

Программа курса «Основы программирования»
мехмат, 1 курс, специальность «Информационные технологии»
1 семестр 2012–2013 уч. г.

1. Понятие алгоритма. Свойства алгоритма. Эквивалентность алгоритмов. Исполнитель алгоритма. Показатели эффективности алгоритма.
2. Способы описания алгоритмов: словесный, псевдокод, блок-схемы, диаграммы деятельности, языки программирования.
3. Язык программирования (ЯП). Синтаксис и семантика ЯП. Способы задания синтаксиса ЯП: формы Бэкуса-Наура, расширенные формы Бэкуса-Наура, синтаксические диаграммы.
4. Компиляторы и интерпретаторы. Виртуальные машины, JIT-компиляция, промежуточный код. ~~Преимущества и недостатки виртуальных машин.~~
5. Общая структура программы на языке Паскаль. Лексемы языка Паскаль.
6. Понятие типа данных. Основные типы языка Паскаль. Разделы описания переменных и констант. Внутриблочные описания.
7. Оператор присваивания. Примеры использования. Совместимость типов по присваиванию. Расширенные операторы присваивания: $+=$ $-=$ $*=$ $/=$. Вычисление степеней за наименьшее число умножений.
8. Оператор вывода. Форматы вывода : и ::. Процедура `writelnFormat`.
9. Оператор ввода. Обработка ошибок ввода. Оператор `try ... except`.
10. Арифметические выражения. Операции в арифметических выражениях. Тип арифметического выражения. Операции `div` и `mod`, `shl` и `shr`. Стандартные арифметические функции и процедуры.
11. Логические выражения и операции. Таблицы истинности. Таблица приоритетов операций языка Паскаль.
12. Условный оператор. Вложенные условные операторы. Составной оператор.
13. Оператор выбора варианта. Тип выражения-переключателя в операторе выбора варианта.
14. Цикл с предусловием. Цикл с постусловием. Цикл с параметром. Моделирование цикла `repeat` с помощью цикла `while`. За цикливание и бесконечные циклы.
15. Примеры использования циклов: сумма и произведение введенных чисел; факториал; подсчет количества чисел, удовлетворяющих некоторому условию.
16. Примеры использования циклов: табулирование функции. Решение с помощью цикла `for` и
17. Погрешность округления и вычислительная погрешность. Хранение вещественного значения в памяти. Мантисса и порядок.
18. Рекуррентные последовательности. Арифметическая и геометрическая прогрессии как частные случаи рекуррентных последовательностей. Числа Фибоначчи.
19. Примеры использования циклов: алгоритм Евклида нахождения НОД.
20. Предусловия и постусловия. Процедура `assert`.
21. Примеры использования циклов: сумма и произведение цифр целого числа.
22. Примеры использования циклов: поиск максимума (минимума), поиск условного максимума (минимума). Понятие инварианта цикла. Доказательство корректности алгоритма поиска максимума с помощью инвариантов.
23. Примеры использования циклов: суммирование конечных и бесконечных рядов.
24. Операторы `break` и `continue`. Использование флагов при решении задач. Эквивалентные циклы без `break` и `continue`. Примеры использования циклов: поиск заданного значения среди введенных.
25. Примеры использования циклов: определение простоты числа.
26. Примеры использования циклов: разложение целого числа на простые сомножители. Доказательство завершимости алгоритма.
27. Понятие асимптотической сложности алгоритма. Асимптотически эквивалентные последовательности ($a_n \sim b_n, n \rightarrow \infty$). Нотация Θ .
28. Примеры использования циклов: схема Горнера. Асимптотическая сложность «наивного» алгоритма вычисления значения многочлена. Асимптотическая сложность алгоритма схемы Горнера.
29. Примеры использования циклов: вычисление корня функции на отрезке. Метод половинного деления.
30. Вложенные циклы, примеры их использования. Метод окаймления и метод последовательной детализации.
31. Переборные задачи и оптимизация самого внутреннего цикла. Преждевременная оптимизация.
32. Оператор `goto`. Семантические ограничения для оператора `goto`. Примеры использования.

