ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ 1. Клавишни комбинации в средата на Borland Pascal

Команди за преместване на курсора

Знак наляво	Ctrl+S или Лява стрелка
Знак надясно	Ctrl+D или Дясна стрелка
Дума наляво	Ctrl+A или Ctrl+Лява стрелка
Дума надясно	Ctrl+F или Ctrl+Дясна стрелка
Ред нагоре	Ctrl+E или Горна стрелка
Ред надолу	Ctrl+X или Долна стрелка
Екран нагоре	Ctrl+W
Екран надолу	Ctrl+Z
Страница нагоре	Ctrl+R или PgUp
Страница надолу	Ctrl+C или PgDn

Команди за вмъкване и изтриване

Режим вмъкване да/не	Ctrl+V или Ins
Вмъкване на ред	Ctrl+N
Изтриване на ред	Ctrl+Y
Изтриване до края на реда	Ctrl+Q Y
Изтрива знака вляво	Ctrl+H или Backspace
Изтрива текущия знак	Ctrl+G или Del
Изтрива дума	Ctrl+T

Команди за работа с блокове

Маркира начало на блок	Ctrl+K B
Маркира край на блок	Ctrl+K K
Маркира дума	Ctrl+K T
Копира блок	Ctrl+K C
Премества блок	Ctrl+K V
Изтрива блок	Ctrl+K Y
Чете файл от диска	Ctrl+K R
Записва блок на диска	Ctrl+K W
Скрива/показва маркиран блок	Ctrl+K H
Отпечатва блок на принтер	Ctrl+K P
Премества блок надясно	Ctrl+K I
Премества блок наляво	Ctrl+K U
Скок в началото на блока	Ctrl+Q B
Скок в края на блока	Ctrl+Q K
Преход към главното меню	Ctrl+K D
Маркира ред	Ctrl+K L
Копира блок в Клипборда	Ctrl+Ins
Премества блок в Клипборда	Shift+Del
Изтрива блок	Ctrl+Del
Извлича блок от Клипборда	Shift+Ins

Команди за маркиране с използване дясната част на клавиатурата (стрелки+цифри)

Знак вляво	Shift+Лява стрелка
Знак вдясно	Shift+Дясна стрелка
До края на реда	Shift+End
До началото на реда	Shift+Home
До същата позиция в долния ред	Shift+Долна стрелка
До същата позиция в горния ред	Shift+Горна стрелка
Страница надолу	Shift+PgDn
Страница нагоре	Shift+PgUp
Думата вляво	Shift+Ctrl+Лява стрелка
Думата вдясно	Shift+Ctrl+Дясна стрелка
До края на файла	Shift+Ctrl+End
До началото на файла	Shift+Ctrl+Home

Други команди

Преход към главното меню	F10
Запис на текущия файл	Ctrl+K S или F2
Отваряне на файла	F3
Затваря текущия прозорец	Alt+F3
Tab	Ctrl+I или Tab
Режим на табулация	Ctrl+O T
Автоматично позициониране на новия ред да/не	Ctrl+O I
Връща предишната редакция	Alt+BackSpace
Маркира позиция (0–9)	Ctrl+K *n* (*n* = 0..9)
Преход към маркирана позиция(0–9)	Ctrl+Q *n* (*n* = 0..9)
Преход към предишната позиция	Ctrl+Q P
Преход към съответстващата скоба	Ctrl+Q [
Контекстно–зависима помощ	Ctrl+F1
Вмъкване на контролен знак	Ctrl+P знака
Търсене	Ctrl+Q F
Търсене и замяна	Ctrl+Q A
Повтаря последното търсене	Ctrl+L
Прекъсване на операцията	Esc

Клавиши за работа с прозорците

Alt+0	Извежда списък с отворените прозорци
Alt+n	Прави активен прозорец с номер *n* (*n* = 1..9)
Alt+F3	Затваря текущия прозорец
Alt+F5	Извежда DOS – екрана (могат да се видят резултатите от програмата след приключването й и връщането в средата)
F5	Текущият прозорец изпълва целия екран и обратно
F6	Преминава към следващия прозорец
Shift+F6	Връща се към предишния активен прозорец
Ctrl+F5	Смяна размера и положението на текущия прозорец

