⁶.

Z prawnoporównawczego punktu widzenia interesujące są elementy, którym w drugim kroku testu odmówiono ochrony. Sąd zakwalifikował tak ideę graficznego interfejsu użytkownika i metaforę pulpitu. Zaaprobował też ustalenia sądu pierwszej instancji, że do sfery niechronionych idei należy: 1) wyświetlanie obrazów w oknach i umożliwienie użytkownikom interakcji z programami wyświetlającymi informacje w oknach; 2) reprezentacja poszczególnych obiektów (plików, programów) za pomocą ikon; 3) kontrolowanie pracy komputera za pomocą manipulacji ikonami; 4) użycie menu jako sposobu przechowywania informacji o funkcjach komputera w sposób łatwo dostępny dla użytkownika, ale niezajmujący zbyt wiele miejsca na ekranie; 5) reprezentacja przechowywania, przesyłania i uzyskiwania informacji jako zamykania i otwierania obiektów. Dlatego do naruszenia prawa autorskiego nie może prowadzić podobieństwo wynikające tylko z faktu, że Windows 2.03, 3.0 i NewWave mają okna, ikony reprezentujące znane przedmioty ze środowiska biurowego, które opisują wykonywane funkcje

⁸⁵ Zob. E. Derclaye, *Software...*, part 2, s. 59.

⁸⁶ D.L. Hayes, *A Comprehensive Analysis of Software Look and Feel Protection*, part II. *Apple v. Microsoft*, CLSR 1996/1, s. 15.

i które mogą być przesuwane po ekranie w celu sterowania pracą komputera, menu, które dają łatwy dostęp do informacji lub funkcji bez użycia miejsca na ekranie, lub obiekty, które można otwierać i zamykać.

4.5. Test *abstraction-filtration-comparison* w sporach dotyczących interfejsów użytkownika

Test AFC wypracowany w sprawie *Altai i Gates Rubber* szybko został zastosowany w sporach, w których nieliteralne kopiowanie programu miało polegać na podobieństwie interfejsu użytkownika. Jest on przydany w tych przypadkach, w których powód opiera swoje roszczenia na podobieństwie interfejsów jako całości, nie zaś na podobieństwie poszczególnych elementów składowych. Przykładem może być wyrok sądu apelacyjnego 5 okręgu w sprawie *Engineering Dynamics, Inc. v. Structural Software*⁸⁷.

Przedmiotem sporu był format wprowadzania danych (*input formats*) i raportów wyjściowych (*output reports*) w oprogramowaniu do analizy strukturalnej. Formaty wprowadzania danych, w postaci zorganizowanych, opisowych tabel do wprowadzania danych, miały swoją genezę w czasach, gdy dane do komputerów były wprowadzane za pomocą kart perforowanych i powód w części zapożyczył je z wcześniejszego oprogramowania⁸⁸. Z tego względu nie rościł sobie praw do ochrony żadnego z licznych formatów wejściowych i raportów wyjściowych. Twierdził natomiast, że sekwencja i organizacja formatów i raportów podlega jako całość ochronie prawnoautorskiej. Sąd pierwszej instancji orzekł na korzyść pozwanego, odmawiając ochrony⁸⁹. Sąd apelacyjny uchylił wyrok sądu pierwszej instancji i przekazał sprawę do ponownego rozpoznania, nakazując zastosowanie testu AFC zgodnie z zawartymi w wyroku wytycznymi. Na etapie filtracji doszedł przy tym do wniosku, że formaty wprowadzania danych i raportów rozpatrywane jako całość mogą stanowić chronioną ekspresję z uwagi na ich rolę w pośredniczeniu między programem a użytkownikiem, określaniu wymaganych informacji i sposobie ich uporządkowania. Wskazówki sądu odnoszące się do tego etapu testu AFC opierają się przy tym w dużym zakresie na wyroku sądu dystryktowego w sprawie *Lotus Development Corp. v. Paperback Software International* i można im zarzucić ten sam błąd. Sąd uznał bowiem w istocie, że kreatywność w metodologicznych lub systemowych cechach interfejsu programu może sama w sobie prowadzić do ochrony prawnoautorskiej⁹⁰.

⁸⁷ *Engineering Dynamics, Inc. v. Structural Software, Inc.*, 26 F.3d 1335 (5th Cir. 1994).

