

Spis treści

Od Autora	17
Dla kogo jest ta książka?	17
Dlaczego ta książka jest taka gruba?	18
Co znajduje się w tej książce?	18
Co znajduje się na dyskietce?	18
Czego będziesz potrzebował?	18
Rozdział 1.	
Ogólna charakterystyka języka AutoLISP	21
1.1. Kompilatory a interpretery	21
1.2. AutoLISP jako język	22
1.3. Dlaczego LISP?	22
1.4. Typy danych w AutoLISPie	23
1.5. Symbole predefiniowane	25
1.6. Konwencje leksykalne	25
1.7. Zmienne w AutoLISPie	27
1.8. Wyrażenia AutoLISPu	29
1.9. Funkcje	29
1.10. Programy	30
1.11. Komentarze	30
1.12. Sposób pracy interpretera AutoLISPu	31
1.13. Obsługa błędów	32
Rozdział 2.	
Zanim zaczniemy programować	33
2.1. Przystosowanie edytora tekstu do pracy z poziomym programem AutoCAD	33
2.1.1. Nazwa polecenia	34
2.1.2. Polecenie SO (systemu operacyjnego)	34
2.1.3. Obszar zwalnianej pamięci	34
2.1.4. Zgłoszenie	34
2.1.5. Kod powrotu	35
2.2. Kilka najważniejszych uwag na temat programowania w AutoLISPie	37

2.3. Ładowanie i uruchamianie programów w AutoLISPie	39
2.3.1. Interakcyjne ładowanie programów w AutoLISPie	39
2.3.2. Ładowanie programów w AutoLISPie za pomocą pliku ACAD.LSP	41
2.3.3. Ładowanie programów w AutoLISPie za pomocą okna dialogowego LOAD AUTOLISP AND ADS FILES	43
2.3.4. Funkcja S::STARTUP	44
2.3.5. Uruchamianie programów w AutoLISPie	53
Podsumowanie.....	54
Rozdział 3.	
Zrozumienie interpretera AutoLISPU	55
3.1. Atomy	55
3.1.1. Atom cyfrowy	56
3.1.2. Atom literowy — identyfikator zmiennej	56
3.2. Listy	56
Podsumowanie.....	60
Rozdział 4.	
Ogólny zarys programu w języku AutoLISP	63
4.1. Pierwszy program	64
4.2. Drugi program	66
4.2.1. Komentarze	68
Podsumowanie.....	69
Rozdział 5.	
Instrukcje podstawiania	71
5.1. Instrukcja podstawiania SETQ	72
5.1.1. Przypisanie liczb całkowitych i rzeczywistych	72
5.1.2. Przypisanie stałych znakowych	73
5.1.3. Przypisanie stałych tekstowych (łańcuchów znaków)	75
5.1.4. Przypisanie list	77
5.2. Instrukcja podstawiania SET	77
5.2.1. Zwrot wartości przez funkcję pod podanym identyfikatorem.....	78
5.2.2. Dynamiczne tworzenie zmiennych.....	82
Podsumowanie.....	86
Rozdział 6.	
Operatory: arytmetyczne, porównania oraz logiczne .	87
6.1. Prawda i fałsz w języku AutoLISP.....	88
6.2. Operatory arytmetyczne	88
6.2.1. Standardowe operatory arytmetyczne.....	88
6.2.2. Operator REM.....	89
6.2.3. Operatory inkrementacji i dekrementacji	91

6.3. Operatory porównania	91
6.3.1. Standardowe operatory porównania (występujące również w innych językach programowania)	91
6.3.2. Operatory porównania charakterystyczne dla języka AutoLISP	92
(atom <i>item</i>)	92
(boundp <i>atom</i>)	92
(eq <i>expr1 expr2</i>)	93
(equal <i>expr1 expr2 [fuzz]</i>)	94
(listp <i>item</i>)	94
(minusp <i>item</i>)	95
(null <i>item</i>)	95
(numberp <i>item</i>)	95
(zerop <i>item</i>)	96
6.4. Operatory logiczne	96
6.4.1. Operatory sumy logicznej OR i iloczynu logicznego AND	96
6.4.2. Operator negacji NOT	99
Podsumowanie	101

Rozdział 7.