Клавиши за работа с програмата

Alt+F9	Компилира текущата програма
Ctrl+F2	Прекъсва трасирането на текущата програма
Ctrl+F3	Показва прозорец със състоянието на стека
Ctrl+F4	Изчислява израз, показва и променя стойност на променлива
Ctrl+F7	Добавя променлива към прозореца с текущи стойности на променливи
Ctrl+F8	Поставя, или премахва точка на прекъсване
Ctrl+F9	Изпълнява програмата
F4	Изпълнява програмата до позицията на курсора
F7	Изпълнява само текущия ред и ако той съдържа обръщение към процедура, или функция започва постъпково да ги изпълнява
F8	Изпълнява само текущия ред и ако той съдържа обръщение към процедура, или функция ги изпълнява без да влиза в тях
F9	Компилира програмата, намираща се в Главния файл (Primary file). Ако няма такъв, компилира текущата програма.

Клавиши за работа с менютата

Alt+Spacebar	Управляваща кутия (System)
Alt+C	Compile
Alt+D	Debug
Alt+E	Edit
Alt+F	File
Alt+H	Help
Alt+O	Options
Alt+R	Run
Alt+S	Search
Alt+W	Window
Alt+X	Изход в DOS
F10	Преход към Главното меню

Клавиши за работа с помощта (Help)

F1	Извиква помощта
Shift+F1	Показва съдържанието на помощта в азбучен ред
Ctrl+F1	Показва контекстно-зависима помощ
Alt+F1	Показва съдържанието на помощта, използвана последно

ПРИЛОЖЕНИЕ 2. Съобщения за грешки

Съобщения за грешки при компилиране

По-долу е даден списък на възможните съобщения за грешки от компилатора по време на компилиране на програма. Винаги, когато е възможно, компилаторът дава допълнителна информация във форма на идентификатор или име на файл. Например: **Error 15:File not found (window.tpu).**

1. Out of memory

Тази грешка се появява, когато за компилатора няма свободна памет.

2. Identifier expected

На това място трябва да има идентификатор. Може би се опитвате да предекларирате резервирана дума.

3. Unknown identifier

Идентификаторът не е деклариран.

4. Duplicate identifier

Идентификаторът вече е използван в текущия блок.

5. Syntax error

В текста на програмата е намерен невалиден знак.

6. Error in real constant

Синтаксисът на константи от тип real е неправилен.

7. Error in integer constant

Синтаксисът на константи от тип **integer** е неправилен.

8. String constant exceeds line

Най-вероятно е да сте забравили затварящ апостроф в константа от тип знаков низ.

9. Too many nested files

Много вложени файлове.

10. Unexpected end of file

Това съобщение за грешка се получава, ако:

– текстът на програмата завършва преди последното **end** на главната част. Най-вероятно е **begin** и **end** да са неуравновесени.

– някой коментар не е затворен.

11. Line top long

Максималната дължина на ред е 127 знака.

12. Type identifier expected

Идентификаторът не обозначава тип, както се очаква.

13. Too many open files.

Файлът **CONFIG.SYS** не включва ред **FILES = xx** или този ред определя прекалено малко файлове. Увеличете броя до някаква подходяща стойност, например 20.

14. Invalid file name

Името на файла е неправилно или съдържа несъществуващ път на достъп.

15. File not found

Файлът не може да се намери в текущия справочник или, в който и да е справочник, подходяща за типа на такъв файл.

16. Disk full

Дискът е пълен. Изтрийте някои файлове или използвайте нова дискета.

17. Invalid compiler directive

Директивата за компилатора е непозната, един от параметрите на директивата за компилатора е неправилен или използвате глобална директива за компилатора, когато вече е започната компилация на тялото на програмата.

18. Too many files

Има прекалено много файлове, включени в компилацията на програмата или **unit.**

19. Undefined type in pointer definition

В декларация на тип **pointer** участвува недекларирaн тип.

20. Variable identifier expected

Идентификаторът не обозначава променлива, както би трябвало.

21. Error in type

С този знак не може да започва дефиниция на тип

22. Structure too large

Максималният допустим размер на структуриран тип е 65520 байта.

23. Set base type out of range

Базовият тип на множество трябва да бъде диапазон с граници 0..255 или изброим тип с не повече от 256 възможни стойности.

24. File components may not be files or objects

Компонентите на файла не могат да бъдат файлове или обекти. Като тип на елементите във файла не може да се задават обекти или файлове. Не може да се задава и структурен тип с обектна или файлова компонента.

