

Московский авиационный институт  
(Национальный исследовательский университет)

# Курсовой проект

по курсу

«Языки и методы программирования»

2 семестр

Задание 8

Выполнил: Тимирчев Янис

Группа: М8О-111Б-21

Руководитель: Никулин С.П.

Оценка:

Дата:

Москва

2022г

## Задание

Составить программу на языке Си для обработки двунаправленного списка с отображением на динамические структуры. Предусмотреть реализацию 4 стандартных действий и одного нестандартного.

Стандартные:

1. Печать списка
2. Вставка нового элемента в список
3. Удаление элемента из списка
4. Подсчёт длины списка

Тип элемента списка - строковый

Вид списка - линейный двунаправленный

Нестандартное действие - удалить элементы списка со значениями, находящимися в заданном диапазоне(в качестве диапазона выбрана первая буква строки)

Программа должна вводить значения элементов неупорядоченной таблицы и проверять работу процедуры сортировки в трёх случаях:

## Структура проекта.

Проект состоит из 3 файлов:

- 1) **main.c** - Меню.
- 2) **list.c** - Функции списка
- 3) **list.h** - заголовочный файл с описанием структуры.

## Программный код и текстовые данные.

**main.c:**

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "ctype.h"
#include "lsst.h"

void printCommands() {
    printf("1: Напечатать список\t2: Вставить новый элемент в список\n3: Удалить элемент из списка\t4: Посчитать длину списка\n5: Удалить значения из списка в диапазоне\t6: Выйти\nВведите номер операции\nНомер == ");
}

int is_empty(Dblist *lst){
    return lst->size == 0;
}

int main(void) {
    Dblist *list = createList();
    int flag = 1, rule, c = 1, ind;
    char sym, a, b;
    char *string;// Объявление строки, которая будет заполняться символами
    int i = 0;
    bool countSpace = false;
    while (flag) {
        printCommands();
        scanf("%d", &rule);
        switch (rule) {
            case 1:
                if (!is_empty(list)){
                    printList(list);
                }
                else{
                    printf("\nСписок пуст!\n");
                }
                break;
            case 2:
                getchar();
                string = (char *)malloc(sizeof(char));
                while (c) {
                    sym = getchar();
                    if (sym == '\n'){
                        string[i] = '\0';
                        addElem(list, string); //добавление строки в список
                        i = 0;
                        break;
                    }
                }
                break;
        }
    }
}
```

```

    }
    else if (sym >= 35 && sym <= 126){
        string[i] = sym;
        i++;
        string = realloc(string, sizeof(char) * (i + 1));
    }
}
// string = (char *)realloc(string, sizeof(char) * (i + 1));
break;
case 3:
    if (!is_empty(list)){
        printf("Введите индекс элемента: ");
        scanf("%d", &ind);
        if (ind < list->size && ind >= 0){
            delElem(list, ind);
            printf("\nГотово\n\n");
        }
        else{
            printf("\nОшибка, введён некорректный индекс!\n\n");
        }
    }
    else{
        printf("\nСписок пуст!\n");
    }

    break;
case 4:
    if (!is_empty(list)){
        printf("Длина списка: %d\n\n", list->size);
    }
    else{
        printf("Список пуст\n\n");
    }
    break;
case 5:
    printf("Введите значения диапазона в формате [a, b]\nВводите символы с
которых должна начинаться строка\n");
    getchar();
    printf("Введите первое значение диапазона: ");
    scanf("%c", &a);
    getchar();
    printf("\nВведите второе значение диапазона: ");
    scanf("%c", &b);
    if (tolower(a) <= tolower(b)){
        delDiap(list, tolower(a), tolower(b));
        printf("\nГотово\n\n");
    }

    break;

```

```

        case 6:
            flag = 0;
            break;
        }
    }
}

```

#### **table.c:**

