

Приложение Е. Диагностические сообщения

При использовании UniPascal-я возможно, что программа составлена с ошибками. Ошибки можно обособить в следующие классы:

- предупреждения в процессе компиляции;
- ошибки, обнаруживаемые операционной системой, выполняющей некоторое действие по заказу программы (например, попытка обращения к несуществующему файлу);
- ошибки, обнаруживаемые в процессе выполнения программы, вызываемые другими причинами (например, недостаточный объем памяти);
- другие ошибки, обнаруживаемые в процессе выполнения программы;
- ошибки, дефинированные стандартом ISO и необнаруживаемые системой UniPascal;
- алгоритмические ошибки.

Предупреждения выдаются только компилятором. В некоторых случаях они являются предупреждением о возможной ошибке, а в других о существующих ошибках, которые не оказывают влияние на процесс компиляции (например, использование несуществующей директивы для управления работой компилятора).

Следует особо отметить, что язык программирования UniPascal, реализованный описываемым компилятором, является расширением стандарта ISO Pascal-я и некоторые ошибки, заданные стандартом, являются вполне приемлемыми конструкциями. Такие ошибки могут появиться и если Вы попытаетесь использовать программу другой реализацией языка Pascal. Например, не все реализации поддерживают тип STRING.

Алгоритмические ошибки при каких обстоятельствах нельзя обнаружить автоматическим способом. В этом случае программа выдает неправильный результат (т.е. не так, как ожидалось) или вообще не останавливается и т.д.

Е.1. Выдаваемые компилятором сообщения об ошибках

При обнаружении ошибки компилятор выводит номер и содержание строки, в которой обнаружена ошибка, следующая строка с указателем ^ (стрелка вверх) на точное место ошибки и строка с диагностическим сообщением. Если на диске, где записан компилятор, не обнаружен файл **UNIPAS.ERR**, выводится только номер ошибки (без текста диагностического сообщения).

Ошибки, обнаруживаемые в процессе компиляции можно разделить на три группы:

- ошибки, которые не влияют на процесс компиляции - при обнаружении таких ошибок компилятор выводит предупредительное сообщение и продолжает свою работу, генерируя код и после места ошибки. Ошибки этой группы имеют номера с 1 по 13;
- нефатальные ошибки - генерация кода прекращается, но компиляция продолжается в виде анализа исходного текста с целью обнаружения других ошибок. Ошибки этой группы имеют номера с 14 по 239;
- фатальные ошибки - после обнаружения такой ошибки компиляция прекращается. Ошибки этой группы имеют номера с 240 по 255.

Если когда нибудь во время компиляции получите диагностическое сообщение с номером ошибки вне интервала [1..255] настойчиво просим выслать в адрес НИПЛА "Програмно осигуряване" программный текст, в процессе компиляции которого получено такое сообщение.

Далее в тексте приведены номера ошибок, сообщение на русском языке и, если причина не очевидна, соответствующие пояснения.

Е.1.1. Предупредительные сообщения компилятора

- 1: Директива управления компиляцией должна полностью помещаться в одной строке исходного текста
- 2: Несуществующая директива управления компиляцией
- 3: Ошибка в директиве управления компиляцией. Например, {\$M*} вместо {\$M+}
- 5: Вероятное использование переменной ссылочного типа (но не обнаружен ^)
- 7: Базовой тип указателя совпадает с собою, т.е. указатель указывает на себя.
- 8: Использована процедура FillChar или FillWord и заданная длина превышает длину заполняемого участка.
- 10: Стандартной функции ORD подано целое число в качестве параметра. Например, ORD(0) вместо ORD('0').
- 11: Дата и/или час установлены в операционной системе. Время очень важно при компиляции описательной части модуля (см. п. 12.6).
- 12: Длина экспортируемого ассемблером идентификатора превышает 8 символов. Используются только первые 8 символов.
- 13: Неправильный символ в тексте программы. Вероятнее всего найден символ с ASCII-кодом меньше \$20 или больше \$7F. Символ игнорируется.

