

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Казанский национальный исследовательский технический университет
им. А.Н. Туполева – КАИ»

Институт компьютерных технологий и защиты информации
Отделение СПО ИКТЗИ (Колледж информационных технологий)

ЛАБОРАТОРНАЯ РАБОТА № 5
по дисциплине

Основы алгоритмизации и программирования

Тема: «Программирование алгоритмов с использованием динамического
выделения памяти»

Работу выполнил
Студент гр.4235
Желваков А. С.

Принял
Преподаватель Шмидт. И.Р.

Казань 2023

ВАРИАНТ 9

1) Цель работы

Приобрести умения и практические навыки для программирования алгоритмов с использованием механизма динамического выделения памяти.

2) Задание на лабораторную работу

1. Написать программу, вычисляющую разность квадратов и разность кубов двух чисел. Расчет должен производиться в функции, которая получает данные по указателю и возвращает результат по указателю.
2. Написать программу, определяющую модуль максимального отклонения элементов массива от среднего значения всех элементов одномерного массива. При вводе/выводе элементов использовать индексы, а обработку элементов массива осуществлять с помощью указателей. Расчет должен производиться в функции, в которую массив передается по указателю
3. Написать программу, выполняющую отражение неквадратной матрицы относительно центральной вертикальной оси. Матрицы должны храниться в памяти в виде двумерного динамического массива, размерности исходной матрицы вводятся пользователем с клавиатуры. Предусмотреть генерацию значений матрицы как случайных вещественных чисел в диапазоне от -10 до 10

3) Результат выполнения работы

1.
 - Импортируем необходимые библиотеки.
 - Пишем функцию `calculateDifferences`, которая принимает два числа, указатель на переменную разницы квадратов и указатель на переменную разницы кубов. Делает соответствующие расчеты.
 - В функции `main` принимаем с клавиатуры два числа и вычисляем разницу квадратов и разницу кубов этих чисел.

```
Enter the 1st number: 3
Enter the 2nd number: 2
The difference of squares: 5
The difference of cubes: 19
```

Рис. 1 - Результат выполнения первой программы

2.

- Импортируем необходимые библиотеки.
- Пишем функцию `calculateMaxDeviation`, которая принимает указатель на массив чисел и размер этого массива. В функции вычисляем среднее арифметическое число массива и находим то число, которое максимально от него отдалено.
- В функции `main` запрашиваем размер массива и каждый элемент. Делаем соответствующие расчеты.

```
Enter the size of the array: 5
Enter the elements of the array:
arr[0]: 1
arr[1]: 2
arr[2]: 5
arr[3]: 6
arr[4]: 7
Absolute value of the maximum deviation: 3.2
```

Рис. 2 - Результат выполнения второй программы

3.

- Импортируем соответствующие библиотеки.
- Пишем функцию `generateRandomNumber`, которая возвращает число в диапазоне от -10 до 10.
- Пишем функцию `reflectMatrix`, которая принимает указатель на матрицу, количество строк и столбцов. Функция переворачивает матрицу по главной диагонали.
- Пишем функцию `printMatrix`, которая принимает указатель на матрицу, количество строк и столбцов. Функция выводит матрицу в консоль

- В функции `main` запрашиваем количество строк и столбцов, формируем матрицу из случайных чисел, выводим ее, переворачиваем и еще раз выводим в консоль.

```
Enter the number of rows: 3
Enter the number of columns: 4

Original Matrix:
2      -9      2      -8
-7      9      -2     10
-1     -4     -5      2

Reflected Matrix:
-8      2     -9      2
10     -2      9     -7
2     -5     -4     -1
```

Рис. 3 - Результат выполнения третьей программы

4) Ответы на контрольные вопросы.

1. Что такое указатель?

- Это переменная, которая содержит адрес в памяти другой переменной. Он указывает на местоположение в памяти, где хранится значение. Используется для эффективной работы с памятью и динамического выделения ресурсов.

2. Чем отличается статическое выделение памяти от динамического выделения памяти?

- Статическое выделение памяти происходит во время компиляции и предполагает, что размер памяти известен заранее. Динамическое выделение памяти, напротив, происходит во время выполнения программы с использованием операторов **new** и **delete**. Он позволяет управлять памятью более гибко, но требует явного освобождения выделенной памяти.

3. С помощью каких операций можно манипулировать над указателями?

- Разыменование (*): Получение значения, на которое указывает указатель.

- Взятие адреса (&): Получение адреса переменной.
- Арифметика указателей: Увеличение/уменьшение указателя, сложение/вычитание указателей, что особенно полезно при работе с массивами.

4. Может ли функция возвращать значение типа указатель? Если да, то как объявляется прототип такой функции?

- Да, функция может возвращать значение типа указатель.

Пример: `int* Func(int arg) {};`

5. Могут ли параметрами функции быть указатели на функции? Если да, то как происходит передача фактических параметров при вызове функции?

- Параметрами функции могут быть указатели на функции. При передаче фактических параметров происходит указание имени функции без круглых скобок (без вызова функции).

Например: `void Func(int (*otherFucn)(int, int)) {};`

6. Для чего применяются двойные указатели?

- Двойные указатели используются, когда требуется работать с указателями на указатели. Они часто применяются при работе с динамическими массивами и изменении значений указателей в функциях.

5) Листинг программы

Код на ГХ.

https://github.com/ArseniyZh/CIT/tree/main/2nd_year/OAIP/labs/5

Вывод:

В ходе выполнения лабораторной работы я приобрел умения и практические навыки программирования алгоритмов с использованием механизма динамического выделения памяти.

