**­­Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет «Высшая школа экономики»**

Московский институт электроники и математики им. А.Н.Тихонова

НИУ ВШЭ

Департамент компьютерной инженерии (или департамент электронной инженерии)

**Курс: Алгоритмизация и программирование**

**ОТЧЁТ**

**по лабораторной работе №7**

|  |  |  |  |
| --- | --- | --- | --- |
| **Раздел** | **Маx**  **оценка** | **Итог.**  **оценка 1** | **Итог.**  **оценка 2** |
| Работа программы | 1 |  |  |
| Тесты | 1 |  |  |
| Правильность алгоритма | 3 |  |  |
| Ответы на вопросы | 2 |  |  |
| Доп. задания | 3 |  |  |

**Студент: Федоров Матвей Евгеньевич**

**Группа: БИВ243**

**Вариант: №208(12, 9)**

**Руководитель: Литвиненко Алексей Михайлович**

**Оценка: \_\_\_\_\_\_**

**Дата сдачи: \_**

**МОСКВА 2024**

**Содержание**

[Задание 3](#_Toc183088895)

[Листинг программы 4](#_Toc183088896)

[Распечатка тестов к программе и результатов 7](#_Toc183088897)

# **Задание**

* 1. Создать связанный список для хранения целых чисел. Число записей неизвестно, данные читаются из файла, имя которого задает пользователь. Имя выходного файла также задается пользователем. Входной файл содержит данные для исходного списка, выходной – для конечного списка. Для четных номеров вариантов организовать очередь, а для нечетных – стек.
  2. Для исходного списка решить следующую задачу: вставить заданное число A1 после каждого элемента с нечетным номером.
  3. Упорядочить по убыванию методом "установки" элементы списка, расположенные после первого нечетного элемента;

# **Листинг программы**

#include "stdio.h"

#include "stdlib.h"

#include "string.h"

#define maxLen 100

typedef char string[maxLen];

typedef struct node {

int num;

struct node **\***next;

} Node;

Node **\***getQueue(string *filename*) {

FILE **\***file **=** fopen(*filename*, "r");

if (**!**file) {

perror("Входной файл не найден.");

return NULL;

}

int curr\_el;

Node **\***first **=** NULL, **\***last **=** NULL;

while (fscanf(file, "%d", **&**curr\_el) **==** 1) {

Node **\***newNode **=** (Node**\***)malloc(**sizeof**(Node));

newNode->num **=** curr\_el;

newNode->next **=** NULL;

if (**!**first) {

first **=** last **=** newNode;

} else {

last->next **=** newNode;

last **=** newNode;

}

}

fclose(file);

return first;

}

Node **\***insertA(Node **\****queue*, int *a*) {

if (**!***queue*) return NULL;

int idx **=** 1;

Node **\***el **=** *queue*;

while (el) {

if (idx **%** 2 **!=** 0) {

Node **\***newNode **=** (Node **\***)malloc(**sizeof**(Node));

newNode->num **=** *a*;

newNode->next **=** el->next;

el->next **=** newNode;

if (newNode->next) {

el **=** newNode->next;

} else {

break;

}

} else {

el **=** el->next;

}

idx**++**;

}

return *queue*;

}

Node **\***sortQueue(Node **\****queue*) {

Node **\***el **=** *queue*;

while (el **&&** el->num **%** 2 **==** 0) {

el **=** el->next;

}

Node **\***sortedPart **=** el->next;

el->next **=** NULL;

while (sortedPart) {

Node **\***current **=** sortedPart;

sortedPart **=** sortedPart->next;

Node **\***prev **=** el;

while (prev->next **&&** prev->next->num **>** current->num) {

prev **=** prev->next;

}

current->next **=** prev->next;

prev->next **=** current;

}

return *queue*;

}

void printQueue(Node **\****queue*) {

Node **\***el **=** *queue*;

while (el) {

printf("%d ", el->num);

el **=** el->next;

}

printf("\n");

}

void freeQueue(Node **\****queue*) {

while (*queue*) {

Node **\***temp **=** *queue*;

*queue* **=** *queue*->next;

free(temp);

}

}

int main(void) {

string inputFile;

int a1;

printf("Введите имя входного файла: ");

scanf("%s", inputFile);

printf("Введите число A1: ");

scanf("%d", **&**a1);

Node **\***queue **=** getQueue(inputFile);

puts("Задание 1:");

printQueue(queue);

queue **=** insertA(queue, a1);

puts("Задание 2:");

printQueue(queue);

queue **=** sortQueue(queue);

puts("Задание 3:");

printQueue(queue);

freeQueue(queue);

}

# **Распечатка тестов к программе и результатов**

|  |  |  |
| --- | --- | --- |
| № | Исходные данные | Результаты |
| 1 | Введите имя входного файла: queue.txt  Введите число A1: 1 | Задание 1:  1 2 3 4 5 6 7 8 9 10  Задание 2:  1 1 2 3 1 4 5 1 6 7 1 8 9 1 10  Задание 3:  1 10 9 8 7 6 5 4 3 2 1 1 1 1 1 |
| 2 | Введите имя входного файла: ыфвафыва  Введите число A1: 12 | Ошибка открытия входного файла: No such file or directory |