Е.1.2. Нефатальные ошибки

- 14: Ожидается (но не найдена) цифра.
- 15: Ожидается (но не найдено) целое число в диапазоне [0..255].
- 16: Ожидается (но не найдено) целое число.
- 17: Ожидается (но не найдено) "program"
- 18: Ожидается (но не найдено) "unit"
- 19: Ожидается (но не найдено) "interface"
- 20: Ожидается (но не найдено) "implementation"
- 21: Ожидается (но не найдено) "begin"
- 22: Ожидается (но не найдено) "do"
- 23: Ожидается (но не найдено) "to" или "downto"
- 24: Ожидается (но не найдено) "end"
- 25: Ожидается (но не найдено) "of"
- 26: Ожидается (но не найдено) "then"
- 27: Ожидается (но не найдено) "to"
- 28: Ожидается (но не найдено) "until"
- 29: Ожидается (но не найдено) "procedure" или "function"
- 30: Ожидается (но не найдено) "case"

- 31: Ожидается (но не найдено) "for"
- 32: Ошибка в целой или вещественной константе
- 33: Ожидается (но не найдено) "if"
- 34: Ожидается (но не найдено) "while"
- 35: Ожидается (но не найдено) "with"
- 36: Ожидается (но не найдена) метка (идентификатор или число)
- 37: Ожидается (но не найден) символ ":" или ","
- 38: Ожидается (но не найдено) "begin" или декларация (var, type и т.д.)
- 39: Ожидается (но не найден) символ "="
- 40: Ожидается (но не найден) символ "=="
- 41: Ожидается (но не найден) символ ".."
- 42: Ожидается (но не найден) символ "."
- 43: Ожидается (но не найден) символ "["
- 44: Ожидается (но не найден) символ "]"
- 45: Ожидается (но не найден) символ "("
- 46: Ожидается (но не найден) символ ")"
- 47: Ожидается (но не найден) символ ";"
- 48: Ожидается (но не найден) символ ":"
- 49: Ожидается (но не найден) символ ","
- 50: Ожидается (но не найден) идентификатор
- 51: Ошибка в части определения (в декларативной части)
- 52: Попытка повторного определения идентификатора
- 53: Ошибка при определении типа
- 54: Ошибка в простом типе (simple type)
- 55: Ожидается (но не найдена) целая константа в диапазоне [1..255] для определения длины строки символов
- 56: Границы диапазонного типа должны быть простого типа
- 57: Нижняя граница диапазонного типа больше верхней границы
- 58: Несовместимые типы границ
- 59: Неопределенная forward-процедура или функция
- 60: Неразрешенная ссылка на forward-тип. Вероятнее всего попытка использовать в качестве базы ссылочного типа тип, который не описан в том-же самом разделе описания типов
- 61: Этот тип не может быть использован в качестве индексного типа
- 62: Задание файлов здесь не разрешено. Вероятнее всего попытка задать файл в качестве элемента массива или записи
- 63: Предыдущее определение не типа forward. Попытка применить в качестве идентификатора процедуры или функции уже определенный идентификатор
- 64: Ошибка при определении типа forward-заданной функции
- 65: Тип функции должен быть простым, диапазонным или ссылочным (в частности не разрешены массивы, записи или строки)
- 66: При определении функции не указан тип результата
- 67: Попытка повторного определения интерфейсной процедуры
- 68: Попытка повторного forward-определения процедуры
- 69: Процедуры типа Inline не могут иметь опережающее описание (forward)
- 70: Процедуры типа Inline не могут быть оверлейнами