⁸⁸ Zob. w tym zakresie cytowany już wyrok *Synercom Technology, Inc. v. University Computing Co.*, 462 F. Supp. 1003, 199 U.S.P.Q. (BNA) 537 (N.D. Tex. 1978).

⁸⁹ *Engineering Dynamics, Inc. v. Structural Software, Inc.*, 785 F.Supp. 576, 582 (E.D.La. 1991).

⁹⁰ Zob. D.S. Karjala, *A Coherent Theory for the Copyright Protection of Computer Software and Recent Judicial Interpretations*, U. Cin. L. Rev. 1997/66, s. 108.

Kolejnym przykładem zastosowania testu AFC w sprawach dotyczących interfejsów jest spór w sprawie *Productivity Software International Inc. v. Healthcare Technologies Inc.*⁹¹ Naruszenie miało tu polegać, podobnie jak w sprawie *Engineering Dynamics, Inc. v. Structural Software*, na przejęciu projektu układu pól służących do wprowadzania danych. Sąd nie „odfiltrował” wszystkich elementów interfejsu użytkownika w czasie drugiego etapu testu, w konsekwencji przyznając „słabą” ochronę konkretnemu układowi pól wyświetlanych na ekranie, wskazując na jego niewielką ekspresywność⁹².

4.6. Interfejs użytkownika jako „metoda działania” lub „proces”

Odmienne podejście zostało zastosowane w sprawie *Lotus Development Corp. v. Borland International Inc.* Przedmiotem sporu był tu „tryb emulacji” arkusza kalkulacyjnego Lotus 1-2-3 w programie Quattro Pro. Przełączenie arkusza Quattro Pro w tryb emulacji powodowało, że układ poszczególnych elementów interfejsu obejmujący hierarchię poleceń menu stawał się niemal identyczny, jak w przypadku programu Lotus 1-2-3. „Tryb emulacji” miał dwa cele. Po pierwsze ułatwiał użytkownikom przyzwyczajonym do arkusza Lotus 1-2-3 rozpoczęcie korzystania z arkusza Quattro Pro. Po drugie pozwalał na uruchamianie makr (makropoleceń) napisanych dla arkusza Lotus 1-2-3, które odwoływały się do określonych poleceń z menu⁹³. W czasie sporu Borland usunął z arkusza Quattro Pro „tryb emulacji” w ten sposób, że nie dołączał do programu „trybu emulacji” menu arkusza Lotus 1-2-3. Wprowadził jednak w to miejsce funkcję Key Reader, która powodowała, że arkusz Quattro Pro odczytywał sekwencję znaków składającą się na makropolecenie arkusza Lotus 1-2-3, a następnie wykonywał ją dokładnie tak, jakby hierarchia menu Quattro Pro odpowiadała menu arkusza Lotus 1-2-3. W związku z tym powód rozszerzył powództwo o żądania związane z funkcją Key Reader, upatrując naruszenia w zamieszczeniu w arkuszu Quattro Pro niewidocznych dla użytkownika

⁹¹ *Productivity Software International, Inc. v. Healthcare Technologies Inc.*, No. 93 Civ. 6949, 1995 WL 437526 (S.D.N.Y. 1995).

⁹² Zob. E. Derclaye, *Software...*, part. 2, s. 58.

⁹³ Zrozumienie istoty sporu wymaga wyjaśnień, jak działał system makropoleceń w arkuszu Lotus 1-2-3. Makro 1-2-3 składało się z tekstu zawartego w komórce arkusza kalkulacyjnego poprzedzonego ukośnikiem („/”). Gdy makro było wywoływane, program rozpoczynał odczytywanie tekstu po ukośniku od lewej do prawej, traktując go tak, jakby użytkownik wcisnął odpowiedni skrót klawiaturowy, wywołując określone polecenia programu. Na przykład po wywołaniu makra składającego się z tekstu „/wp” program zachowywał się dokładnie tak, jakby użytkownik wpisał do zwykłego interfejsu „/” (wywołując menu główne), a następnie „w” (wybierając z menu podmenu „Worksheet”), a następnie „p” (wybierając pozycję „Page”). Odzwierciedlenie struktury menu pozwalało więc na wykonywanie makropoleceń w taki sam sposób, jakby były uruchamiane w arkuszu Lotus 1-2-3. Co do aspektów technicznych zob. M. Hamilton, T. Sabety, *Computer Science Concepts in Copyright Cases*, Harv. JL & Tech. 1997/2, s. 274. Por. też uwagi w pkt 4.2 dotyczące sporu *Lotus Development Corp. v. Paperback Software International* 740 F Supp. 37 (D. Mass. 1990).

tabel, odtwarzających hierarchię menu i polecenia arkusza Lotus 1-2-3. Między stronami niesporne było natomiast, że kod obu programów był całkowicie różny.

