

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів
Кафедра систем управління літальних апаратів

Лабораторна робота № 8

з дисципліни «Алгоритмізація та програмування»
на тему «Реалізація алгоритмів сортування та робота з файлами на мові C
++»

ХАІ.301.електроенергетика,електромеханіка і електротехніка
319а,№11ЛР

Виконав студент гр. 319а

| | |
|----------------|---------------|
| | Нікіта Єфімов |
| (підпис, дата) | (П.І.Б.) |

Перевірив

| | |
|-------------------------------------|----------|
| _____ к.т.н., доц. Олена ГАВРИЛЕНКО | |
| (підпис, дата) | (П.І.Б.) |

МЕТА РОБОТИ

масивів на мові

реалізувати оголошення,

одновимірних і двовимірних

Вивчити теоретичний матеріал по алгоритмам обробки

C++, а також бібліотеки для роботи з файлами і

введення з файлу, обробку і виведення в файл

масивів на мові C ++ в середовищі Visual Studio.

ПОСТАНОВКА ЗАДАЧІ

Завдання 1. За допомогою текстового редактору створити текстовий файл

«array_in_n.txt» з елементами вихідного масиву ($n - 11$). У

програмі на C++ перетворити масив ім'я файлу і необхідні змінні ввести з консолі. Вивести

результати у файл «array_out_n.txt». Дан масив розміру N і два цілих числа K і M ($1 \leq K \leq N$, $1 \leq M \leq 10$). Перед

елементом масиву з номером K вставити M нових елементів з нульовими

значеннями.

Щоб реалізувати це завдання, спочатку потрібно зчитати дані з файлу

"array_in_n.txt", а саме розмір масиву N та елементи масиву. Потім

користувачеві буде запропоновано ввести два цілих числа K і M , які

відповідають номеру позиції, перед якою треба вставити M нулів. Після вставки нулів масив буде записаний у файл "array_out_n.txt".

Ось код на мові C++ для вирішення цього завдання:

```
include <iostream>
include <fstream>
include <vector>
```

```
using namespace
```

```
int main
{
    int
```

```
        "Enter the size of the array (N): "
```

```
        int array
```

```

        inputFile "array_in_n.txt"
    if
        "Unable to open input file array_in_n.txt"
    return 1

    for int 0

        close

        "Enter the position (K) to insert zeros: "

        "Enter the number of zeros to insert (M): "

        insert begin 1 0

        outputFile "array_out_n.txt"
    if
        "Unable to open output file array_out_n.txt"
    return 1

    for int 0 size
        " "

        close

        "Zeros inserted successfully. Check array_out_n.txt for the result."

    return 0

```

Цей код спочатку зчитує розмір масиву N з консолі, а потім зчитує елементи масиву з файлу "array_in_n.txt". Після цього він запитує користувача ввести номер позиції K та кількість елементів M для вставки нулів. Потім він вставляє M нулів перед позицією K у масиві та записує оновлений масив у файл "array_out_n.txt".

Завдання 2.У

програмі на C++За допомогою текстового редактору створити текстовий файл

«matr_in_n.txt» з елементами вихідного двовимірного масиву ($n - 11$). У програмі обробити матрицю Дана матриця розміру $M \times N$. Знайти номер рядка і стовпчика для

елемента матриці, найбільш близького до середнього значення всіх її

елементів., ім'я файлу і необхідні змінні ввести з консолі.

Дописати результати в той же файл.Щоб вирішити це завдання, спочатку потрібно зчитати дані з файлу "matr_in_n.txt", а саме розмір матриці $M \times N$ та елементи матриці. Після цього знаходимо середнє значення всіх елементів матриці. Потім знаходимо елемент матриці, найближчий до середнього значення, та його рядок і стовпчик. Результати записуємо у файл "matr_in_n.txt".

Ось код на мові C++ для вирішення цього завдання:

```
include <iostream>
```

```
include <fstream>
```

```
include <vector>
```

```
include <cmath>
```

```
using namespace
```

```
int main
```

```
int
```

```
"Enter the number of rows (M): "
```

```
"Enter the number of columns (N): "
```

```
int matrix vector int
```

```
inputFile "matr_in_n.txt"
```

```
if
```

```
"Unable to open input file matr_in_n.txt"
```

```
return 1
```

```
for int 0
```

```
for int 0
```

```
close
```

```
double 0
```

```
for int 0
```

```
for int 0
```

```
double
```

```
int          0          0
```

```
double      abs      0  0
```

```
for int     0
```

```
for int     0
```

```
double      abs
```

```
if
```

```
outputFile "matr_in_n.txt"
```

```
if
```

```
"Unable to open output file matr_in_n.txt"
```

```

return 1

        "\nClosest element to the average ("          ") is: "

        "It is located at row "          1  " and column "
1

close

"Results appended to matr_in_n.txt"

return 0

```

Цей код зчитує розмір матриці $M \times N$ та елементи матриці з файлу "matr_in_n.txt". Після цього обчислює середнє значення всіх елементів матриці, знаходить елемент матриці, найближчий до середнього значення, та записує результати у файл "matr_in_n.txt".

