

## Задания к практическим занятиям по информатике.

Составители: Бахта Наталья Сергеевна

Тыркова Надежда Петровна

### Задания к практическим занятиям по информатике.

Настоящее электронное издание предназначено для студентов первого курса (1 семестр) ИМИТ и соответствует реализуемой в ОмГУ образовательной программе дисциплины информатика по следующим направлениям подготовки: Прикладная математика и информатика, код направления 010400.62 (бакалавры); Фундаментальная математика и Механика, код направления 010701.65 (специалисты).

В пособии по информатике приведены задания к практическим занятиям и домашние задания.

## Практическое занятие 1.

*Системы счисления. Перевод целых и дробных чисел в двоичную систему.*

*Данные с фиксированной точкой.*

*Представление чисел с плавающей точкой в разных форматах.*

### 1. Запись чисел в разных системах счисления.

- 1) Записать в двоичной системе счисления следующие десятичные числа:
  - a) 17;
  - b) 396;
  - c) 65535.
- 2) Записать в десятичной системе счисления следующие двоичные числа:
  - a) 10010110;
  - b) 11100011.
- 3) Записать в шестнадцатеричной системе счисления следующие десятичные числа:
  - a) 17;
  - b) 65535.
- 4) Записать в десятичной системе счисления следующие шестнадцатеричные числа:
  - a) 96;
  - b) E3.
- 5) Записать в двоичной системе счисления следующие шестнадцатеричные числа:
  - a) 15AF3B;
  - b) CABF11.
- 6) Записать в шестнадцатеричной системе счисления следующие двоичные числа:
  - a) 1001101011110001;
  - b) 11100101000111.
- 7) Записать в двоичной системе счисления следующие дробные числа:
  - a) 35,125;
  - b) 21,375.

### 2. Представление целых чисел в знаковых и беззнаковых типах.

1) Выписать битовое представление следующих целых чисел в знаковых целых типах (для каждого числа выбрать минимальный по памяти, но достаточный для его представления тип):

- a) -17;
- b) -396;
- c) -65535.

### 3. Домашнее задание (рассчитано на выполнение после прослушивания лекции №2):

1) Записать в нормализованном виде следующие числа. Какие цифры в их записи являются значащими?

- a) 31,7503(10);
- b) -0,00035(10);
- c) -1101,100011(2);
- d) 0,000101(2).

2) Выписать битовое представление следующих чисел в указанных форматах с плавающей точкой:

- a) -1101,100011(2) (одинарная точность);
- b) 0,000101(2) (двойная точность);
- c) 21,375(10) (одинарная точность);
- d)  $-30657/32768$  (расширенная точность);

3) По заданному битовому представлению числа в формате с плавающей точкой получить запись числа в десятичной системе счисления:

a) 11000010 00001100 10000000 00000000;

b) 00111111 11001001 10011001 10011001 10011001 10011001 10011001 10011010

(известно, что это число задается бесконечной периодической двоичной дробью).

## Практическое занятие 2.

*Простейшие алгоритмы с числовыми данными, составление блок-схем.*

**1.** Составить блок-схемы алгоритмов, решающих нижеследующие задачи. Определить все входные и выходные параметры, описать локальные переменные; если удобно выделить подпрограмму, которая может использоваться самостоятельно, то стоит это сделать.

1) Подсчитать количество цифр в позиционной записи целого числа (возможно отрицательного) в системе счисления с заданным основанием. Число задано своим значением.

2) Подсчитать количество нечетных цифр среди первых  $k$  цифр дробной части позиционной записи числа, обратного к заданному значению целому числу  $N$ .

3) Вычислить количество целых чисел, удовлетворяющих неравенству  $x^2 + px + q \leq 0$ .

4) Вычислить количество точек с целыми координатами внутри круга  $(x-a)^2 + (y-b)^2 \leq r^2$ .

**2.** Домашнее задание:

1) Нарисовать блок-схемы для задач, которые не были до конца разобраны на занятии.