W pierwszej instancji spór dotyczył najpierw „trybu emulacji”. Sąd, stosując test wypracowany w sprawie Paperback, uznał, że hierarchia menu arkusza Lotus 1-2-3 stanowi chronioną ekspresję. Zgodnie z rozumowaniem sądu poza zakresem prawa autorskiego pozostaje zorganizowanie poleceń sterujących pracą programu za pomocą hierarchicznie ułożonych menu i podmenu. Jednak już konkretny sposób ich ułożenia podlega ochronie, podobnie jak konkretne nazwy poszczególnych poleceń. Zdaniem sądu zamiast polecenia „Copy” Borland mógł użyć w menu np. polecenia „Replicate”, „Duplicate”, „Reproduce”, „Repeat”, „Ditto”. Po rozszerzeniu powództwa sąd analogicznie ocenił funkcję Key Reader. Stwierdził, że tabele niezbędne dla działania funkcji Key Reader zawierają „praktycznie identyczną kopię struktury drzewa menu Lotusa, ale reprezentowaną w innej formie i z pierwszymi literami nazw poleceń menu w miejsce pełnych nazw poleceń menu”. Ponieważ „struktura menu Lotusa, organizacja i pierwsze litery nazw komend [...] stanowią część chronionej ekspresji” arkusza Lotus 1-2-3, Borland naruszył prawa autorskie powoda⁹⁴.

W wyniku apelacji złożonej przez spółkę Borland sprawa trafiła przed sąd apelacyjny 1 okręgu. W pierwszej kolejności sąd rozważał, czy w tej sprawie może być zastosowany test AFC, sformułowany w sprawie Altai. W ocenie sądu test ten nie powinien być jednak stosowany do przypadków literalnego kopiowania⁹⁵. Ponieważ w trybie emulacji interfejs arkusza Quatro Pro był identyczny z interfejsem arkusza Lotus 1-2-3, niezbędne jest odwołanie się do innych kryteriów oceny. Sąd zaaprobował natomiast podniesiony przez Borland argument, że hierarchia poleceń w menu programu stanowi system, metodę działania, proces lub procedurę wyłączone spod ochrony stosownie do § 102 (b) *Copyright Act*. Uznał, że użyty w tym przepisie termin „metoda działania” (*method of operation*) nie odnosi się do abstrakcyjnej idei, a konkretnego sposobu, w jaki dana osoba coś obsługuje, czy to będzie samochód, robot kuchenny czy komputer. Paragraf 102 (b) *Copyright Act* pozostawia więc poza zakresem ochrony nie tylko ideę obsługi programu za pomocą poleceń sterujących zorganizowanych w hierarchicznie ułożone menu, ale również konkretny sposób, w jaki obsługa ta jest realizowana, nawet jeżeli przy projektowaniu menu twórcy arkusza 1-2-3 dokonali pewnych ekspresyjnych wy-

⁹⁴ Lotus Dev. Corp. v. Borland Int'l, Inc., 831 F.Supp. 223, 230, 30 USPQ2d 1081, 1099 (D. Mass. 1993). Rozstrzygnięcie to spotkało się z krytyką części doktryny. Zob. D.S. Karjala, P.S. Menell, *Applying Fundamental Copyright Principles to Lotus Development Corp. v. Borland International, Inc.*, BTLJ 1995/10, s. 177; P. Samuelson, *Brief Amicus Curiae of Copyright Law Professors in Lotus Development Corp. v. Borland International, Inc.*, J. Intell. Prop. L. 1995/3, s. 117. Argumenty podniesione przez powołanych autorów znalazły odzwierciedlenie w późniejszym wyroku sądu apelacyjnego.