- 71: Заголовок (процедуры или функции) не соответствует ранее данному определению
- 72: Ошибка в списке параметров
- 73: В качестве параметров файла разрешено использование только переменных
- 74: Указание внешних файлов разрешено только в заголовках программ (но не и в модулях)
- 75: Заданный параметр не является стандартным файлом
- 76: Переменное поле файла (tag field) должно быть простого типа
- 77: Несоответствие типу переменного поля (tag field)
- 78: Ошибка в списке полей записи
- 79: Базовый тип множества должен быть простым
- 80: Не разрешено опережающее (forward) определение процедур и функций в интерфейсной части модуля
- 82: Не указан файл для связывания с внешней процедурой. Эта ошибка получается, если данная процедура описана как внешняя (external) и не была активизирована директива `{ $L xxx }`
- 83: Внешняя процедура (функция) не обнаружена в указанном директивой `{ $L xxx }` файле
- 84: Метки должны быть в диапазоне [0..9999]
- 85: Попытка применить зарезервированное имя в качестве метки
- 86: Попытка повторного определения метки
- 87: Использование меток не разрешено в интерфейсных частях модулей
- 88,89: Ошибка в указанном директивой `{ $L xxx }` файле. Его невозможно прочитать или его формат не отвечает требованиям. Компилируйте его снова
- 90: Файл, содержащий ассемблерские подпрограммы, не связан. Он использует другие ассемблерские идентификаторы и должен быть связанным с другими ассемблерскими подпрограммами до компиляции основной программы на UniPascal-е
- 91: Внешняя процедура (функция) не обнаружена в указанном для связывания файле
- 92: Внешняя процедура (функция) не является переместваемой (relocatable). Эта ошибка получается, если ассемблирование ассемблерской подпрограммы выполнено директивой ORG или если при связывании ассемблерских подпрограмм не указано явным образом, что нужно получить снова OBJ файл
- 93: Неожиданное появление зарезервированного слова 'begin'. В `{ $M+ }` 'begin' находится только в начале блока
- 94: Метка не описана в текущем блоке. В UniPascal-е не разрешается переход из тела процедуры в другую
- 95: Неопределенная метка
- 96: Недекларированная метка
- 97: Попытка повторной декларации метки
- 98: Ожидается (но не найдена) метка
- 99: Неправильная метка
- 100: Ошибка в арифметическом или логическом выражении (factor)
- 101: Ошибка при применении переменной
- 102: Выражение должно быть типа Boolean

- 103: Выбирающее выражение CASE оператора должно быть ординального типа
- 104: Дублирование константы в CASE операторе
- 105: Тип константы не совпадает с типом выражения выбора в CASE операторе
- 106: Ожидается выражение дискретного типа
- 107: Элементы множества должны быть совместимыми по типу
- 108: Нижняя граница превышает верхнюю
- 109: Много меток в одном операторе. При необходимости применяйте пустой оператор. Например, вместо неправильного Lab1: Lab2: Lab3: ... используйте правильное Lab1: ; Lab2: ; Lab3: ...
- 110: Ожидается (но не найдено) выражение типа Boolean
- 111: Переменная управления циклом FOR должна быть локальной для текущей процедуры
- 112: Неправильный тип переменной управления FOR-циклом
- 113: Неправильный тип выражения
- 114: Несовпадение типов (тип выражение не совпадает с типом переменной управления циклом)
- 115: Структура не является пакетированной на границе байта
- 116: Несовместимость выражения по присваиванию с переменной левой части оператора присваивания
- 117: Переход через EXIT (Попытка при помощи GOTO передать управление из завершающей части процедуры, функции, программы или модуля в тело)
- 118: Неправильный тип переменной
- 119: Управляющая переменная FOR цикла не может иметь тип LongInt
- 120: Подпрограммы ассемблера (MC6800 native code) должны начинаться с \$0000.
- 121: Нет такого варианта в этой записи
- 122: Длина неизвестна компилятору
- 123: Несовместимость с переменным полем записи
- 124: Параметры типа процедуры (функции) не поддерживаются этой версией UniPascal-я
- 125: Упакованные переменные не могут выступать в роли фактических параметров-переменных
- 139: Переменная не является указателем
- 140: Переменная не является массивом
- 141: Тип индекса должен быть дискретным типом
- 142: Тип индекса не совпадает с декларацией
- 143: Тип переменной не является записной
- 144: Нет такого поля в этой записи
- 145: Выражение не обладает множественным типом
- 146: Константа типа строки имеет неправильную длину
- 147: Нельзя использовать эту процедуру как фактический параметр процедуры EXIT.
- 149: Не разрешено применение текстового файла (в качестве параметра)
- 150: Не разрешено в этом месте присваивать значение этой функции (Вероятнее всего нарушение правил вложенности)
- 151: Процедуры не имеют имплицитного результата
- 152: Возвращаемое функцией значение должно быть определено (хотя бы так - return())