25. Invalid string length

Декларираната максимална дължина на знаков низ трябва да бъде в диапазона 0..255.

26. Type mismatch

– несъвместими типове отляво и отдясно на оператор на присвояване.

– несъвместими типове на фактическия и формалния параметър при активиране на процедура или функция.

– несъвместимост на типа на израза с типа на индекса в индексиран масив.

– несъвместими типове на операнди в израз.

27. Invalid subrange base type

Правилни базови типове са само изброимите типове.

28. Lower bound greater than upper bound

Декларацията на тип подобласт определя долна граница, която по-голяма от горната граница.

29. Ordinal type expected

Очаква се дискретен тип.

30. Integer constant expected

Очаква се константа от целочислен тип.

31. Constant expected

Очаква се константа.

32. Integer or real constant expected

Очаква се константа от целочислен или реален тип.

33. Pointer type identifier expected

Идентификаторът не обозначава тип указател, както се очаква.

34. Invalid function result type

Правилните типове на резултата на функция са прости, низ и указател.

35. Label identifier expected

Идентификаторът не обозначава етикет, както се очаква.

36. BEGIN expected

Очаква се **begin.**

37. END expected

Очаква се **end.**

38. Integer expression expected

Очаква се израз от целочислен тип.

39. Ordinal expression expected

Очаква се израз от изброим тип.

40. Boolean expression expected

Очаква се израз от логически тип.

41. Operand types do not match operator

Тази операция не допуска операнди от такъв тип.

42. Error in expression

Знакът не може да участва в израз по този начин. Може би сте забравили да напишете знак за операция между два операнда.

43. Illegal assignment

– На файлове и безтипови променливи не могат да бъдат присвоявани стойности.

– На идентификатор на функция могат да бъдат присвоявани стойности само в дефиницията на функцията.

44. Field identifier expected

Идентификаторът не обозначава поле в променлива от тип запис.

45. Object file too large

Обектният файл е недопустимо дълъг.

46. Undefined external

Неопределен външен знак.

47. Invalid object file record

Файлът **.OBJ** съдържа неправилен обектен запис: уверете се, че файлът действително е **.OBJ**.

48. Code segment too large

Максималният размер на кода на програмата или на **unit** е 65520 байта.

Ако компилирате програма, преместете някои процедури или функции в **unit**. Ако компилирате unit, разделете го на два или повече.

49. Data segment too large

Максималният размер на сегмента с данни на програмата е 65520 байта, включително данни, декларирани от използвани модули. Ако се нуждаете от повече глобални данни, декларирайте по-големите структури като указатели и ги назначете динамично чрез използване на процедурата **New.**

50. Do expected

Очаква се **do.**

51. Invalid PUBLIC definition

– Идентификаторът е станал общодостъпен чрез директива **PUBLIC** на Асемблер, но няма съответстваща външна декларация в програма или модул на БП.

– Две или повече директиви **PUBLIC** на Асемблер дефинират един и същ идентификатор.

– Файлът **.OBJ** дефинира символи **PUBLIC**, които не се намират в кодовия сегмент.

52. Invalid EXTRN definition

– Към този идентификатор е имало препратка чрез директива **EXTERN** на Асемблер, но той не е деклариран в програма или unit на Pascal, нито в частта interface на използваните модули.

– Идентификаторът обозначава абсолютна **(absolute)** променлива.

– Идентификаторът обозначава процедура или функция **inline.**

53. Too many EXTERN definitions

БП не може да обработва файлове **.OBJ** с повече от 256 дефиниции **EXTERN**.

54. OF expected

Очаква се **of**.

55. INTERFACE expected

Очаква се **interface.**

56. Invalid relocatable reference

– Файлът **.OBJ** съдържа данни и преразпределени препратки в сегменти, различни от кодовия. Опитвате се, например, да декларирате инициализирани променливи в сегмента за данни.

– Файлът **.OBJ** съдържа побитови препратки към преразпределени символи. Тази грешка се появява, ако използвате операторите **HIGH** и **LOW** с преразпределени символи или ако имате препратки към преразпределени символи в директиви **DB**.

–Операнд отпраща към преразпределен символ, който не е дефиниран в кодовия сегмент или в сегмента за данни.