Operatory konwersji	103
(angtof <i>string [mode]</i>)	103
(angtos <i>angle [mode [precision]]</i>)	104
(ascii <i>string</i>)	104
(atof <i>string</i>)	105
(atoi <i>string</i>)	105
(chr <i>number</i>)	106
(cvunit <i>value from to</i>)	106
(distof <i>string [mode]</i>)	107
(fix <i>number</i>)	107
(float <i>number</i>)	107
(itoa <i>int</i>)	108
(rtos <i>number [mode [precision]]</i>)	108
Podsumowanie	117

Rozdział 8.

Pętle i konstrukcje sterujące	119
8.1. Instrukcja warunkowa IF	121
8.1.1. Instrukcje proste i złożone	123
8.1.2. Zagnieżdżone instrukcje IF	125
8.2. Pętla WHILE	132
8.2.1. Zagnieżdżone instrukcje WHILE	136
8.3. Instrukcja COND — wybór wielowariantowy	138
8.4. Pętla REPEAT	147
8.4.1. Zagnieżdżone instrukcje REPEAT	149

8.5. Realizacja pętli FOR w AutoLISPie	150
8.5.1. Zagnieżdżone instrukcje FOR	162
8.6. Realizacja pętli DO_WHILE w AutoLISPie.....	164
8.7. Nawiasy w instrukcjach sterujących	169
Podsumowanie.....	173

Rozdział 9.

Funkcja COMMAND	175
9.1. Oczekiwanie na akcję użytkownika	180
9.2. Przesyłanie wskazań punktów do poleceń AutoCADa	181
Podsumowanie.....	182

Rozdział 10.

Działania na listach.....	183
10.1. Tworzenie listy	189
10.1.1. Tworzenie listy za pomocą funkcji LIST	189
10.1.2. Tworzenie listy za pomocą funkcji CONS	191
10.1.3. Funkcja APPEND — umieszczanie elementów na liście już utworzonej.....	194
10.2. Wyszukiwanie i zamiana elementów na liście	198
10.2.1. Określanie długości listy.....	199
10.2.2. Wyszukiwanie elementu listy o zadanym kluczu.....	200
10.2.3. Wyszukiwanie n-tego elementu listy.....	201
10.2.4. Dane ostatniego elementu listy.....	203
10.2.5. Funkcje CAR i CDR oraz ich kombinacje	203
10.2.6. Zamiana elementów na liście.....	205
10.2.7. Odwracanie kolejności elementów na liście.....	215
10.2.8. Zwrot podlisty od danego elementu	215
10.2.9. Wyszukiwanie minimum i maksimum w podanej liście	217
10.2.10. Ustalenie pozycji elementu na liście	222
10.2.11. Wyświetlenie wybranych elementów z listy	226
10.3. Kasowanie elementów na liście	231
10.3.1. Usuwanie wybranego elementu na podanej liście.....	232
10.3.2. Usuwanie wybranej grupy elementów na podanej liście	234
10.3.3. Usuwanie elementów listy o podanych numerach (indeksach).....	242
10.4. Sortowanie listy	244
10.5. Lista DXF opisu elementów	249
10.5.1. Dostęp do listy DXF	250
10.5.2. Modyfikacja elementu	267
10.5.2.1. Modyfikacja elementu w wykorzystaniem funkcji entmod	268
10.5.2.2. Modyfikacja elementu z wykorzystaniem funkcji entmod i entupd	278
Podsumowanie.....	283

