

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных средств

ОТЧЕТ

по лабораторной работе №6

«Очереди»

Выполнил
студ. гр.150702 (пг1)
Ефимчик А.И.

Проверил
ст. преп. каф. ЭВС
Демидович Г.Н.

Минск 2022

1 ЦЕЛЬ И ЗАДАЧИ ЛАБОРАТОРНОЙ РАБОТЫ

1.1 Целью лабораторной работы является изучение структуры данных "Очередь" и формирование практических навыков разработки алгоритмов и компьютерных программ с использованием очередей.

1.2 Для достижения поставленной цели необходимо решить следующие задачи:

- 1) Изучить лекционный материал по теме «Очередь» [1]
- 2) Дополнить и расширить сведения по теме ЛР из учебного пособия [2].

1.3 Выполнить следующие задания по ЛР в соответствии с вариантом №5, разработав алгоритмы их реализации, запрограммировав их с использованием языка «Си», отладив и представив результаты работы компьютерных программ.

Задание 1 Организовать три очереди с одинаковым количеством элементов, содержащие соответственно имена, отчества и фамилии людей. Составьте очередь из элементов, содержащих полную информацию о людях, воспользовавшись уже созданными очередями и запросив дополнительную информацию.

2 РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

2.1 Результат выполнения задания 1

2.1.1 На рисунке 1 приведена блок-схема алгоритма для выполнения задания №1 (сформировать очередь, содержащую полную информацию о людях, а так же дополнительно введенную информацию).

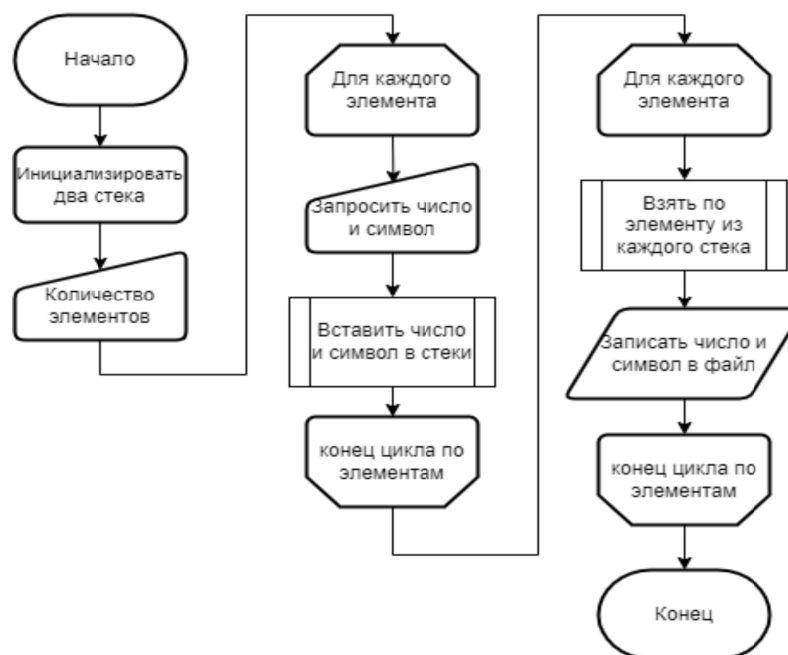


Рисунок 1 - Блок-схема алгоритма функции, выполняющей создание очереди, содержащей ФИО на основе данных из 3 исходных очередей

2.1.2 Листинг компьютерной программы по заданию 1 (создать другой текстовый файл, в котором числа и слова чередовались, а порядок чисел и слов был бы сохранен) .

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
typedef struct Node
{
    char *value;
    struct Node *next;
} Node;

typedef Node *NodePtr;

char *inputString();
void pushElement(NodePtr *srcRoot, char *element);
char *popElement(NodePtr *srcRoot);
void printList(NodePtr current);

int main()
{
    printf("LAB 6.1 by Efimchik Alexandr from GROUP 150702 \
n");

    NodePtr fName_root = NULL;
    NodePtr lName_root = NULL;
    NodePtr mName_root = NULL;

    int person_num;
    printf("Enter person num: ");
    scanf("%d", &person_num);

    for (int i = 0; i < person_num; i++)
    {
        printf("Enter %d first name: ", i);
        pushElement(&fName_root, inputString());
        printf("Enter %d last name: ", i);
        pushElement(&lName_root, inputString());
        printf("Enter %d middle name: ", i);
        pushElement(&mName_root, inputString());
    }
```