Завдання 3.У

програмі на C++ 3. Вивчити метод сортування відповідно до свого варіанту , проаналізувати його складність і продемонструвати на прикладі з 7-ми

елементів

Метод сортування Порядок Тип елементів

11 Вставки Зростання Дійсний. Реалізувати у вигляді окремої функції

алгоритм сортування елементів масиву. Також окремими функціями

реалізувати зчитування масиву з текстового файлу і виведення відсортованого

масиву в консоль.

Метод сортування вставками (Insertion Sort) – це простий алгоритм сортування, який працює шляхом послідовного вставлення кожного елементу масиву в правильне місце в уже відсортованій частині масиву.

Ось код на мові C++ для реалізації методу сортування вставками та функцій зчитування масиву з файлу та виведення відсортованого масиву у консоль:

```
include <iostream>
```

```
include <fstream>
```

```
include <vector>
```

```
using namespace
```

```
void insertionSort      double
```

```
int      size
```

```
for int      1
```

```
double
```

```
int      1
```

```
while      0
```

```
1
```

```
1
```

```
1
```



```

double readArrayFromFile(const
double
inputFile
if
    "Unable to open input file "
exit 1

double
while
    push_back

    close

return

```

```

void printArray(const double
for double
    " "

```

```
int main

    double      readArrayFromFile "array_in.txt"

    "Original array: "

    printArray

    insertionSort

    "Sorted array: "

    printArray

    return 0
```

Складність сортування вставками в найгіршому, середньому та кращому випадках є $O(n^2)$, де n - кількість елементів у масиві. Враховуючи це, метод сортування

вставками найкраще підходить для невеликих масивів або вже відсортованих або майже відсортованих масивів.

Завдання 4.У

програмі на С++ Завдання 4. Для багаторазового виконання будь-якого з трьох зазначених

вище завдань на вибір розробити алгоритм організації меню в командному

вікні. Введення, виведення, обробку масивів реалізувати окремими функціями з

параметрами.

Ось приклад реалізації меню в командному вікні для вибору завдання (задачі) та виконання відповідної операції:

```
include <iostream>
```

```
include <vector>
```

```
include <fstream>
```

```
include <string>
```

```
using namespace
```

```
void insertionSort      double
```

```
int      size
```

```
for (int i = 1
```

```
double
```

```
int i = 1
```

```
while (i != 0
```

```
1
```

```
1
```

```
1
```

```
double readArrayFromFile(const
```

```
double
```

```
inputFile
```

```
if
```

```
"Unable to open input file "
```

```
exit(1)
```

```
double
```

```
while
```

```
    push_back
```

```
    close
```

```
return
```

```
void printArray const      double
```

```
for double
```

```
    " "
```

```
void processTask1
```

```
void processTask2
```

```
void processTask3
```

```
int main
```

```
int
```

```
do
```

```
    "==== Menu ====="
```

```
    "1. Task 1"
```

```
    "2. Task 2"
```

```
    "3. Task 3"
```

```
"4. Exit"
```

```
"Enter your choice: "
```

```
switch
```

```
case 1
```

```
processTask1
```

```
break
```

```
case 2
```

```
processTask2
```

```
break
```

```
case 3
```

```
processTask3
```

```
break
```

```
case 4
```

```
"Exiting the program."
```

```
break
```

```
default
```

```
"Invalid choice. Please enter a valid option."
```

```
while 4
```

```
return 0
```

У цьому коді використовується цикл `do-while` для відображення меню в командному вікні. Користувач вибирає номер опції, і виконується відповідна функція. Після виконання операції користувач може знову вибрати іншу опцію або вийти з програми.

ВИСНОВКИ

Після вивчення різних методів сортування в програмуванні, було обрано метод сортування вставками для реалізації завдання. Цей метод було успішно відпрацьовано в коді програми, яка була створена для сортування масиву. Під час виконання завдання виникли незначні труднощі з вибором оптимального алгоритму сортування, проте вони були вирішені шляхом дослідження та експериментів.