⁹⁵ Zastosowanie testu AFC do przypadków literalnego kopiowania było przedmiotem kontrowersji, a test został odrzucony m.in. w sprawie Data Gen. Corp. v. Grumman Sys. Support Corp., 803 F.Supp. 487 (D. Mass. 1992). Dopiero w sprawie Bateman v. Mnemonics, Inc., 37 U.S.P.Q.2d 1225 (11th Cir. 1995), sąd apelacyjny potwierdził zasadność stosowania testu AFC również do przypadków literalnego kopiowania. Zob. E. Derclaye, *Software...*, part 2, s. 59.

borów. Sąd porównał polecenia w menu do przycisków sterujących pracą magnetowidu i podkreślił, że uporządkowanie i oznakowanie przycisków nie czyni z nich „dzieła literackiego” ani też nie czyni z nich „ekspresji” abstrakcyjnej „metody obsługi” magnetowidu za pomocą zestawu oznakowanych przycisków. Zamiast tego przyciski są same w sobie „metodą obsługi”, tak samo jak „metodą obsługi” są umieszczone w menu polecenia, za pomocą których użytkownik kontroluje program. W sprawie *Borland* przejście hierarchii poleceń menu programu *Lotusa* zarówno w „trybie emulacji”, jak i w oparciu o funkcję *Key Reader* stanowi więc wyłącznie przejście „metody działania” pozostającej poza zakresem ochrony prawnoautorskiej⁹⁶.

Podobne motywy legły u podstaw rozstrzygnięcia w sprawie *Manufacturers Technologies, Inc. v. Cams, Inc.*⁹⁷ W tym wypadku za ideę wyłączoną spod ochrony na podstawie § 102 (b) *Copyright Act* sąd uznał elementy umożliwiające nawigację między różnymi komponentami programu i konkretne przedstawienia ekranowe. Paragraf 102 (b) *Copyright Act* został zastosowany również w sprawie *MiTek Holdings Inc. v. Arce Engineering Co. Inc.*⁹⁸ w ramach testu wzorowanego na zastosowanym w sprawie *Apple Computer Inc. v. Microsoft Corp.* Inaczej niż w sprawie *Lotus v. Borland* brak było identyczności elementów interfejsu porównywanych programów. Program pozwanego (*Arce*) używał ikon w przeciwieństwie do słów używanych w programie powoda. Podobieństwo dotyczyło jednak przebiegu etapów komputerowo wspomaganego projektowania kratownic drewnianych. Sąd uznał, że odzwierciedlone w programie etapy projektowania stanowią „proces” wyłączony spod ochrony w świetle § 102 (b) *Copyright Act*. Układ etapów naśladuje bowiem kroki, które projektant wykonałby, rysując ręcznie projekt takich kratownic. Nie dopatrywał się więc naruszenia.

4.7. Krytyka w nauce prawa

Kierunek, w jakim rozwinęło się orzecznictwo sądów w sporach dotyczących interfejsów spotkał się z krytyką części przedstawicieli nauki prawa, którzy wyrażali obawy, że doprowadzi on do stagnacji na rynku oprogramowania spowodowanej niedostatecznym poziomem ochrony⁹⁹. Wyrażano przy tym przekonanie, że prawo autorskie w ujęciu zastosowanym w sprawach *Brown Bag Software v. Symantec*, *Apple Computer*

⁹⁶ *Lotus Development Corp. v. Borland International Inc.*, 49 F.3d 807.

⁹⁷ *Manufacturers Technologies, Inc. v. Cams, Inc.*, 706 F. Supp. 984 (D. Connecticut 1989). Zob. B. Dorny, M. Friedland, *Copyrighting „Look and Feel”*. *Manufactures Technologies v. CAMS*, Harv. JL & Tech. 1990/3, s. 195 i n.

⁹⁸ *MiTek Holdings Inc. v. Arce Engineering Co. Inc.*, 89 F. 3d 1584 (11th Cir. 1996).

⁹⁹ Zob. J. Myers, *AppleI...*, s. 5; J. Phillips, *Sui Generis Intellectual Property Protection for Computer Software*, GWLR 1992, s. 1007; J.A. Whong, A.T. Lee, *Lotus v. Borland: Defining the Limits of Software Copyright Protection*, HTLJ 1996/12, s. 217; D.R. Owen, *Interfaces and Interoperability in Lotus v. Borland: A Market-Oriented Approach to the Fair Use Doctrine*, „Fordham Law Review” 1995/64, s. 2403. Por. też E. Derclaye, *Software...*, part 2, s. 61 i cytowaną tam literaturę.