

1. Список литературы

1. Ван Кеннейт М.. Справочное руководство для FPC. 2005.
2. Van Canneyt M. Free Pascal: Reference Guide. 2007.
3. Van Canneyt M. Free Pascal: Programmer's manual. 2007.
4. Van Canneyt M., Klampfl F. Free Pascal: Users' manual. 2007.
5. Van Canneyt M. Free Pascal code documenter: Reference manual. 2007.
6. Van Canneyt M. Free Component Library (FCL): Reference guide. 2007.
7. Van Canneyt M. Run-Time Library (RTL). Reference guide. 2007.
8. Van Canneyt M. Free Component Library (FCL): Reference guide. 2007.
9. <http://www.FPC.org/download/i386/win32-ftp.FPC.org.html>
10. <http://www.FPC.org/download.html>
11. Рудюк С.А., Lazarus - кросс-платформенный Delphi. Программирование для свободных людей. Компания НеРусСофт
12. Акчурин Э.А. Программирование на языке высокого уровня. Учебное пособие. Самара, ПГАТИ, 2003.
13. Акчурин Э.А., Стефанова И.А. Программирование в среде Delphi. Методические указания к лабораторным работам. Самара, ПГАТИ, 2003.
14. Акчурин Э.А. Задания и методические указания к курсовой работе. ПГАТИ, 2001.

2. Введение

2.1. История FPC

В 1970 году Никлаусом Виртом был создан язык программирования Pascal, как язык для обучения процедурному программированию. Особенности языка являются строгая типизация, средства структурного (процедурного) программирования, сведены к минимуму возможные синтаксические неоднозначности, а сам синтаксис интуитивно понятен даже при первом знакомстве с языком.

Достоинства языка заставляли многие коммерческие и некоммерческие организации разрабатывать системы программирования на основе языка Паскаль. Из них выделяется разработка от Borland - язык Turbo Pascal, в котором устранены многие недостатки языка, добавлены новые возможности. Язык стал богаче, но одновременно, потерял переносимость и общность. Развитием языка Borland Pascal является Object Pascal — версия языка Паскаль, расширенная средствами объектно-ориентированного программирования. Последние версии Borland Pascal лежат в основе среды программирования Delphi.

Важным шагом в развитии языка, является появление свободных версий языка Паскаль. В настоящий момент пользуются популярностью такие версии, как TMT Pascal, FPC (Free Pascal Compiler) и GNU Pascal. Продолжает использоваться и Borland Pascal.

Реализация языка Delphi проектом FPC позволяет использовать его для создания приложений не только в Windows, но и для платформ Mac OS X, Windows CE и Linux. Важной особенностью компилятора FPC, в отличие, например, от GNU Pascal, является ориентация на распространённые коммерческие диалекты языка: Object Pascal и Delphi. Компилятор распространяется на

Оглавление

1. Список литературы	8
2. Введение	8
2.1. История FPC	8
2.2. Основы FPC	9
2.3. Режимы FPC	9
2.4. Интегрированные среды разработки	9
3. Программа на FPC	10
4. Модули	10
4.1. Модуль System	12
4.1.1. Разные средства	12
4.1.2. Средства работы с операционной системой	12
4.1.3. Математические функции	12
4.1.4. Средства работы с динамическими переменными	13
4.2. Модуль SysUtils	13
4.3. Модуль ObjPas	13
4.4. Модуль Math	13
4.4.1. Геометрические функции	14
4.4.2. Определение максимума/минимума	14
4.4.3. Экспоненциальные и логарифмические функции	14
4.4.4. Функции преобразования чисел	14
4.4.5. Функции преобразования единиц	14
4.4.6. Тригонометрические функции	15
4.4.7. Гиперболические функции	15
4.4.8. Статистические функции	15
5. Директивы компилятора	16
5.1. Глобальные директивы	16
5.2. Локальные директивы	17
6. Язык и средства FPC	19
6.1. Стандартные литералы	19
6.2. Специальные символы	19
6.3. Лексемы языка	20
6.4. Комментарии	21
6.5. Зарезервированные слова	21
6.5.1. В режиме Turbo Pascal	21
6.5.2. В режиме Delphi	23
6.5.3. В режиме FPC	23
6.5.4. Модификаторы	23
6.6. Разделители	24
6.7. Идентификаторы	24
6.8. Числа	25
6.9. Метки	26
6.10. Символьные строки	26