– Операнд отпраща към процедура или функция **EXTRN** с отместване, например: **CALL Sort Proc +8.**

57. THEN expected

Очаква се **then.**

58. TO or DOWNTO expected

Очаква се **to** или **downto.**

59. Undefined forward

– Процедурата или функцията е декларирана в секция **interface** на **unit**, но нейната дефиниция липсва в секция **implementation**.

– Процедурата или функцията е декларирана с **forward**, но нейната дефиниция не е намерена.

60. Too many procedures

БП не позволява повече от 512 процедури или функции на модул. Ако компилирате програма, преместете някои от процедурите или функциите в модул. Ако компилирате модул, разделете го на два или повече.

61. Invalid typecast

– Размерите на обръщение към променлива и на получавания тип при замяна на променливи **(variable typecast)** се различават.

– Опитвате се да замените тип на израз, в който е позволено само обръщение към променлива.

62. Division by zero

Опит за деление на нула.

63. Invalid file type

Типът на файла не се поддържа от процедурата, обработваща файла:
например: **read** с типов файл или **seek** с текстов файл.

64. Cannot Read or Write variables of this type

–**Read** и **Readln** могат да въвеждат само променливи от тип **char**, **integer, real** и **string**.

–**Write** и **Writeln** могат да извеждат само променливи от тип **char, integer, real, string** и **boolean.**

65. Pointer variable expected

Очаква се променлива от тип **pointer**.

66. String variable expected

Очаква се променлива от низов тип.

67. String expression expected

Очаква се израз от низов тип.

68. Unit not found

Един или повече модули, използвани от този модул, не са обозначени в клаузата **uses**.

69. Unit name mismatch

Името на модул във файла **.TPU** не съответства на името, определено в клаузата **uses**.

70. Unit version mismatch

Един или повече модули, използвани от този модул, са променени след компилирането на модула. Използвайте опциите **Compile/Make** или **Compile/Build** в интегрираната работна среда или опциите **/M** или **/B** в командния ред на компилатора, за да компилирате отново модула (ако е нужно).

71. Duplicate unit name

Вече сте наименували този модул в клаузата **uses**.

72. Unit file format error

Файлът **.TPU** е недействителен: уверете се, че това наистина е файл **.TPU**.

73. Implementation expected

Очаква се секция **implementation** на модула.

74. Constant and case types do not match

Типът на селектора на **case** оператор е несъвместим с константите за избор.

75. Record or object variable expected

Очаква се променлива от тип запис или обект.

76. Constant out of range

– Опитвате се да индексирате масив с константа, която е извън разрешения диапазон от стойности.

– Опитвате се да присвоите на променлива константа, която се намира извън разрешения диапазон от стойности.

– Опитвате се да дадете като параметър на процедура или функция константа, която е извън разрешения диапазон от стойности.

77. File variable expected

Очаква се променлива от тип файл.

78. Pointer expression expected

Очаква се променлива от тип указател.

79. Integer or real expression expected

Очаква се променлива от тип **integer** или **real**.

80. Label not within current block

Оператор **goto** не може да съдържа етикет, който е извън текущия блок.

81. Label already defined

Етикетът вече е маркирал оператор.

82. Undefined label in preceding statement part

Обръщение към недекларирaн етикет.

83. Invalid @ argument

Валидни аргументи са обръщения към променливи и идентификатори на процедури или функции.

84. UNIT expected

Очаква се **unit.**

85 ";" expected

Очаква се „;“.

86. ":" expected

Очаква се „:“.

87. "." expected

Очаква се „.“.

88. "(" expected

Очаква се „(“.

89. ")" expected

Очаква се „)“.

90. "=" expected

Очаква се „=“.

91. ":=" expected

Очаква се „:=“.

92. "[" or "(." expected

Очаква се „[“ или „(“.

93. "]" or ".)" expected

Очаква се „]“ или „.)“.

94. "." expected

Очаква се „.“.

95. ".." expected

Очаква се „..“.

96. Too many variables.

– Общият размер на глобалните променливи, декларирани в програма или модул, не може да превишава 64К.

– Общият размер на локалните променливи, декларирани в процедура или функция, не може да превишава 64 КВ.

97. Invalid FOR control variable

Управляващата променлива на оператор **for** трябва да бъде от прост дискретен тип и декларирана в текущата подпрограма.