Rozdział 11.	
Działania na łańcuchach.....	285
11.1. Standardowe funkcje AutoLISPu do obsługi łańcuchów.....	287
(<i>strcase string [which]</i>).....	287
(<i>strcat string1 [string2] ...</i>).....	287
(<i>strlen [string] ...</i>).....	287
(<i>substr string start [length]</i>).....	288
(<i>wcmatch string pattern</i>).....	288
11.2. Funkcje klasyfikujące poszczególne znaki	295
11.3. Funkcje wyznaczające porządek leksykograficzny znaków	306
11.4. Pozostałe funkcje do obróbki łańcuchów	311
Podsumowanie.....	349
Rozdział 12.	
Definiowanie własnych funkcji	351
12.1. Metoda projektowania „od ogółu do szczegółu” oraz „od szczegółu do ogółu” — pojęcie funkcji	353
12.2. Funkcja DEFUN	360
12.3. Argumenty funkcji — sposób ich pobierania i przekazywania, konwersacyjne(interakcyjne) i argumentowe wywoływanie funkcji	362
Metoda 1.....	367
Metoda 2.....	370
12.3.1. Konwersacyjne (interakcyjne) wywołanie funkcji	371
12.3.2. Argumentowe wywołanie funkcji.....	372
12.3.3. Połączenie konwersacyjnego i argumentowego wywołania funkcji	374
12.4. Sposoby zwrotu rezultatu (rezultatów)działaniafunkcji.....	376
12.4.1. Zwrot pojedynczej wartości działania funkcji.....	377
12.4.2. Zwrot dwóch lub więcej wartości działania funkcji.....	378
12.5. Zmienne globalne i lokalne — zasięg zmiennej w programie	385
12.6. Funkcje typu C:XXX — definiowanie nowych poleceń AutoCADa	394
12.7. Funkcje rekurencyjne	398
12.8. Definiowanie funkcji wewnątrz innych funkcji	405
12.9. Funkcje LAMBDA, APPLY, FOREACH i MAPCAR	408
12.10. Budowa, zasada działania iwykorzystanielisty-funkcji	416
12.10.1. Realizacja pętli FOR.....	422
12.10.2. Realizacja pętli DO_WHILE.....	423
12.10.3. Wielolinia przebiegająca przez zadane punkty	425
12.10.4. Dynamiczna konstrukcja argumentów funkcji INITGET	429
12.11. Wywoływanie funkcji użytkownika z menu AutoCADa.....	438
Podsumowanie.....	440

Rozdział 13.

Pobieranie informacji od użytkownika	441
13.1. Funkcje typu GETXXX.....	442
13.2. Funkcja INITGET	449
13.2.1. Opcje odpowiedzi dla funkcji pobierających dane od użytkownika	450
13.2.2. Opcje słów kluczowych	454
13.3. Wartości domyślne w funkcjach typu GETXXX.....	461
13.4. Dowolne odpowiedzi z klawiatury	465
13.5. Wywołanie HELPu z funkcji typu GETXXX	473
Podsumowanie.....	481

Rozdział 14.

Tworzenie i operacje na zbiorach wyboru	483
14.1. Tworzenie zbiorów wyboru za pomocą funkcji SSET.....	484
14.1.1. Listy filtrów dla zbiorów wskazań	487
14.1.2. Znaki uniwersalne w listach filtrujących.....	488
14.1.3. Testowanie relacji.....	488
14.1.4. Logiczne grupowanie testów filtrujących.....	489
RYSUNEK_PROTOTYPOWY.....	495
UTWORZ_ZBIORY_WYBORU	495
WYPISZ_LISTY_DXF.....	496
ZWOLNIJ_ZBIORY_WYBORU.....	496
14.2. Operacje na zbiorach wyboru.....	497
RYSUNEK_PROTOTYPOWY.....	503
OPERACJE_NA_ZBIORACH_WSKAZAN.....	503
Podsumowanie.....	514

Rozdział 15.

