МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
Кафедра «Вычислительной техники»

**ОТЧЁТ**

по лабораторной работе №3  
по курсу «Логика и основы алгоритмизации в инженерных задачах»  
на тему «Оценка времени выполнения программ»

Выполнил:

Студент группы 23ВВВ2

Попков С.А

Приняли:

Митрохин М. А.  
Юрова О.В.

Пенза 2024

**Цель работы**

Разработка и реализация структур данных, таких как список с приоритетом элементов, очередь и стек, с использованием базовых принципов программирования.

**Задание**

**Задание 1:**

Реализовать приоритетную очередь, путём добавления элемента в список в соответствии с приоритетом объекта (т.е. объект  с большим приоритетом становится перед объектом с меньшим приоритетом).

**Задание 2**:

На основе приведенного кода реализуйте структуру данных *Очередь*.

**Задание 3**:

На основе приведенного кода реализуйте структуру данных *Стек*.

**Ход работы**

Задание 1.

С использованием заранее приведённых функций внесли изменения для создания рабочей программы. Использовали int prior для приоритетного расположения элементов в списке.

struct node

{

char inf[256]; // полезная информация

int prior;

struct node\* next; // ссылка на следующий элемент

};

Добавили сортировку по приоритету при добавлении элемента.

if((struc->prior < p->prior)&&(n == NULL)){

struc->next = p;

last = p;

break;

}

else if ((struc->prior > p->prior)) {

p->next = struc;

if (x!=NULL)

{

x->next = p;

}

if (struc == head) {

head = p;

}

break;

}

else if((struc->prior <= p->prior)){

x = struc;

struc = struc->next;

}

**Задание 2.**

Структура данных *Очередь* была реализована на базе выполненной программы, с изменением функции удаления и записи элементов.

Удаляется только первый элемент списка.

void del()

{

struct node\* struc = head;

struct node\* prev=NULL;

int flag = 0

if (head == NULL)

{

printf("Список пуст\n");

return;

}

prev = struc->next;

head = prev;

free(struc);

}

Добавляется элемент же только в конец.

### Результаты работы программы

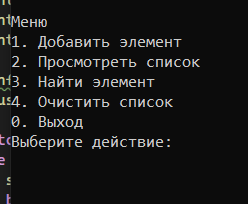


Рисунок 1 — Результаты работы программы пункта №1

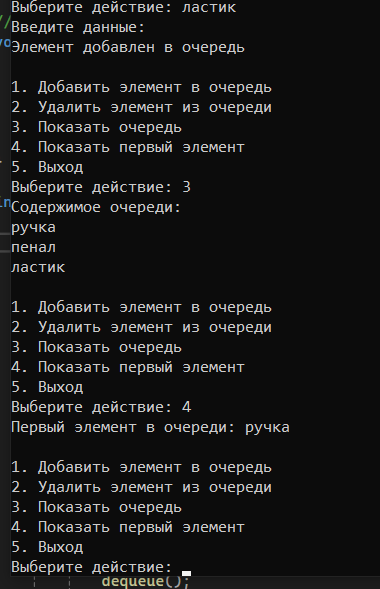


Рисунок 2 — Результаты работы программы пункта №2

### Вывод

В результате выполнения работы были успешно разработаны и реализованы три ключевые структуры данных: приоритетная очередь, очередь . Каждая из этих структур имеет свои уникальные свойства и области применения, что делает их важными инструментами в арсенале программиста. Освоение этих структур данных помогает лучше понимать алгоритмы и оптимизировать решения для различных задач.