2) Написать соответствующие всем четырем блок-схемам подпрограммы на Паскале.

### Практическое занятие 3.

*Подпрограммы обработки прямоугольных таблиц, заданных в двумерных статических массивах.*

**1. Составить подпрограммы на Паскале, описав необходимые типы данных.**

1) Дана прямоугольная таблица вещественных чисел. Среди сумм элементов строк найти значение максимальной.

2) Дана прямоугольная таблица вещественных чисел. Среди тех строк, которые содержат нули, найти номер первой строки, в которой сумма модулей отрицательных элементов максимальна.

**2. Домашнее задание:**

1) Составить подпрограмму, решающую поставленную задачу: дана прямоугольная таблица вещественных чисел. Элемент  $a[i, j]$  назовем особым, если он равен сумме остальных элементов  $i$ -й строки. Найти количество особых элементов матрицы.

2) Написать программу, которая реализует простейшее меню, обеспечивающее создание таблицы, заполнение ее с клавиатуры или случайным образом, вызов любой из 3-х подпрограмм обработки таблицы, просмотр элементов таблицы и завершение работы. (Простейшее меню выглядит так: на экране написаны несколько действий, которые можно выполнить, а внизу подсказка с просьбой ввести номер действия, которой необходимо выполнить. Например, экран может иметь такой вид:

- 3) Определить размеры таблицы
- 4) Заполнить таблицу с клавиатуры
- 5) Заполнить таблицу случайными числами
- 6) Показать элементы таблицы
- 7) Вызвать первую подпрограмму
- 8) Вызвать вторую подпрограмму
- 9) Вызвать третью подпрограмму
- 10) Закончить работу

Введите номер действия:

Если действие, номер которого ввел пользователь нельзя выполнить (например, показать элементы таблицы, если она еще не заполнена и т.д.), должно появляться соответствующее сообщение. После того, как действие выполнено (или выведено сообщение о его невозможности), меню появляется снова и т.д. пока не выберут завершение работы.

## Практическое занятие 4.

### *Подпрограммы обработки прямоугольных таблиц, упакованных в одномерные динамические массивы.*

Упаковка в одномерный массив делается двумя способами: 1) сначала в массив записываются все элементы первой строки, затем все элементы второй строки и т.д. – *упаковка таблицы по строкам*; 2) сначала в массив записываются все элементы первого столбца, затем все элементы второго столбца и т.д. – *упаковка таблицы по столбцам*. Для того, чтобы получить элемент таблицы, находящийся в строке с номером  $i$  и столбце с номером  $j$ , нужно вычислить его номер в одномерном массиве и обратиться к нему.

Например таблица:

	1	2	3
1	1	2	5
2	-3	7	4
3	8	-4	-2

после упаковки в одномерный массив по строкам будет выглядеть так:

1	2	3	4	5	6	7	8	9
1	2	5	-3	7	4	8	-4	-2

Элемент таблицы с индексами (2, 3) в одномерном массиве имеет номер 6.

Если упаковать таблицу по столбцам, то получится так:

1	2	3	4	5	6	7	8	9
1	-3	8	2	7	-4	5	4	-2

Элемент таблицы с индексами (2, 3) в одномерном массиве имеет номер 8.

**1.** Составить подпрограммы на Паскале, описав необходимую структуру данных и написав подробный комментарий о параметрах подпрограммы и ее действии:

1) Для заданной прямоугольной таблицы вещественных чисел (упакованной в одномерный массив по строкам) подсчитать количество отсортированных по неубыванию строк.

2) Дана прямоугольная таблица вещественных чисел (упакованная в одномерный массив по строкам). Определить, чего в таблице больше – столбцов, содержащих отрицательные числа, или строк, имеющих положительные суммы.

**2.** Домашнее задание:

1) В таблице вещественных чисел (упакованной в одномерный массив по столбцам) найти максимальный среди элементов, встречающихся более одного раза в каждой строке.