- 153: Файл должен быть нетипизированным
- 154: Ожидается (но не найдено) имя файла (переменная типа STRING).
- 155: Ожидается (но не найдена) переменная типа файла
- 156: Неправильный тип выражения
- 157: Неправильный тип параметра стандартной процедуры
- 158: Неправильный тип операндов
- 159: Несовместимые по типу операнды
- 160: Разрешена только проверка на равенство
- 161: Число параметров не совпадает с заданным в декларации
- 162: Фактический параметр должен быть переменной
- 163: Несоответствие типу формального параметра
- 164: Необходим тип scalar или subrange
- 165: Несовместимость с размером (формального) параметра
- 166: Проверка на строгое включение не разрешена
- 167: Недопустимое использование присваивания
- 168: Недопустимое использование константы
- 169: Нигде в теле функции не указан ее результат
- 170: Ожидается переменная типа типизированного файла
- 171: Ожидается целое выражение
- 172: Невозможно читать или писать переменные этого типа. Вероятнее всего попытка применения массива или записи в стандартных процедурах READ, READLN, WRITE, WRITELN. Каждый элемент должен быть обработан в отдельности.
- 173: Ожидается переменная типа текстового файла (для операторов READLN, WRITELN)
- 183: Ошибка в факторе (неправильное выражение)
- 184: Несовместимый или неправильный тип операндов (бинарной операции)
- 185: Нулевое значение знаменателя
- 186: Неправильный тип константы
- 187: Недопустимое значение константы
- 190: Настоящая версия не позволяет применение диапазона типа LongInt
- 191: Настоящая версия не позволяет сравнение констант (в разделе декларации)
- 192: Настоящая версия не позволяет применение этой стандартной функции с константным параметром (в разделе декларации)
- 193: Настоящая версия не позволяет применение выражений с участием вещественных констант (в разделе декларации)
- 194: Размер структуры больше допустимого (64 килобайта)
- 195: Базовый тип множества вне интервала [0..255]
- 196: Очень много и/или очень большими размерами переменные. Теоретически размер локальных переменных не должен превышать 32767 слов (64 килобайта).
- 211: Идентификатор неправильного класса
- 212: Недекларированный идентификатор
- 213: Неразличимый идентификатор. Идентификатор определен в не менее двух модулях - необходимо указать точно в каком из модулей будет использован
- 214: Неправильный символ (вероятно в предыдущей строке пропущен символ ";")

- 215: Здесь не разрешено использование константы
- 216: Тип переменной или выражения несовместим с ожидаемым типом
- 217: Неправильное использование возможности для изменения типа (Type Cast). Обычно ошибка вызывается попыткой изменения типа переменной, длина которой отличается от длины типа.
- 218: Компилятор ожидает переменной
- 219: Указанный идентификатор не является идентификатором процедуры
- 220: Указанный идентификатор не является идентификатором функции
- 227: Недопустимое значение константы
- 228: Константа типа строки длиннее исходной строки
- 229: Константа типа строки длиннее 255 символов
- 230: Недопустимо длинная процедура. Разделите ее на две части.
- 231: Общая длина "длинных" констант больше 32 килобайта. Совет - организуйте обработку одной из них в качестве файла
- 232: Число процедур или функций больше 255. Совет - объявить часть процедур с вложенными процедурами как сегментные или разделить программу на модули
- 233: Число использованных модулей превышает допустимое. Совет - объединить некоторые из них
- 234: Сегмент длиннее допустимого. Совет - разделить его на два сегмента или модуля
- 235: Число сегментов больше 255. Совет - объединить некоторые из них. Если и это не помогает, значит общее число процедур и функций больше 50000. Следует обдумать заново всю программу. В этом виде ее нельзя выполнить на компьютере Пылдин.
- 237: Ожидается (но не найдена) директива `{$endif}`, закрывающая тело директивы типа `{$IFxxx}`
- 238: Общая длина дефинированных символов условной компиляции превышает допустимую
- 239: Неправильное применение директивы условной компиляции

Е.1.3. Фатальные ошибки

При обнаружении ошибки этого класса, компилятор выдает сообщение и прекращает компиляцию.