98. Integer variable expected

Очаква се променлива от тип **integer**.

99. Files are not allowed here

Не са разрешени файлови или процедурни типове.

100. String length mismatch

Дължината на низовата константа не съответства на броя на компонентите в знаковия масив.

101. Invalid ordering of fields

Полетата на константата от тип **record** трябва да бъдат написани в порядъка на декларацията.

102. String constant expected

Очаква се константа от тип низ.

103. Integer or real variable expected

Очаква се променлива от тип **integer** или **real.**

104. Ordinal variable expected

Очаква се променлива от изброим тип.

105. INLINE error

Операторът **<** не е позволен заедно с **relocatable** обръщения към променливи. Тези обръщения винаги трябва да бъдат във **word** размер.

106. Character expression expected

Очаква се израз от знаков тип.

107. Too many relocation items

Размерът на таблицата за преадресация в **.EXE** файла достига 64K, което е горната граница за БП. Тази грешка означава, че програмата е прекалено голяма за обработка от свързващия редактор на БП. Може би програмата е прекалено голяма и за изпълнение от **DOS**. Ще трябва да разделите програмата на главна част, която изпълнява две или повече подпрограми, използувайки процедурата **Exec** в **DOS** модула.

108. Overflow in arithmetic operation

Препълване при аритметична операция.

109. No enclosing For, While or Repeat statement

Break и Continue не могат да се използват извън цикъл.

112. CASE constant out of range

Константата в **CASE** е извън допустимия интервал. За **CASE** оператор със селектор от целочислен тип константите трябва да бъдат в диапазона –32768..32767.

113. Error in statement

Грешка в оператора. Може би преди else е записана „;“.

114. Cannot call an interrupt procedure

Не можете директно да извикате процедура за прекъсване.

116. Must be in 8087 mode to compile this

Тази конструкция може да бъде компилирана само в режим **{$N+}.** В режима **{$N–}** не са позволени операции с 8087 реални типове **single, double, extended** и **comp**.

117. Target address not found

При режим **Search/Find** командата за търсене на грешки в интегрираната работна среда или опцията **/F** във версията с команден ред не може да намери оператор, съответствуващ на посочения адрес.

118. Include files are not allowed here

Всеки блок **begin–end** трябва да бъде изцяло в един файл.

121. Invalid qualifier

– Опитвате се да индексирате променлива, която не е от тип масив.

– Опитвате се да посочите полета в променлива, която не е от тип запис.

– Опитвате се да осъществите обръщение към променлива чрез променлива от тип указател, която не е декларирана.

122. Invalid variable reference

Конструкцията има синтаксис на обръщение към променлива, но не означава адрес в паметта. Опитвате се най-вероятно, да извикате функция от тип указател, но сте забравили да напишете ^.

123. Too many symbols

Програмата или модулът декларира повече от 64K символи. Ако компилирате с **{$D+},** опитайте да го изключите, въпреки че това няма да позволи намирането на грешки по време на изпълнение на този модул. Може да опитате и някои декларации в отделен **unit.**

124. Statement part too large

БП ограничава всеки блок **begin-end** до 24k. Ако срещнете тази грешка, преместете секции от блока **begin–end** в една или повече процедури.

126. Files must be var parameters

Опитвате се да декларирате параметър-стойност от тип файл. Параметрите от тип файл трябва да бъдат параметри-променливи.

127. Too many conditional symbols

Няма достатъчно място за деклариране на други условни символи. Опитайте се да премахнете някои символи или да съкратите част от имената на символи.

128. Misplaced conditional directive

Компилаторът е срещнал директива **{$ELSE}** или **{$ENDIF}** без съответстващa директива **{$IFDEF}**, **{$IFNDEF}** или **{$IFOPT}**.

129. FNDIF directive missing

В програмата трябва да има равен брой директиви **{$IFxxx}** и **{$ENDIF}**.

130. Error in initial conditional defines

Инициализираните числови символи, определени с опция **D/C/Conditional defines** или в справочник **/D** са невалидни. БП очаква нула или повече идентификатори, разделени с интервали, запетаи или точка и запетая.

131. Header does not match previous definition

– Несъответствие със заглавието на процедура или функция, определено в секция **interface**.

– Несъответствие със заглавието на процедура или функция, определено в декларацията **forward.**

133. Cannot evaluate this expression

Невъзможно е да се изчисли този израз.