Obsługa plików tekstowych w AutoLISPie	515
15.1. Plik jako nośnik informacji	517
15.2. Pobieranie nazwy pliku	518
15.2.1. Pobieranie nazwy pliku za pomocą funkcji GETSTRING.....	520
15.2.2. Pobieranie nazwy pliku za pomocą funkcji GETFILED.....	521
15.2.3. Sprawdzanie istnienia pliku za pomocą funkcji FINDFILE	528
15.2.4. Funkcja RETURN_PATH — rozszerzenie ścieżki poszukiwania plików	530
15.3. Otwieranie i zamykanie pliku — tryby otwarcia pliku, funkcje wprowadzania i wyprowadzania.....	535
15.4. Zapis danych do pliku	537
15.4.1. Zapis danych pojedynczo znak po znaku	537
15.4.2. Zapis danych łańcuchami	541
CZEKAJ	553
FORMAT_VALUE.....	553
UTWORZ_LISTY_DANYCH	553
ZAPISZ_MODUL_CONTROL.....	553

ZAPISZ_WEZLY	553
ZAPISZ_ELEMENTY	554
ZAPISZ_ZBIOR_DANYCH	554
15.4.3. Zapis danych w postaci listy	554
15.4.4. Formatowany zapis danych	559
Znaki typu	560
Specyfikator precyzji	561
Specyfikator szerokości	561
Flagi.....	561
15.4.5. Porównanie zapisu do pliku przykładowych linii z wykorzystaniem funkcji PRINC, PRIN1, WRITE-LINE oraz PRINTF	572
PRINC	577
PRIN1	577
WRITE-LINE.....	578
FPRINTF	578
15.4.6. Zapis danych w postaci rekordów	578
15.5. Odczyt danych z pliku	585
15.5.1. Odczyt danych pojedynczo znak po znaku.....	585
15.5.2. Odczyt danych łańcuchami.....	587
FOR	607
RYSUNEK_PROTOTYPOWY	607
ZMIENNE_GLOBALNE	607
ODBIERZ_ZBIOR_DANYCH	607
CZYTAJ_ZBIOR_DANYCH.....	607
CZYTAJ_MODUL_NODES	608
CZYTAJ_MODUL_ELEMENTS	608
OBSZAR_MODELU	608
MIN_MAX_NODES	608
GRANICE	608
ODBIERZ_WEZLY.....	608
ODBIERZ_ELEMENTY	608
USTAW_MODEMACRO	608
ODBIERZ_ELEMENT_2W	608
ODBIERZ_ELEMENT_3W	609
ODBIERZ_ELEMENT_4W	609
DOPISZ_NUMER_ELEMENTU	609
C:TEST.....	609
RYSUNEK_PROTOTYPOWY	609
ZMIENNE_GLOBALNE	609
ODBIERZ_ZBIOR_DANYCH	610
CZYTAJ_ZBIOR_DANYCH.....	610
CZYTAJ_MODUL_NODES	611
CZYTAJ_MODUL_ELEMENTS	611
OBSZAR_MODELU	612
MIN_MAX_NODES	612
GRANICE	612
ODBIERZ_WEZLY.....	613

ODBIERZ_ELEMENTY	613
USTAW_MODEMACRO	614
ODBIERZ_ELEMENT_2W	614
ODBIERZ_ELEMENT_3W	615
ODBIERZ_ELEMENT_4W	615
DOPISZ_NUMER_ELEMENTU	615
C:TEST.....	615
15.5.3. Odczyt danych w postaci listy	615
15.5.4. Formatowany odczyt danych	617
PAUSE_1	630
ERROR_SSCANF	630
CHECK_ARGUMENT_TYPE.....	630
READ_LISP_DATA.....	630
CHECK_ARGUMENT_FORMAT	631
15.5.5. Odczyt danych w postaci rekordów.....	638
15.6. Dodawanie danych do pliku	643
15.7. Wydruk pliku na drukarkę lub ekran (monitor)	646
WYSWIETL_PLIK.....	649
DRUKUJ_PLIK	649
Podsumowanie.....	649