33. Подпрограммы. Цели введения подпрограмм в ЯП.
34. Процедуры. Описание и вызов процедуры. Формальные и фактические параметры. Семантические правила при вызове. Оператор exit. Примеры использования процедур.
35. Функции. Описание и вызов функции. Переменная Result. Примеры использования функций: сумма квадратов чисел от a до b, поиск заданного числа среди введенных. Сравнение функций и процедур.
36. Способы передачи параметров: по значению и по ссылке. Понятие ссылки. Внутренний механизм передачи параметра по ссылке. Входно-выходные параметры.
37. Локальные и глобальные переменные. Инициализация локальной и глобальной переменной. Обращение к нелокальной переменной на чтение и запись и побочный эффект.
38. Область видимости и время жизни объекта. Пространство имен.
39. Принцип локальности при описании переменных.
40. Перегрузка имен подпрограмм. Разрешение перегрузки. Неоднозначности при разрешении перегрузки.
41. Параметры по умолчанию.
42. Предварительное объявление подпрограмм.
43. Шаблоны подпрограмм. Вывод типа шаблона при вызове подпрограммы.
44. Процедурный тип и процедурные переменные. Процедурные переменные как параметры подпрограмм. Подпрограммы обратного вызова (callback). Пример использования процедурных переменных – вычисление определенного интеграла.
45. Методы разработки программ сверху вниз и снизу вверх. Преимущества и недостатки каждого метода.
46. Алгоритм вызова подпрограммы на этапе выполнения. Программный стек и запись активации подпрограммы. Замечания: накладные расходы на вызов, инициализация локальных переменных, переполнение программного стека, ~~скорость выделения памяти под запись активации, доступ к глобальным переменным.~~
47. Модули. Цели введения модулей. Структура модуля. Разделы интерфейса и реализации. Необходимость разделения на интерфейс и реализацию. Принцип сокрытия информации. Упрощенная структура модуля.
48. Разделы инициализации и финализации. Схема компиляции программы с модулями.
49. Алгоритм поиска имен в модулях. Модуль как пространство имен. Системный модуль PABCSysSystem.
50. Библиотеки. Отличие библиотек от модулей. Подключение библиотек в PascalABC.NET. Представление о директивах компилятора.
51. Перечислимый и диапазонный типы. Стандартные подпрограммы для работы с перечислимым типом. Использование перечислимого типа в for и case.
52. Описание пользовательского типа. Структурная и именная эквивалентность типов.
53. Структурированные типы данных. Статические массивы, описание, инициализация. Тип индексов массива. Обращение к элементу по индексу. Выход за границы диапазона. Хранение статических массивов в памяти. Присваивание статических массивов.
54. Динамические массивы. Хранение динамических массивов в памяти. Присваивание динамических массивов. Сравнение динамических массивов. Изменение длины динамического массива по ходу работы программы. Цикл foreach.
55. Передача статических и динамических массивов в качестве параметров: особенности использования модификаторов var и const. Именная и структурная эквивалентность для типов статических и динамических массивов. Динамические массивы как возвращаемые значения функций.
56. Переменное число параметров подпрограмм.
57. Задачи на одномерные массивы: сумма и произведение элементов; инвертирование массива; поиск (реализации с for и while); поиск с барьером; минимальный/максимальный элемент и их индексы; сдвиг влево/вправо; циклический сдвиг; вставка элемента по заданному индексу, удаление элемента по заданному индексу; слияние двух упорядоченных в один упорядоченный.
58. Задачи на одномерные массивы с процедурными переменными в качестве параметров: поиск элемента по заданному условию; количество по условию; условный минимум/максимум; удаление всех по условию.
59. Сортировки массивов. Сортировка выбором. Пузырьковая сортировка (обычная и с флагом). Сортировка вставками. Асимптотическая сложность алгоритмов сортировки.
60. Бинарный поиск в упорядоченном массиве. Его асимптотическая сложность.
61. Двумерные массивы (матрицы). Двумерные динамические массивы. Элементы строки, столбца. Создание, заполнение, вывод.

62. Задачи на двумерные массивы: вывод; сумма элементов в j-том столбце; сумма элементов в каждом столбце; минимальный элемент в каждой строке (~~2 варианта: с функцией и без~~); поиск элемента; обнуление элементов выше/ниже диагонали в квадратной матрице.
63. Записи. Поля записей. Инициализация записей. ~~Оператор with~~. Хранение записей в памяти. Передача записей как параметров подпрограмм.
64. Методы в записях. Объектно-ориентированный и процедурно-ориентированный подходы, их сравнение. ~~Запись как модуль и как пространство имен~~. Виды методов. Обращение к полям записей из методов.
65. Записи как средство упаковки параметров (~~на примере модуля графических объектов~~).
66. Индексная сортировка массивов записей. Сортировка массива записей по критерию сравнения, передаваемому в качестве параметра.
67. Статические методы в записях.
68. Множества. Базовый тип множеств. Инициализация множеств. Операции с множествами. Вывод множеств. Цикл по множеству.
69. Решето Эратосфена.
70. Символы. Коды символов. Однобайтовые кодировки. Двухбайтовые кодировки. Стандартные подпрограммы работы с символами. Статические методы типа char.
71. Строки. Операции со строками. Основные подпрограммы работы со строками. Основные методы типа string (экземплярные и статические).
- ~~72. Некоторые задачи на строки: формирование строки 'ABCDEF...Z', сумма всех цифр строки; удаление всех вхождений подстроки в строку; сортировка строки слов, разделенных пробелами.~~
73. Преобразование строка <-> число.
74. Некоторые задачи на строки: выделение из строки целых и вещественных чисел.