```

    }

    NodePtr DoB_root = NULL;
    NodePtr final_root = NULL;
    for (int i = 0; i < person_num; i++)
    {
        char *first_name = popElement(&fName_root);
        char *last_name = popElement(&lName_root);
        char *middle_name = popElement(&mName_root);
        printf("Enter date of birth for %s %s: ", first_name,
last_name);
        char *dOb = inputString();

        size_t final_len = strlen(first_name) +
strlen(last_name) + strlen(middle_name) + strlen(dOb) + 3;
        char *person = (char*)calloc(sizeof(char) * final_len,
1);
        strcat(person, first_name);
        strcat(person, " ");
        strcat(person, last_name);
        strcat(person, " ");
        strcat(person, middle_name);
        strcat(person, " ");
        strcat(person, dOb);
        pushElement(&final_root, person);
    }
    printList(final_root);
    return 0;
}

void pushElement(NodePtr *srcRoot, char *element)
{
    NodePtr newElement = (NodePtr)calloc(sizeof(Node), 1);
    newElement->value = element;
    newElement->next = NULL;
    NodePtr *temp = srcRoot;
    if ((*srcRoot) == NULL) {
        *srcRoot = newElement;
        return;
    }
    while ((*temp)->next != NULL)
    {

```

```

    temp = &(*temp)->next;
}
(*temp)->next = newElement;

}

char *popElement(NodePtr *srcRoot)
{
    char *value;
    if ((*srcRoot)->next == NULL) {
        value = (*srcRoot)->value;
        (*srcRoot) = NULL;
        return value;
    }
    value = (*srcRoot)->value;
    NodePtr backupRoot = *srcRoot;
    (*srcRoot) = backupRoot->next;
    free(backupRoot);
    return value;
}

void printList(NodePtr current)
{
    printf("Printed list: \n");
    if (current == NULL)
    {
        printf("Empty list\n");
        return;
    }
    while (current != NULL)
    {
        printf("%s -> ", current->value);
        current = current->next;
    }
    printf("\nlist end\n");
}

char *inputString()
{
    char *current_word;
    current_word = calloc(sizeof(char), 1);
    int word_size = 0;

```

```

char current_char = 0;
current_char = getc(stdin);
if (current_char != '\n') {
    current_word = realloc(current_word, sizeof(char) *
(word_size + 1));
    current_word[word_size] = current_char;
    word_size++;
}
while (1)
{
    current_char = getc(stdin);
    if (current_char == '\n')
    {
        break;
    }
    current_word = realloc(current_word, sizeof(char) *
(word_size + 1));
    current_word[word_size] = current_char;
    word_size++;
};
current_word = realloc(current_word, sizeof(char) *
(word_size + 1));
current_word[word_size] = '\0';
return current_word;
}

```

2.2.3 Результат выполнения компьютерной программы по заданию 2 в виде «скрин-шот» изображения на мониторе.

```

PS C:\Users\Alexandr\Documents\CODE\BSUIR_OAIP\Semester_2\LAB6> .\6_1.exe
LAB 6.1 by Efimchik Alexandr from GROUP 150702
Enter person num: 2
Enter 0 first name: Ivan
Enter 0 last name: Ivanov
Enter 0 middle name: Fedorovich
Enter 1 first name: Abduhamed
Enter 1 last name: Abragimovich
Enter 1 middle name: Kormich
Enter date of birth for Ivan Ivanov: 10.02.2000
Enter date of birth for Abduhamed Abragimovich: 15 May 1998
Printed list:
Ivan Ivanov Fedorovich 10.02.2000 -> Abduhamed Abragimovich Kormich 15 May 1998 ->
list end

```

Рисунок 2 - скриншот результата выполнения программы, создающей очередь, содержащую ФИО на основе данных из 3 исходных очередей

2. ВЫВОДЫ ПО РЕЗУЛЬТАТАМ ВЫПОЛНЕНИЯ ЛР

В результате выполнения ЛР изучены методы работы с очередями (добавление элемента, считывания значения элемента), получены практические навыки по написанию функций работы с очередями на языке С.

3 КОНТРОЛЬНЫЕ ВОПРОСЫ И ПРЕДЛОЖЕНИЯ

3.1 Пояснить основные положения, термины и определения в материалах лекции (лекций) и литературе по теме ЛР.

3.2 Объяснить алгоритмы выполнения заданий, указанных в данном варианте ЛР.

3.3 Прокомментировать листинги (фрагменты листингов) компьютерных программ в данном варианте ЛР.

3.4 Прокомментировать результаты выполнения заданий, указанных в варианте ЛР.

4 СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Лекция «Очередь» . Конспект лекций (*личный*) по дисциплине ОАиП, Минск, БГУИР, 2022.
2. Луцик, Ю. А. Основы алгоритмизации и программирования : язык Си : учебно-метод. пособие / Ю. А. Луцик, А. М. Ковальчук, Е. А. Сасин. – Минск : БГУИР, 2015. – 170с. : ил.