Часть описываемых здесь ошибок возникают при попытке открыть файл. Одной из возможных причин является то, что открыто уже максимальное число (N) одновременно открытых файлов. Допустимое число одновременно открытых файлов устанавливается командной линией системы UniDOS при помощи команды HANDLES. По умолчанию это число принимается равным 10. Число N распределяется следующим образом. Надо резервировать 5 для системных файлов INPUT, OUTPUT, MESSAGE, AUXILIARY и PRINTER (в операционной системе их имена следующие StdIn, StdOut, StdErr, StdAux и StdPrn). Компилятору необходимо открыть одновременно хотя бы 4 файла: файл, содержащий оверлейный код компилятора; выходной текст программы, подлежащей компиляции; файл, в котором будет записываться генерированный код программы; рабочий файл. Если есть включение файла (директива include), это число увеличивается числом вложенных включений (например, если в тексте программы имеются две директивы для включения, необходимо добавить еще один файл, но если вторая директива является вложенной по отношению к первой, надо открыть еще два

файла). Лучше всего поставьте команду HANDLES 16 в файл AUTOEXEC.JOB. Этим устанавливается число 16 в качестве допустимого числа открытых файлов (для пользователя остаются 11). Это число в большинстве случаев достаточно.

Другой возможной причиной для того, чтобы компилятор не смог создать и/или открыть для чтения файл, является неправильное определение пути в конфигурационном файле. Например:

```
/T d:\temp
/S d:\sym
/B d:\bdy
/P d:\pgm
/U d:\sym
```

а таких поддиректорий и/или дисковых устройств нет.

- 240: Неправильная директива включения внешнего файла (include) или связывания внешнего файла (link)
- 241: Использование директивы включения файла здесь не разрешено (перед "program")
- 242: Попытка осуществить больше 4 включения файлов
- 243: Неоконченный комментарий
- 244: Ошибка в таблице символов. Совет - перекомпилировать соответствующий модуль
- 245: Конфликтная ситуация при использовании модулей. Вероятнее всего описательная часть некоторого модуля была перекомпилирована, а использующие ее модули - нет
- 246: Ошибка при вводе символьной таблицы из .SYM файла. Вероятнее всего ошибка технического устройства - перекомпилируйте модуль
- 247: Невозможно найти таблицу символов (.SYM файл). Совет - перекомпилируйте модуль
- 248: Невозможно записать выходной (.PGM или .BDY или .SYM) файл. Совет - проверьте есть ли на диске свободное место
- 249: Невозможно открыть выходной (.pgm или .bdy или .sym) файл. Совет - проверьте есть ли на диске свободное место
- 250: Не найден файл для включения в исходный файл или файл для связывания. Проверьте указано ли ПОЛНОЕ имя файла (с расширением и, если необходимо, с логическим устройством, директории и т.д.)
- 251: Попытка компиляции реализационной (implementation) части модуля, который по дефиницию не может иметь такой части (interface only unit)
- 252: Физический конец процедуры/функции найден перед логическим окончанием. Процедура или функция должна быть записана ПОЛНОСТЬЮ в одном исходном/включаемом файле
- 253: Невозможно прочитать исходный файл (ошибка ввода/вывода). Вероятнее всего - технический сбой.
- 254: Длина строки исходного файла больше 128 символов
- 255: Физический конец программы/модуля найден перед логическим окончанием.