134. Expression incorrectly terminated

Изразът завършва неправилно.

135. Invalid format specifier

Недопустим форматиращ знак.

136. Invalid indirect reference

Недопустим косвен указател.

137. Structured variables are not allowed here

Тук не може да се използуват променливи от структуриран тип.

138. Cannot evaluate System unit

Изразът не може да бъде изчислен чрез модул **SYSTEM.**

139. Cannot access this symbol

Няма достъп до този символ.

140. Invalid floating–point operation

Неправилна операция с реални операнди.

141. Cannot compile overlays to memory

Не може да се извърши в паметта компилация на овърлейни модули.

142. Procedure or function variable expected

Очаква се процедура или функция.

143. Invalid procedure or function reference

Недопустим указател към процедура или функция.

144. Cannot overlay this unit

Невъзможно е препокриване на този модул.

145. Too many nested scopes

Прекалено много вложени обхвати.

146. File access denied

Достъп до файла е отказан от ОС.

147. Object type expected

Тук се очаква обектен тип, но идентификаторът на променливата не е от такъв тип.

148. Local object types are not allowed

Не са допустими локални обектни типове. Обектните типове могат да бъдат дефинирани само в най-външното равнище на програмата или **unit**. Не е допустимо дефинирането на обектен тип вътре в процедури и функции.

149. VIRTUAL expected

Липсва резервирана дума **virtual**.

150. Method identifier expected

Очаква се идентификатор на метод. Зададеният идентификатор не е метод.

151. Virtual **constructors are not allowed**

Не са допустими виртуални конструктори.

152. Constructor identifier expected

Очаква се идентификатор на конструктор.

153. Destructor identifier expected

Зададеният идентификатор не е деструктор.

154. Fail only allowed within constructors

Стандартната процедура **Fail** може да се използува само вътре в конструктор.

155. Invalid combination of opcode and operands

Недопустима комбинация от код на операция и операнди.

156. Memory reference expected

Липсва адрес.

157. Cannot add or subtract relocatable symbols

Не могат да се събират или изваждат премествaеми символи.

158. Invalid register combination

Недопустима регистрова комбинация.

159. 286/287 instructions are not enabled

Не е достъпен наборът от команди на микропроцесорите 286/287.

160. Invalid symbol reference

Недопустим указател към символ.

161. Code generation error

Грешка при генериране на код.

162. ASM expected

Очаква се думата **ASM**.

163. Duplicate dynamic method index

Дублиране при динамичен индексен метод.

164. Duplicate resource identifier

Опит за използване на ресурс, който вече е използван от друг ресурс.

166. Procedure or function identifier expected

Очаква се идентификатор на процедура или функция.

167. Cannot export this symbol

ПП не може да бъде експортирана, освен ако не е декларирана с директива **export**.

168. Duplicate export name

Името е използвано от друга експортираща ПП.

169. Executable file header too large

Изпълнимият **.EXE** файл надхвърля 64K.

Съобщения за грешки при изпълнение

Някои грешки по време на изпълнение на програма предизвикват съобщение за грешка и водят до прекратяване ѝ: **Runtime error nnn at xxxx:yyyy**, където **nnn** е номера на грешката при изпълнение**,** а **xxxx:yyyy** е нейният адрес (сегмент и отместване). Грешките по време на изпълнение се делят на четири категории: грешки открити от **DOS (I/O** с кодове на грешки 1...99), входно–изходни грешки (с кодове на

грешки 100...149), критични грешки (с кодове на грешки 150...199), фатални грешки (с кодове на грешки 200...255).

Грешки открити от DOS

1. Invalid function number

Грешен номер на функция (Опитвате се да извикате несъществуваща функция на **DOS**).

2. File not found

Файлът не е намерен. Съобщава се от **Reset, Append, Rename** или **Erase,** когато името, присвоено на променлива от тип файл, не определя съществуващ файл.

3. Path not found

Пътят до файла не е намерен.

4. Too many open files

Съобщението идва от **Reset, Append** или **Rewrite**, когато в програмата има прекалено много отворени файлове. Увеличете възможността за повече отворени файлове в **CONFIG.SYS** с **FILES = xx**.

5. File access denied

– Съобщението идва от **Reset** или **Append** ако **FileMode** позволява записване, но името на файла, присвоено на файловата променлива, определя справочник или файл, който е само за четене **(read–only)**.