Dodatek A

Spis funkcji AutoLISPU 651

Funkcje AutoLISPU posiadające odpowiedniki w ADS	651
Obsługa funkcji.....	651
Obsługa błędów	651
Zapytania i polecenia AutoCADa.....	651
Narzędzia geometryczne.....	652
Wprowadzanie danych przez użytkownika	652
Konwersje	653
Transformacje układu współrzędnych.....	653
Kalibrowanie pulpitu graficznego	654
Sterowanie wyświetlaczem.....	654
Funkcje graficzne niskiego poziomu	654
Badanie zgodności łańcuchów ze znakami uniwersalnymi.....	655
Zbiory wskazań.....	655
Obsługa elementów.....	655
Dodatkowe dane elementu.....	656
Tablice symboli.....	656
Funkcje ogólnego przeznaczenia.....	656
Algebraiczne	656
Obsługa symboli.....	658
Łańcuchy tekstowe.....	659
Konwersje	659
Relacje.....	660
Operacje na listach.....	660
Obsługa plików	661

Obsługa aplikacji ADS	662
Wyświetlacz	662
Obsługa funkcji	662
Obsługa pamięci	663
Różne	663
Funkcje AutoLISPu zdefiniowane w ADS	663
Polecenia zdefiniowane w ADS	663
Funkcje programowalnych okien dialogowych	664
Otwieranie i zamykanie plików DCL	664
Otwieranie i zamykanie okien dialogowych	664
Inicjowanie wyrażeń akcji lub funkcji zwrotnych	665
Obsługa wycinka atrybutów	665
Ustalanie okien kartotekowych i kartotek rozwijalnych	665
Tworzenie obrazów	665
Dane specyficzne dla aplikacji	666

Dodatek B

Katalog funkcji AutoLISPu 667

(+ <i>number number</i> ...)	667
(- <i>number [number]</i> ...)	667
(* <i>number [number]</i> ...)	668
(/ <i>number [number]</i> ...)	668
(= <i>atom atom</i> ...)	668
(/= <i>atom atom</i> ...)	668
(< <i>atom atom</i> ...)	669
(<= <i>atom atom</i> ...)	669
(> <i>atom atom</i> ...)	669
(>= <i>atom atom</i> ...)	670
(~ <i>number</i>)	670
(1+ <i>number</i>)	670
(1- <i>number</i>)	670
(abs <i>number</i>)	670
(ads)	671
(alert <i>string</i>)	671
(alloc <i>number</i>)	672
(and <i>expr</i> ...)	672
(angle <i>pt1 pt2</i>)	672
(angtof <i>string [mode]</i>)	672
(angtos <i>angle [mode [precision]]</i>)	673
(append <i>expr</i> ...)	674
(apply <i>function list</i>)	674
(ascii <i>string</i>)	674
(assoc <i>item alist</i>)	674
(atan <i>num1 [num2]</i>)	675
(atof <i>string</i>)	675
(atoi <i>string</i>)	676
(atom <i>item</i>)	676