Примечание: Ошибка 243 (неоконченный комментарий) появляется, только если файл заканчивается незакрытым комментарием. Например:

```

...
    queue:= queue^.next;           (* обработка следующего элемента
end { while };                    |
end;                               |_____ незакрытый комментарий.
=== Bottom of File ===

```

Если после незакрытого комментария встретится закрывающая комментарная скобка в символьной строке (например, ' {X} ' или в другом закрытом комментарии с комментарными скобками того же типа, эта ошибка не появляется. Например:

```

...
    queue:= queue^.next;           { обработка следующего элемента
end { while };                    |_____ незакрытый комментарий,
end;                               |_____ комментарий закрывается этой комментарной скобкой.
=== Bottom of File ===

```

Компилятор "не знает" где должен быть конец комментария. Вероятно возникнут синтаксические ошибки другого типа, например, 252, 255 (как в вышеуказанном примере) или появятся ошибки, связанные с недеklarированными идентификаторами. Но возможно и отсутствие каких либо ошибок (если окажется, что в роли комментария выступили несколько операторов и этим не нарушается синтаксис и семантика языка). Если ошибка выдана сразу после закрытого комментария и нет никаких оснований для ее появления, проверьте нет ли где-то в направлении к началу программы незакрытый комментарий.

Е.2. Диагностические сообщения интерпретатора

Кроме обнаруженных в процессе компиляции ошибок, программа может содержать ошибки, которые появляются только в процессе выполнения. При выполнении никакой программы не должно появляться сообщение об ошибке с номером, превосходящим 240. Но если произошло такое событие, обратитесь к НИПЛ "Програмно осигуряване". Для повторения ситуации необходимо сохранить текст и компилированный код программы, вызвавшей эту ошибку. Сначала Вам необходимо убедиться, что при повторении ситуации (компиляция, выполнение) получается та же самая ошибка и что Ваша программа не разрушает часть операционной системы или себя, тогда дальнейшее поведение компьютера нельзя предсказать. Ошибки этого типа самые трудные для обнаружения и устранения. Попытайтесь включить проверку границ {\$R+} и/или поставьте тестовой печать в программе.

Следует перечень ошибок, выдаваемых интерпретатором в программе.

- 1: Недостаточная свободная память для локальных переменных или для обслуживания заявки на память, происходящей из процедур NEW /GET-MEMWORDS
- 3: Нулевое значение знаменателя
- 4: Переполнение строки символов
- 5: Номер элемента множества вне интервала [0..255]
- 7: Неправильное индексирование переменной типа строки
- 8: Значение данного выражения меньше допустимой нижней границы. Получается при включенной директиве {\$R+}

- 9: Значение данного выражения больше допустимой верхней границы. Получается при включенной директиве {\$R+}
- 32: Ошибка при использовании стандартной процедуры DISPOSE или FREEMEMWORDS. Попытка освободить память, которая не была резервирована применением NEW или GETMEMWORDS (например, память, принадлежащая операционной системе, или незанятая память)
- 65: Переполнение при вычислениях с плавающей точкой (например, $1.25e20 * 3.47e20$)
- 66: Неправильная операция или операнд с плавающей точкой (например, LN(0))
- 67: Нулевое значение знаменателя при делении с плавающей точкой
- 68: Переполнение при превращении вещественного числа в целое (например, trunc(1.5e30))
- 208: Не хватает памяти, чтобы в ней могли поместиться код программы, используемые ею модули, ее глобальные переменные и глобальные переменные ее модулей.
- 209: Ошибка при вводе исполнимого файла. Вероятнее всего - технический сбой.
- 210: Нет выполнимого кода указанного модуля
- 211: Не найден файл сегмента. Не меняйте дискет во время выполнения.
- 212: Неправильная версия программы/модуля
- 214: Попытка для выхода из или входа в несуществующей процедуре. Вероятно программа разрушает системные области компьютера.
- 216: Рекурсивная инициализация модулей. Например, модуль А использует модуль В, который, со своей стороны, использует модуль А. Этого нельзя делать. Один из них должен быть подчиненным другому.
- 217: Неправильный формат библиотеки (SYSTEM.UPL). Вероятнее всего структура библиотеки разрушена неправомерным записей. Восстановите библиотеку с дистрибутивного дискета.
- 218: Неправильный формат .BDY файла. В силе то, что сказано для ошибки 217. Перекомпилируйте разрушенного модуля.
- 219: При выполнении программы, использующей явным или неявным образом данный модуль, не обнаружена его реализационная (implementation) часть, т.е. .BDY файл. Имя этого модуля выдается в строке, за которой следует строка сообщения. Используйте VLpath, чтобы указать где надо искать реализационных частей модулей или полного имени библиотеки. Если уже Вы сделали это, проверьте записаны ли там необходимые .BDY файлы и/или библиотека.