6.11. Константы	26
6.11.1. Истинная константа	26
6.11.2. Типизированные константы	27
6.11.3. Ресурсные строки	28
7. Типы	28
7.1. Базовые (простые) типы	29
7.1.1. Порядковые типы	29
7.1.2. Вещественные (действительные) типы	30
7.2. Символьные типы	31
7.2.1. Символы	31
7.2.2. Строки	33
7.3. Структурированные типы	35
7.3.1. Массивы	35
7.3.2. Записи (комбинированные типы)	36
7.3.3. Множества	37
7.3.4. Файлы	38
7.4. Указатели	38
7.5. Типы пользователя	38
7.5.1. Перечислимый тип	39
7.5.2. Интервальный	39
7.6. Процедурный тип	39
7.7. Тип Variant	40
7.8. Тип дата-время	41
7.9. Совместимость типов	41
7.9.1. Идентичность	41
7.9.2. Совместимость типов для сравнений	41
7.9.3. Совместимость типов для присваивания	42
8. Переменные	42
9. Классы и объекты	42
9.1. Объекты	42
9.2. Классы	43
9.2.1. Инкапсуляция	44
9.2.2. Наследование	44
9.2.3. Полиморфизм	44
9.2.4. Создание класса	45
9.2.5. Поля класса	45
9.2.6. Свойства класса	45
9.2.7. Методы класса/объекта	46
9.3. Создание и удаление объекта	47
9.4. Работа с объектом	48
9.5. Обработка событий	49
9.6. Совместимость объектов в операторах присвоения	49
10. Интерфейсы	49
10.1. Интерфейсы и технология COM	50
10.2. CORBA и другие интерфейсы	50

11. Шаблоны (дженерики)	50
12. Выражения	50
12.1. Арифметические операции	51
12.2. Строковые операции	51
12.3. Операции сравнения	51
12.4. Логические операции	52
12.4.1. Операция not - отрицание ("НЕ")	52
12.4.2. Операция or - логическое сложение ("ИЛИ")	52
12.4.3. Операция and - логическое умножение ("И")	53
12.4.4. Операция xor - исключающее "ИЛИ"	53
12.4.5. Операция shl - логический сдвиг влево	53
12.4.6. Операция shr - логический сдвиг вправо	54
12.5. Операции над указателями	54
12.6. Операции над множествами	55
12.6.1. Основные операции	55
12.6.2. Процедуры для множеств	55
12.7. Операции над классами	55
12.8. Приоритеты операций	56
13. Инструкции	56
13.1. Простые инструкции	56
13.1.1. Инструкции присваивания	56
13.1.2. Инструкция безусловного перехода	57
13.1.3. Пустая инструкция	58
13.1.4. Инструкция вызова процедуры	58
13.2. Структурированные инструкции	58
13.2.1. Блочная инструкция	58
13.2.2. Условные инструкции	58
13.2.3. Инструкции циклов	63
13.2.4. Инструкция доступа with	69
14. Обработка исключений	70
14.1. Инструкция Raise	70
14.2. Инструкция try...except	71
14.3. Инструкция try...finally	71
15. Использование ассемблера	72
16. Процедуры и функции	72
16.1. Формальные и фактические параметры	72
16.2. Процедура	73
16.3. Функция	74
16.4. Области действия имен	75
16.5. Нетипизированные параметры	75
16.6. Директивы подпрограмм	75
16.6.1. Директива External	76
16.6.2. Директива Overload. Перегрузка функций	76
16.6.3. Процедура Exit	77
17. Средства работы с файлами в FPC	78

17.1. Файловые типы	78
17.1.1. Типизированные файлы	79
17.1.2. Текстовые файлы	79
17.1.3. Нетипизированные файлы	79
17.2. Операции над файлами модуля System	80
17.2.1. Установочные и завершающие операции	80
17.2.2. Операции перемещения по файлу	80
17.2.3. Операции ввода/вывода	80
17.2.4. Специальные операции	82
17.2.5. Текстовые файлы в модуле System	82
17.2.6. Типизированные файлы в модуле System	84
17.2.7. Нетипизированные файлы в модуле System	85
17.3. Средства работы с файлами модуля SysUtils	86
17.3.1. Атрибуты файлов	86
17.3.2. Режимы файлов	86
17.3.3. Установочные и завершающие операции	86
17.3.4. Операции перемещения по файлу	87
17.3.5. Операции ввода/вывода	87
17.3.6. Специальные операции	88
17.3.7. Работа с файлами средствами SysUtils	88
18. Динамическая память	89
18.1. Свойства указателей	89
18.2. Средства работы с динамической памятью	90
18.3. Работа с динамическими структурами	90
19. Модули	92
19.1. Модуль общего назначения	92
19.1.1. Структура модуля	92
19.1.2. Модуль комплексной арифметики	93
19.2. Динамически связываемая библиотека DLL	94
19.2.1. Введение	94
19.2.2. Структура DLL	95
19.2.3. DLL комплексной арифметики	96
20. Работа со строками	97
20.1. Структура строк	97
20.1.1. Строка ShortString	97
20.1.2. Строка AnsiString = LongString	97
20.1.3. Строка WideString	97
20.1.4. Строка PChar	97
20.1.5. Строка PWideChar	98
20.2. Методы обработки строк	98
20.3. Строки String	98
20.3.1. Средства модуля System	98
20.3.2. Средства модуля SysUtils	98
20.3.3. Примеры	99
20.4. Строки PChar	101

21. Отладка программ	105
21.1. Введение	105
21.2. Классификация ошибок	105
21.2.1. Введение	105
21.2.2. Синтаксические ошибки	105
21.2.3. Ошибки времени выполнения	105
21.2.4. Алгоритмические ошибки	106
21.2.5. Предотвращение и обработка ошибок	106

ЭБС ПШУТИИ