– Съобщението идва от **Rewrite**, когато справочникът е запълнен или името, присвоено на файловата променлива, специфицира справочник или файл, който е само за четене.

– Съобщението идва от **Rename**, когато името, присвоено на файловата променлива, определя справочник или новото име определя съществуващ файл.

– Съобщението идва от **Erase,** когато името, присвоено на файловата променлива, определя справочник или файл, който е само за четене.

– Съобщението идва от **MkDir,** когато в родителския справочник съществува файл със същото име, няма място или ако пътят за достъп задава устройството.

– Съобщението идва от **RmDir**, когато справочникът не е празен, пътят за достъп не задава справочник или е посочен главният справочник.

– Съобщението идва от **Read** или **BlockRead** на типов или безтипов файл, когато той не е отворен за четене.

– Съобщението идва от **Write** или **BlockWrite** на типов или безтипов файл, когато файлът не е отворен за записване.

6. Invalid file handle

Тази грешка се получава при невалиден манипулатор на файл (**handle)** при повикване от **DOS** и означава, че файловата променлива по някакъв начин е „орязана“.

12. Invalid file access code

Недопустим код за достъп до файл. Съобщението идва от **Reset** или **Append** на типов или безтипов файл, когато стойността на **FileMode** е невалидна.

15. Invalid drive number

Съобщението идва от **GetDir**, когато номерът на устройството е невалиден.

16. Cannot remove current directory

Съобщението идва от **RmDir**, когато пътят на достъп определя текущия справочник.

17. Cannot rename across drives

Съобщението идва от **Rename**, когато двете имена не са в едно и също дисково устройство.

Входно–изходни грешки

100. Disk read error

Грешка при четене от диск. Съобщението идва от **Read** на типов файл, когато се опитвате да четете след края на файла.

101. Disk write error

Грешка при запис върху диск. Съобщението идва от **Close, Write, Writeln, Flush** или **Page**, когато дискетата се запълни.

102. File not assigned

Съобщението идва от **Reset, Rewrite, Append, Rename** или **Erase,** когато на файловата променлива не е присвоено име на файл чрез активиране на **Assign**.

103. File not open

Съобщението идва от **Close, Read, Write, Seek, Eof, FilePos, FileSize, Flush, BlockRead** или **BlockWrite**, когато файлът не е отворен.

104. File not open for input

Съобщението идва от **Read, Readln, Eoln, SeekEof** или **SeekEoln** на текстов файл, когато файлът не е отворен за вход.

105. File not open for output

Съобщението идва от **Write, Writeln** и **Page** на текстов файл, когато файлът не е отворен за изход.

106. Invalid numeric format

Съобщението идва от **Read** или **Readln**, когато числовата стойност, прочетена от текстов файл, не съответствува на правилния числов формат.

Критични грешки

150. Disk is write protected

Дискът е защитен срещу запис.

152. Drive not ready

Дисковото устройство не е готово.

154. CRC error in data

Грешка в изходните данни.

156. Disk seek error

Грешка при търсене на писта върху диска.

157. Unknown media type

Непознат тип на носител.

158. Sector Not Found

Не е намерен сектор.

159. Printer out of paper

Принтерът няма хартия.

160. Device write fault

Грешка при запис на устройството.

161. Device read fault

Грешка при четене от устройството.

162. Hardware failure

Повреда в апаратурата на компютъра.

Фатални грешки

Тези грешки незабавно прекъсват изпълнението на програмата.

200. Division by zero

Делене на нула.

201. Range check error

Съобщението за тази грешка идва от оператори, компилирани в режим **{$R+}** и при едно от следните положения:

– индексният израз на масив не е бил в границите на разрешения диапазон.

– направен е опит за присвояване на стойност, намираща се извън диапазона на променлива.

– направен е опит като фактически параметър на процедура или функция да се даде стойност, намираща се извън разрешения диапазон.

202. Stack overflow error

Съобщението се получава на входа на процедура или функция, компилирана в режим **{$S+}**, когато в стека няма достатъчно място за разпределение на локалните променливи на подпрограмата. Увеличете размера на стека чрез директивата за компилатора **$M.**

203. Heap overflow error

Съобщението се получава от **New** или **GetMem**, когато в ОП няма достатъчно свободно място за разпределение на блок с искания размер.