2) Написать тестирующую программу, реализующую меню (см. домашнее задание практического занятия 3), добавить возможность чтения таблицы из типизированного файла, в котором записаны размеры таблицы, а потом элементы, упакованные по строкам или по столбцам.

## Практическое занятие 5.

### *Рекурсия и итерация.*

**1.** Составить подпрограммы на Паскале, описав необходимую структуру данных и написав подробный комментарий о параметрах подпрограммы и ее действии:

1) В динамическом массиве вещественных чисел определить индекс заданного числа, либо сообщить, что его нет.

2) В упорядоченном по неубыванию динамическом массиве вещественных чисел определить индекс заданного числа, либо сообщить, что его нет. При этом постараться максимально использовать упорядоченность для уменьшения числа необходимых действий.

**2.** Указать расстановку 8-ми ферзей на шахматной доске ( $8 \times 8$ ) так, чтобы они не били друг друга.

**3.** Домашнее задание:

1) Написать рекурсивный вариант функции поиска элемента в упорядоченном одномерном массиве.

2) Написать тестирующую программу к написанным подпрограммам поиска в массиве (при создании проверять наличие памяти).

3) Переделать подпрограмму расстановки ферзей в итерационную так, чтобы стек рекурсии моделировался с помощью массива.



## Практическое занятие 6.

### *Инварианты.*

#### 1. Составить подпрограмму на Паскале:

Даны два целых числа  $a$  и  $b$ . Найти три целых числа  $k$ ,  $l$  и  $d$  такие, что  $d = (a, b)$  и  $ka + lb = d$ .

#### 2. Домашнее задание:

1) Написать две подпрограммы для вычисления  $C_n^k$  ( $0 \leq k \leq n$ ): первую с использованием рекуррентного соотношения  $C_n^k = C_{n-1}^k + C_{n-1}^{k-1}$  ( $0 < k < n$ ),  $C_n^0 = 1$ ,  $C_n^n = 1$ ; вторую с использованием формулы  $C_n^k = \frac{n!}{k!(n-k)!}$ . Протестировать на больших  $n$  и  $k$ .

## Практическое занятие 7.

### *Метод индуктивной функции.*

**1.** Составить подпрограммы на Паскале (предусмотреть как можно более полную обработку ошибок):

1) методом индуктивной функции вычислить среднее арифметическое элементов последовательности вещественных чисел. Рассмотреть два варианта задания последовательности: в одномерном массиве и в типизированном файле.

2) методом индуктивной функции при известном значении аргумента вычислить значение многочлена, коэффициенты которого записаны в типизированном файле вещественных чисел по убыванию степеней многочлена.

3) методом индуктивной функции при известном значении аргумента вычислить значение первой производной многочлена, коэффициенты которого записаны в типизированном файле вещественных чисел по убыванию степеней многочлена.

**2.** Домашнее задание:

1) написать подпрограмму на Паскале, определяющую размерность пространства, натянутого на последовательность векторов в трехмерном пространстве, заданных координатами в файле вещественных чисел, и тестирующую программу к ней;

2) написать тестирующую программу к задачам 2) и 3) и протестировать их на различных ошибочных и корректных файлах.

## Практическое занятие 8.

### *Метод индуктивной функции*

**1.** Составить подпрограммы на Паскале (предусмотреть как можно более полную обработку ошибок):

1) методом индуктивной функции вычислить значение средней длины символьного представления целого числа со знаком в последовательности символов из типизированного файла;

2) методом индуктивной функции вычислить значение вещественного числа, заданного своим символьным представлением в десятичной системе счисления; последовательность символов хранится в типизированном файле; число задано в «полунормализованном» виде, т.е. экспоненциальная часть ( $E+/-$  порядок) может присутствовать, но мантисса не обязательно лежит в полуинтервале от 1 до 10.