Е.3. Перечень ошибок ввода/вывода

Ошибки ввода/вывода с кодом, меньшим 128 (\$80) совпадают с теми, выдаваемыми операционной системой UniDOS.

- 0 (\$00): No error
- 1 (\$01): Invalid function number
- 2 (\$02): Invalid drive
- 3 (\$03): Disk write protected
- 4 (\$04): Address failure
- 5 (\$05): Data failure

- 6 (\$06): General failure
- 7 (\$07): Invalid sector
- 8 (\$08): Reserved
- 9 (\$09): Invalid media type
- 10 (\$0a): FAT failure
- 11 (\$0b): Path not found
- 12 (\$0c): Reserved
- 13 (\$0d): Too many open files
- 14 (\$0e): Access denied
- 15 (\$0f): File already open
- 16 (\$10): Invalid file handle
- 17 (\$11): Disk full
- 18 (\$12): File lost in directory
- 19 (\$13): Invalid name
- 20 (\$14): Root directory full
- 21 (\$15): Directory exists
- 22 (\$16): Attempt to remove the current directory
- 23 (\$17): Directory not empty
- 24 (\$18): Bad LSEEK position
- 25 (\$19): Reserved
- 26 (\$1a): Not disk file
- 27 (\$1b): Too many drivers installed
- 28 (\$1c): Not same device
- 29 (\$1d): File exists
- 30 (\$1e): Printer timeout
- 31 (\$1f): Printer general failure
- 32 (\$20): Printer out of paper
- 33 (\$21): AUX: timeout
- 34 (\$22): AUX: parity
- 35 (\$23): AUX: overrun
- 36 (\$24): AUX: framing
- 64 (\$40): Invalid number of parameters
- 65 (\$41): Bad command or file name
- 66 (\$42): Insufficient memory
- 67 (\$43): Failure in .PGM file
- 68 (\$44): Language not available
- 69 (\$45): Cannot execute .CMD file
- 70 (\$46): File cannot be copied onto itself
- 71 (\$47): File opening failure
- 72 (\$48): File creation failure
- 73 (\$49): File not found
- 74 (\$4a): Invalid parameter

Ошибки ввода/вывода с кодом от 128 (\$80) до 255 (\$FF) выдаются интерпретатором Y кода. Они относятся к семантикой языка UniPascal и, обычно, не зависят от операционной системы.

- 128 (\$80): Попытка использования неоткрытого файла. Например, попытка выполнить SEEK, READ или WRITE над описанным, но неоткрытым файлом.
- 129 (\$81): Попытка открытия (OPEN, RESET, REWRITE или APPEND) файла, который уже открыт.
- 130 (\$82): Попытка записи в файл, который открыт только для чтения.
- 131 (\$83): Попытка чтения из файла, который открыт только для записи.
- 132 (\$84): Попытка для чтения или записи из /в файл, котой не открытый для соответствующей операции (BLOCKREAD / BLOCKWRITE).
- 133 (\$85): Попытка выполнения операции для нетипизированного файла над типизированным.
- 134 (\$86): Попытка выполнения операции, несоответствующей виде файла (типизированный/нетипизированный).
- 135 (\$87): Попытка выполнения операции для текстового файла (READLN/WRITELN) над нетекстовым.
- 136 (\$88): Стандартная процедура BLOCKREAD/BLOCKWRITE обработала меньше указанных параметром байтов.
- 137 (\$89): Попытка чтения после конца файла.
- 143 (\$8F): Попытка повторного открытия файла, но режим не соответствует режиму при первом открытии.
- 144 (\$90): Ошибка в формате целого или вещественного числа.