```

#include "lsst.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

```

```

Dblist* createList() {
    Dblist *tmp = (Dblist
    *)malloc(sizeof(Dblist));
    tmp->size = 0;
    tmp->head = tmp->end =
    NULL;

    return tmp;
}

```

```

//Добавление в конец списка
void addElem(Dblist *list, char
    *value) {
    element *tmp = (element
    *)malloc(sizeof(element));
    tmp->str = (char
    *)malloc(sizeof(char) *
    (strlen(value) + 1));
    tmp->str = strcpy(tmp->str,
    value);
    tmp->next = NULL;
    tmp->prev = list->end;

    if (list->end) {
        list->end->next = tmp;
    }
}

```

```

list->end = tmp;

if (list->head == NULL) {
    list->head = tmp;
}

list->size++;
}

void delElem(Dblist *list, int
index){
    element *side = list->head;
    int flag = 0;
    if (list->size != 0 && side !=
NULL){
        int flag = 0;
        for(int i=0; i<list->size;i++){
            if (i == index){
                if (side != list->end &&
side != list->head){
                    side->prev->next =
side->next;
                    side->next->prev =
side->prev;
                    free(side);
                    flag = 1;
                    break;
                }
                else if(side == list->end
&& side != list->head){
                    side->prev->next =
NULL;
                    list->end = side-
>prev;
                    free(side);
                    flag = 1;
                    break;
                }
                else if (side == list-
>head && side != list->end){
                    side->next->prev =

```

```

NULL;
        list->head = side-
>next;
        free(side);
        flag = 1;
        break;
    }
    else if(list->size == 1){
        free(side);
        list->head = list->end
= NULL;
        flag = 1;
        break;
    }
}
else{
    side = side->next;
}
}
if (flag){
    list->size--;
}
}
else{
    printf("Список пуст\n");
}
}

```

```

void delDiap(Dblist *list, char a,
char b){
    element *now = list->head;
    for(int i = 0; i<list->size;i++){
        // printf("%c", now->str[0]);
        char check = now->str[0];
        if (check >= a && check <=
b){
            if (i == 0){
                delElem(list, i);
                i = -1;
            }
            else{

```



```
        delElem(list, i);
        i--;
    }
}
now = now->next;
}
}
```

//Вывод списка

```
void printList(Dblist *list) {
    element *tmp = list->head;

    while (tmp) {
        printf("%s ", tmp->str);
        tmp = tmp->next;
    }
    printf("\n");
}
```

**list.h:**

```
#ifndef __lsst_h__
#define __lsst_h__
#include <stdio.h>
#include "stdlib.h"
#include <stdbool.h>

typedef struct element {
    char *str;
    struct element *prev;
    struct element *next;
} element;

typedef struct Dblist {
    int size;
    element *head;
    element *end;
} Dblist;

Dblist* createList();
void addElem(Dblist *list, char *value);
void printList(Dblist *list);
void delElem(Dblist *list, int index);
void delDiap(Dblist *list, char a, char b);