**2.** Написать тестирующую программу для первой подпрограммы, которая позволяет сгенерировать тестовый файл с заданными частотами появления знаков (+ или -), цифр и остальных символов.

**3.** Домашнее задание:

1) дописать все программы и подпрограммы, не законченные на занятии.

## Практическое занятие 9.

### *Реализация класса «матрица» на основе двумерного динамического массива.*

1. Написать модуль, реализующий класс матрица, в части interface которого содержится описание класса и типа-указателя на класс, а в части implementation – реализация всех методов класса. Индексами строк и столбцов матрицы являются элементы любого порядкового типа. Поля класса (private): типизированный указатель на двумерный динамический массив (индексируемый беззнаковыми целыми числами), границы диапазонов индексов строк и столбцов матрицы (low1..high1 и low2..high2). Методы класса (public, если не указано иное):

1) Создать матрицу с диапазонами индексов L1..H1 и L2..H2 (т.е. L1, H1, L2, H2 – входные параметры конструктора). Т.е. выделить память под двумерный динамический массив размерами (H1–L1) × (H2–L2) и заполнить все поля объекта. Если память выделить невозможно, то вызвать процедуру обработки ошибки (уже выделенную память можно не освобождать, т.к. процедура обработки ошибки завершит работу программы и память освободится операционной системой).

2) Скопировать матрицу с готовой (конструктор копирования). Т.е. в качестве входного параметра передается указатель на уже существующий объект класса «матрица» и конструктор создает новый объект как точную копию этого существующего.

3) Уничтожить матрицу (деструктор). Т.е. освободить память от массива (обязательно) и присвоить всем полям нулевые значения (по желанию).

4) Изменить элемент с индексами i1, i2.

5) Вернуть значение элемента с индексами i1, i2. Можно реализовать как функцию возвращающую значение элемента, а можно как процедуру, которой переменная, предназначенная для значения элемента, передается по ссылке.

6) Вернуть нижнюю границу индексов строк. Функция.

7) Вернуть верхнюю границу индексов строк. Функция.

8) Вернуть нижнюю границу индексов столбцов. Функция.

9) Вернуть верхнюю границу индексов столбцов. Функция.

10) Вернуть число строк. Функция.

11) Вернуть число столбцов. Функция.

12) Private-метод. Обработать ошибку (фатально). Входной параметр – номер ошибки. Процедура выводит на экран описание возникшей ошибки и завершает работу всей программы с помощью процедуры Halt(1) (см. литературу по Паскалю). Это и есть фатальная обработка. Т.е. если в процессе работы какого-то метода возникли ошибки, то этот метод вызывает процедуру обработки ошибки с нужным номером. У каждого метода – свой номер ошибки.

### 2. Домашнее задание:

1) Написать подпрограмму, осуществляющую перестановку строк матрицы так, чтобы в заданном столбце сначала шли все отрицательные числа, потом нули, потом все положительные числа, порядок их следования сохранить.

2) Написать программу, создающую матрицу, заполняющую ее, создающую копию матрицы, вызывающую алгоритм перестановки строк для копии и показывающую элементы обеих матриц.

## Содержание

Практическое занятие 1. Системы счисления. Перевод целых и дробных чисел в двоичную систему. Данные с фиксированной точкой. Представление чисел с плавающей точкой в разных форматах.	3
Практическое занятие 2. Простейшие алгоритмы с числовыми данными, составление блок-схем.	5
Практическое занятие 3. Подпрограммы обработки прямоугольных таблиц, заданных в двумерных статических массивах.	6
Практическое занятие 4. Подпрограммы обработки прямоугольных таблиц, упакованных в одномерные динамические массивы.	7
Практическое занятие 5. Рекурсия и итерация.	8
Практическое занятие 6. Инварианты.	9
Практическое занятие 7. Метод индуктивной функции.	10
Практическое занятие 8. Метод индуктивной функции	11
Практическое занятие 9. Реализация класса «матрица» на основе двумерного динамического массива.	12