(atoms-family <i>format</i> [<i>symlist</i>]).....	676
(boole <i>func int1 int2 ...</i>)	677
(boundp <i>atom</i>)	678
(car <i>list</i>)	678
(cdr <i>list</i>)	678
(caar <i>list</i>), (cadr <i>list</i>), (cddr <i>list</i>), (cadar <i>list</i>), itd.	679
(chr <i>number</i>)	679
(close <i>file-desc</i>)	680
(command [<i>arguments</i>] ...).....	680
(cond (<i>test1 result1 ...</i>) ...).....	682
(cons <i>new-first-element list</i>)	683
(cos <i>angle</i>).....	683
(cvunit <i>value from to</i>).....	683
(defun <i>sym argument-list expr ...</i>)	684
(distance <i>pt1 pt2</i>).....	685
(distof <i>string</i> [<i>tryb</i>])	685
(entdel <i>ename</i>).....	686
(entget <i>ename</i> [<i>applist</i>])	687
(entlast).....	690
(entmake [<i>elist</i>]).....	690
(entmod <i>elist</i>)	694
(entnext [<i>ename</i>]).....	695
(entsel [<i>prompt</i>])	696
(entupd <i>ename</i>).....	697
(eq <i>expr1 expr2</i>)	697
(equal <i>expr1 expr2</i> [<i>fuzz</i>]).....	698
(*error* <i>string</i>)	699
(eval <i>expr</i>)	699
(exit)	699
(exp <i>number</i>)	700
(expand <i>number</i>)	700
(expt <i>base power</i>)	700
(findfile <i>filename</i>).....	700
(fix <i>number</i>)	701
(float <i>number</i>).....	701
(foreach <i>name list expr...</i>).....	701
(gc)	702
(gcd <i>num1 num2</i>).....	702
(getangle [<i>pt</i>] [<i>prompt</i>]).....	702
(getcorner <i>pt</i> [<i>prompt</i>])	703
(getdist [<i>pt</i>] [<i>prompt</i>])	703
(getenv <i>variable-name</i>).....	704
(getfiled <i>title default ext falgs</i>).....	705
(getint [<i>prompt</i>])	707
(getkword [<i>prompt</i>])	708
(getorient [<i>pt</i>] [<i>prompt</i>]).....	708
(getpoint [<i>pt</i>] [<i>prompt</i>])	709
(getreal [<i>prompt</i>]).....	710

(getstring <i>[cr] [prompt]</i>).....	710
(getvar <i>varname</i>).....	711
(graphscr).....	711
(grclear).....	711
(grdraw <i>from to color [highlight]</i>).....	711
(grread <i>[track] [allkeys [curtype]]</i>).....	712
(grtext <i>[box text [highlight]]</i>).....	715
(grvecs <i>vlist [trans]</i>).....	716
(handent <i>handle</i>).....	717
(if <i>testexpr thenexpr [elseexpr]</i>).....	718
(initget <i>[bits] [string]</i>).....	718
Definiowanie słów kluczowych.....	720
(inters <i>pt1 pt2 pt3 pt4 [onseg]</i>).....	721
(itoa <i>int</i>).....	722
(lambda <i>arguments expr ...</i>).....	722
(last <i>list</i>).....	723
(length <i>list</i>).....	723
(list <i>expr ...</i>).....	723
(listp <i>item</i>).....	724
(load <i>filename [onfailure]</i>).....	724
(log <i>number</i>).....	725
(logand <i>number number ...</i>).....	725
(logior <i>integer ...</i>).....	726
(lsh <i>num1 numbits</i>).....	726
(mapcar <i>function list1 ... listn</i>).....	726
(max <i>number number ...</i>).....	727
(mem).....	728
(member <i>expr list</i>).....	728
(menucmd <i>string</i>).....	728
(min <i>number number ...</i>).....	729
(minusp <i>item</i>).....	729
(nentsel <i>[prompt]</i>).....	729
(nentselp <i>[prompt] [pt]</i>).....	731
(not <i>item</i>).....	732
(nth <i>n list</i>).....	732
(null <i>item</i>).....	733
(numberp <i>item</i>).....	733
(open <i>filename mode</i>).....	733
(or <i>expr ...</i>).....	734
(osnap <i>pt mode-string</i>).....	735
pi.....	735
(polar <i>pt angle distance</i>).....	735
(prin1 <i>[expr [file-desc]]</i>).....	735
(princ <i>[expr [file-desc]]</i>).....	737
(print <i>[expr [file-desc]]</i>).....	737
(progn <i>expr ...</i>).....	737
(prompt <i>msg</i>).....	737
(quit).....	738