#endif
```

## Формат программы и примеры работы.

□ ./main

1: Напечатать список    2: Вставить новый элемент в список  
3: Удалить элемент из списка    4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне    6: Выйти  
Введите номер операции  
Номер == 1

Список пуст!

1: Напечатать список    2: Вставить новый элемент в список  
3: Удалить элемент из списка    4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне    6: Выйти  
Введите номер операции  
Номер == 4  
Список пуст

1: Напечатать список    2: Вставить новый элемент в список  
3: Удалить элемент из списка    4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне    6: Выйти  
Введите номер операции  
Номер == 2  
abcdefg

1: Напечатать список    2: Вставить новый элемент в список  
3: Удалить элемент из списка    4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне    6: Выйти  
Введите номер операции  
Номер == 1  
abcdefg

1: Напечатать список    2: Вставить новый элемент в список  
3: Удалить элемент из списка    4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне    6: Выйти  
Введите номер операции  
Номер == 2  
bbc

1: Напечатать список    2: Вставить новый элемент в список  
3: Удалить элемент из списка    4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне    6: Выйти  
Введите номер операции  
Номер == 2  
cdef

1: Напечатать список    2: Вставить новый элемент в список  
3: Удалить элемент из списка    4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне    6: Выйти  
Введите номер операции  
Номер == 4  
Длина списка: 3

1: Напечатать список    2: Вставить новый элемент в список  
3: Удалить элемент из списка    4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне    6: Выйти  
Введите номер операции  
Номер == 1  
abcdefg bbc cdef

1: Напечатать список    2: Вставить новый элемент в список  
3: Удалить элемент из списка    4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне    6: Выйти  
Введите номер операции  
Номер == 3  
Введите индекс элемента: 1

Готово

1: Напечатать список 2: Вставить новый элемент в список  
3: Удалить элемент из списка 4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 1

abcdefg cdef

1: Напечатать список 2: Вставить новый элемент в список  
3: Удалить элемент из списка 4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 5

Введите значения диапазона в формате [a, b]

Вводите символы с которых должна начинаться строка

Введите первое значение диапазона: c

Введите второе значение диапазона: c

Готово

1: Напечатать список 2: Вставить новый элемент в список  
3: Удалить элемент из списка 4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 1

abcdefg

1: Напечатать список 2: Вставить новый элемент в список  
3: Удалить элемент из списка 4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 2

ff

1: Напечатать список 2: Вставить новый элемент в список  
3: Удалить элемент из списка 4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 2

ggg

1: Напечатать список 2: Вставить новый элемент в список  
3: Удалить элемент из списка 4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 2

hhh

1: Напечатать список 2: Вставить новый элемент в список  
3: Удалить элемент из списка 4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 1

abcdefg ff ggg hhh

1: Напечатать список 2: Вставить новый элемент в список  
3: Удалить элемент из списка 4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 4

Длина списка: 4

1: Напечатать список 2: Вставить новый элемент в список  
3: Удалить элемент из списка 4: Посчитать длину списка

5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 5

Введите значения диапазона в формате [a, b]

Вводите символы с которых должна начинаться строка

Введите первое значение диапазона: b

Введите второе значение диапазона: r

Готово

1: Напечатать список 2: Вставить новый элемент в список

3: Удалить элемент из списка 4: Посчитать длину списка

5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 1

abcdefg

1: Напечатать список 2: Вставить новый элемент в список

3: Удалить элемент из списка 4: Посчитать длину списка

5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 2

b

1: Напечатать список 2: Вставить новый элемент в список

3: Удалить элемент из списка 4: Посчитать длину списка

5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 2

c

1: Напечатать список 2: Вставить новый элемент в список

3: Удалить элемент из списка 4: Посчитать длину списка

5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 2

d

1: Напечатать список 2: Вставить новый элемент в список

3: Удалить элемент из списка 4: Посчитать длину списка

5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 2

e

1: Напечатать список 2: Вставить новый элемент в список

3: Удалить элемент из списка 4: Посчитать длину списка

5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 1

abcdefg b c d e

1: Напечатать список 2: Вставить новый элемент в список

3: Удалить элемент из списка 4: Посчитать длину списка

5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 4

Длина списка: 5

1: Напечатать список 2: Вставить новый элемент в список

3: Удалить элемент из списка 4: Посчитать длину списка

5: Удалить значения из списка в диапазоне 6: Выйти

Введите номер операции

Номер == 5

Введите значения диапазона в формате [a, b]

Вводите символы с которых должна начинаться строка

Введите первое значение диапазона: b

Введите второе значение диапазона: e

Готово

1: Напечатать список   2: Вставить новый элемент в список  
3: Удалить элемент из списка   4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне   6: Выйти

Введите номер операции

Номер == 1

abcdefg

1: Напечатать список   2: Вставить новый элемент в список  
3: Удалить элемент из списка   4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне   6: Выйти

Введите номер операции

Номер == 2

b

1: Напечатать список   2: Вставить новый элемент в список  
3: Удалить элемент из списка   4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне   6: Выйти

Введите номер операции

Номер == 2

c

1: Напечатать список   2: Вставить новый элемент в список  
3: Удалить элемент из списка   4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне   6: Выйти

Введите номер операции

Номер == 1

abcdefg b c

1: Напечатать список   2: Вставить новый элемент в список  
3: Удалить элемент из списка   4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне   6: Выйти

Введите номер операции

Номер == 5

Введите значения диапазона в формате [a, b]

Вводите символы с которых должна начинаться строка

Введите первое значение диапазона: a

Введите второе значение диапазона: z

Готово

1: Напечатать список   2: Вставить новый элемент в список  
3: Удалить элемент из списка   4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне   6: Выйти

Введите номер операции

Номер == 1

Список пуст!

1: Напечатать список   2: Вставить новый элемент в список  
3: Удалить элемент из списка   4: Посчитать длину списка  
5: Удалить значения из списка в диапазоне   6: Выйти

Введите номер операции

Номер == 6

## **Заключение.**

Я научился составлять программы на языке Си с использованием процедур и функций для работы с списками, научился их реализовывать.