|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ *Робототехники и комплексной автоматизации*

КАФЕДРА *Системы автоматизированного проектирования (РК-6)*

**ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

по дисциплине: «Основы программирования»

|  |  |  |
| --- | --- | --- |
| Студент |  | Камалов Антон Павлович |
| Группа |  | РК6-15Б |
| Тип задания |  | Лабораторная работа №11 |
| Тема лабораторной работы |  | Стек |

Студент **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_ \_**

*подпись, дата фамилия, и.о.*

Преподаватель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_ \_\_**

*подпись, дата фамилия, и.о.*

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Москва, 2022 г.*

## Цель работы

## Изучение применения функций управления динамической памятью для организации динамического стека.

## Ответы на контрольные вопросы

1. Стек – динамическая структура, в которой элемент можно добавлять только в конец, удалять только последний.
2. Структура – это набор элементов, которые могут иметь различные типы.
3. Структура отличается от массива агрегатным типом данных, то есть, в отличие от массива, в структуре могут храниться различные типы данных.
4. Шаблон структуры – список объявлений переменных, описывающий элементы структуры.
5. Объявить структуру можно:

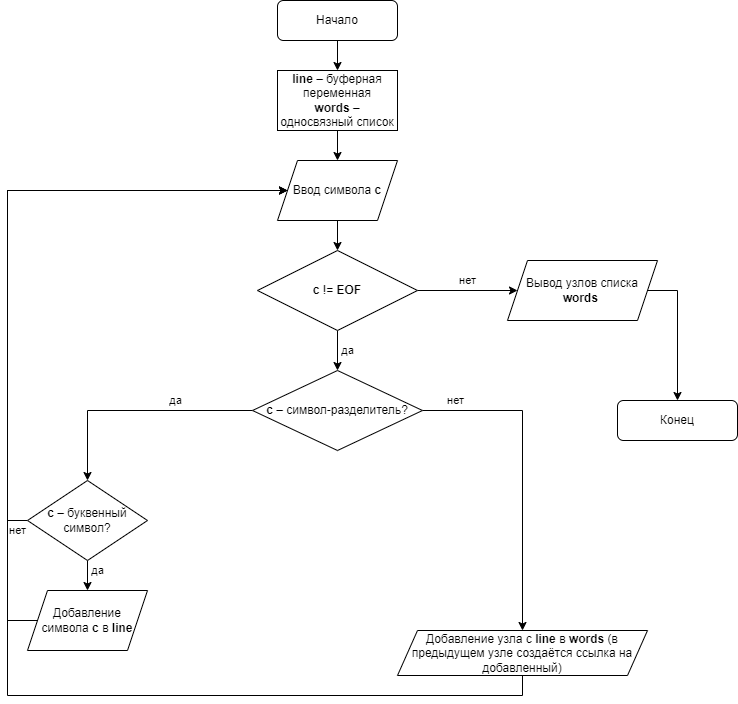
• Сразу после объявления шаблона (если вне функции, то она будет внешней);

• В любом месте программы с указанием имени структуры и ключевого слова;

• В любом месте программы с использованием имени, заданного через typedef.

1. Линейный односвязный список организуется в виде структуры, содержащей данные и адрес предыдущего элемента структуры. При этом адрес, записанный в начальном элементе равен нулю.

## Блок-схема программы



## Текст программы

## #include <stdio.h>

## #include <stdlib.h>

## #include <string.h>

## #include <ctype.h>

## #define LINE\_LEN 4096

## typedef struct Node {

## char \* string;

## struct Node \* next;

## }

## WORD;

## WORD \* in ();

## void push(WORD \*\* , char \*);

## void printList(WORD \*);

## int main() {

## WORD \* words = NULL;

## words = in ();

## printf("\nLines: \n");

## printList(words);

## }

## WORD \* in () {

## WORD \* head = NULL;

## int j = 0, c;

## char line[LINE\_LEN], \* elem;

## while ((c = getchar()) != EOF) {

## if (c != ' ' && c != '\n' && c != '\t') {

## if (isalpha(c)) line[j++] = c;

## } else {

## line[j] = '\0';

## j = 0;

## if (strlen(line) == 0) continue;

## if ((elem = (char \*) malloc(strlen(line))) == NULL) {

## printf("NOT ENOUGH MEMORY!\n");

## exit(-1);

## }

## strcpy(elem, line);

## push(&head, elem);

## }

## }

## if (j > 0 && strlen(line) != 0) {

## line[j] = '\0';

## if ((elem = (char \*) malloc(strlen(line))) == NULL) {

## printf("NOT ENOUGH MEMORY!\n");

## exit(-1);

## }

## strcpy(elem, line);

## push(&head, elem);

## }

## return head;

## }

## void push(WORD \*\* headRef, char \* line) {

## WORD \* newWord;

## if ((newWord = (WORD \*) malloc(sizeof(WORD))) == NULL) {

## printf("NOT ENOUGH MEMORY!\n");

## exit(-1);

## }

## newWord -> string = line;

## newWord -> next = \* headRef;

## \* headRef = newWord;

## }

## void printList(WORD \* head) {

## while (head) {

## WORD \* p = head;

## printf("%s\n", p -> string);

## head = p -> next;

## free(p);

## }

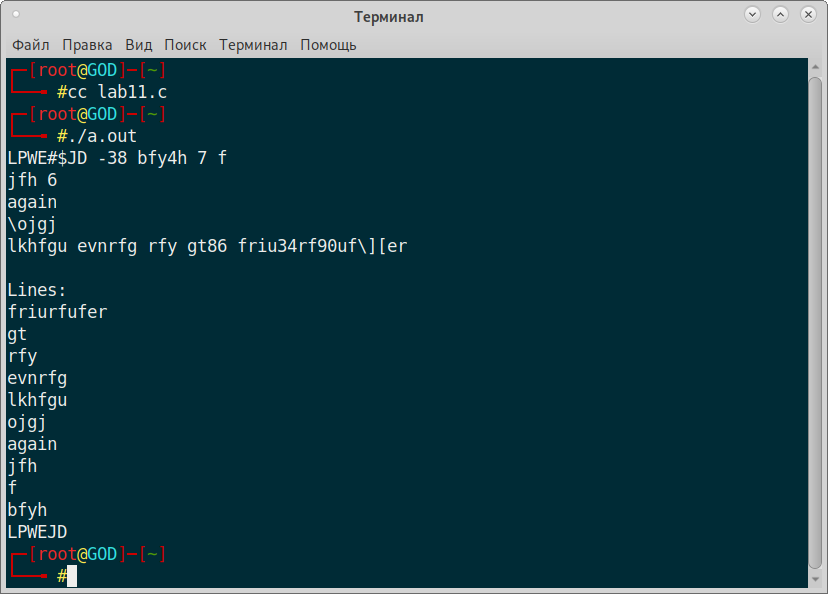
## free(head);

## }

## Результаты отладки

После первой компиляции была найдена синтаксическая ошибка: пропущена «;» после инициализации переменной ***words***.

## Результаты работы программы



## Выводы по работе

Я изучил применение функций управления динамической памятью для организации динамического стека.