(quote <i>expr</i>)	738
(read <i>string</i>)	738
(read-char [<i>file-desc</i>])	739
(read-line [<i>file-desc</i>])	739
(redraw [<i>ename</i> [<i>mode</i>]])	739
(regapp <i>application</i>)	740
(rem <i>num1 num2</i> ...)	741
(repeat <i>number expr</i> ...)	741
(reverse <i>list</i>)	742
(rtos <i>number</i> [<i>mode</i> [<i>precision</i>]])	742
(set <i>sym expr</i>)	742
(setq <i>sym1 expr1</i> [<i>sym2 expr2</i>] ...)	743
(setvar <i>varname value</i>)	744
(sin <i>angle</i>)	745
(sqrt <i>number</i>)	745
(ssadd [<i>ename</i> [<i>ss</i>]])	745
(ssdel <i>ename ss</i>)	746
(ssget [<i>mode</i>] [<i>pt1</i> [<i>pt2</i>]] [<i>pt-list</i>] [<i>filter-list</i>])	746
(sslenght <i>ss</i>)	751
(ssmemb <i>ename ss</i>)	751
(ssname <i>ss index</i>)	751
(strcase <i>string</i> [<i>which</i>])	751
(strcat <i>string1</i> [<i>string2</i>] ...)	752
(strlen [<i>string</i>] ...)	752
(subst <i>newitem olditem list</i>)	752
(substr <i>string start</i> [<i>length</i>])	753
(tablet <i>code</i> [<i>row1 row2 row3 direction</i>])	753
(tblnext <i>table-name</i> [<i>rewind</i>])	754
(tblsearch <i>table-name symbol</i> [<i>setnext</i>])	756
(terpri)	756
(textbox <i>elist</i>)	756
(textpage)	757
(textscr)	757
(trace <i>function</i> ...)	757
(trans <i>pt from to</i> [<i>disp</i>])	758
(type <i>item</i>)	760
(untrace <i>function</i> ...)	761
(ver)	761
(vmon)	761
(vports)	761
(wcmatch <i>string pattern</i>)	762
(while <i>testexpr expr</i> ...)	764
(write-char <i>code</i> [<i>file-desc</i>])	764
(write-line <i>string</i> [<i>file-desc</i>])	765
(xdroom <i>ename</i>)	765
(xdsizel <i>list</i>)	766
(xload <i>application</i> [<i>onfailure</i>])	767

(xunload <i>application [onfailure]</i>)	768
(zerop <i>item</i>)	768
Funkcje AutoLISPu zdefiniowane w systemie ADS	768
(acad_colordlg <i>colornum [flag]</i>)	769
(acad_helpdlg <i>helpfile [topic]</i>)	769
(acad_strlsort <i>list</i>)	770
Polecenia zdefiniowane w systemie ADS	771
(c:gkreskuj <i>pt [ss] [vector]</i>) (c:bhatch <i>pt [ss] [vector]</i>)	771
(c:gplinia <i>pt [ss] [vector]</i>) (c:bpoly <i>pt [ss] [vector]</i>)	772
(bherrs)	772
(c:psdynam <i>mode</i>) (c:psdrag <i>mode</i>)	773
(c:pswyp <i>ent pattern [arg1 [arg2]] ...</i>) (c:psfill <i>ent pattern [arg1 [arg2]] ...</i>)	773
(c:psin <i>filename position scale</i>) (c:pswe <i>filename position scale</i>)	774

Dodatek C

Kody ASCII	775
------------------	-----

Dodatek D

Kody i komunikaty błędów zwracane przez AutoLISP	781
Kody błędów	781
Komunikaty błędów	785
Błędy programów użytkowych	785
Błędy wewnętrzne	791

Dodatek E

Kody grup DXF	793
Kody grup w kolejności wartości liczbowych	793
Kody grup uporządkowane według rodzaju elementu rysunkowego	795
Kody grup elementów rysunkowych	796
Kody grup dla Bloków i tablic	803

Dodatek F

Zmienne systemowe AutoCADa	809
----------------------------------	-----

Dodatek G

Opis i instalacja programów znajdujących się na dyskietce	841
Instalacja programów	851

Literatura	853
------------------	-----