204. Invalid pointer operation

Съобщението за грешка се получава от **Dispose** или **FreeMem**, когато указателят е **nil,** показва адрес извън **heap** или свободния списък **(free list)** не може да бъде разширен.

205. Floating point overflow

Операция с плаваща запетая е предизвикала препълване.

206. Floating point underflow

Операция с плаваща запетая е предизвикала изчезване на порядъка (получаване на машинна нула). За тази грешка се съобщава само когато използувате числов копроцесор 8087 с командна дума (команда), която показва случаите на изчезване на разряди. По подразбиране изчезването на разряди предизвиква връщане на нула.

207. Invalid floating point operation

– Реалната стойност, предадена на **Trunc** или **Round** не може да бъде превърната в цяло число в диапазон **longint** (–2147483648..+2147483647).

– Аргументът на функцията **Sqrt** е бил отрицателен.

– Аргументът на функцията **Ln** е бил отрицателен или равен на нула.

– Препълване на стека на 8087.

208. Overlay manager not installed

Не е установено управление на препокриването.

209. Overlay file read error

Грешка при четене на овърлеен файл.

210. Object not initialized

Обектът не е инициализиран при **{R+}**. Извикан е виртуален метод на обект, макар че този обект още не е инициализиран (с помощта на извикване на конструктор).

211. Call to abstract method

Извикване на абстрактен метод.

212. Stream registration error

Грешка при регистрация.

213. Collection index out of range

Събирателният индекс е извън допустимите граници.

214. Collection overflow error

Събирателна грешка от препълване.

215. Arithmetic overflow error

Аритметична грешка, водеща до препълване.

216. General protection fault

Проблем при работа в защитен режим

.

ПРИЛОЖЕНИЕ 3. Някои стандартни ПП и служебни думи, използвани във втора част:

inc(x {,n}) – увеличава стойността на ***x*** (параметър променлива) с ***n***. Равносилно е на **x:=x+n**. Ако ***n*** липсва, то ***x*** се увеличава с 1: **inc(x)** е равносилно на **x:=x+1**.
Параметърът променлива ***x*** е от дискретен тип, а ***n*** – целочислен израз.

dec(x {,n}) – намалява стойността на ***x*** (параметър променлива) с ***n***. Равносилно е на **x:=x–n**. Ако ***n*** липсва, то ***x*** се намалява с 1: **dec (x)** е равносилно на **x:=x–1**.
Параметърът променлива ***x*** е от дискретен тип, а ***n*** – целочислен израз.
Двете процедури генерират оптимален код и са полезни при цикли.

exit – излиза незабавно от текущия блок. Ако това е главната програма, преустановява действието ѝ.

Вградени процедури и функции в модула Crt	
AssignCrt	Процедура, асоцииращa текстов файл с CRT прозореца.
ClrEol	Процедура, която изчиства знаковете от позицията на курсора до края на реда.
ClrScr	Процедура, която изчиства екрана и оставя курсора в горния ляв ъгъл на екрана.
Delay	Процедура, която прави пауза с дължина въведено число в милисекунди.
DelLine	Процедура, която изтрива реда, съдържащ курсора.
GotoXY	Процедура, която мести курсора в точка с дадени координати вътре във действителния екран.
HighVideo	Процедура, която избира увеличена яркост на знаковете.
InsLine	Процедура, която вмъква празен ред в позицията на курсора.
KeyPressed	Функция, определяща дали е бил натиснат клавиш от клавиатурата.
LowVideo	Процедура, която избира ниска яркост на знаковете.
NormVideo	Процедура, която избира оригиналните текстови атрибути и им дава стойности, каквито са имали при началното стартиране на програмата.
NoSound	Процедура, изключваща вътрешния говорител на компютъра.
ReadKey	Функция, която чете знак, въведен от клавиатурата.
Sound	Процедура, стартираща вътрешния говорител на компютъра.
TextBackground	Процедура, която избира цвета на фона.
TextColor	Процедура, която избира цвета на буквите.
TextMode	Процедура, която избира определен текстов режим.
WhereX	Функция, която връща *X* координатата на текущата позиция на курсора.
WhereY	Функция, която връща *Y* координатата на текущата позиция на курсора.
Window	Процедура, която дефинира текстов прозорец на екрана.