

Министерство образования и науки Российской Федерации
Федеральное агентство по образованию
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Вятский государственный университет»

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Лабораторная работа №4
по курсу «Программирование»

**Реализация элементарных структур данных на основе статической
памяти**

Выполнил студент группы ИВТ-11 _____/Рзаев А. Э./
Проверил преподаватель _____/Чистяков Г. А./

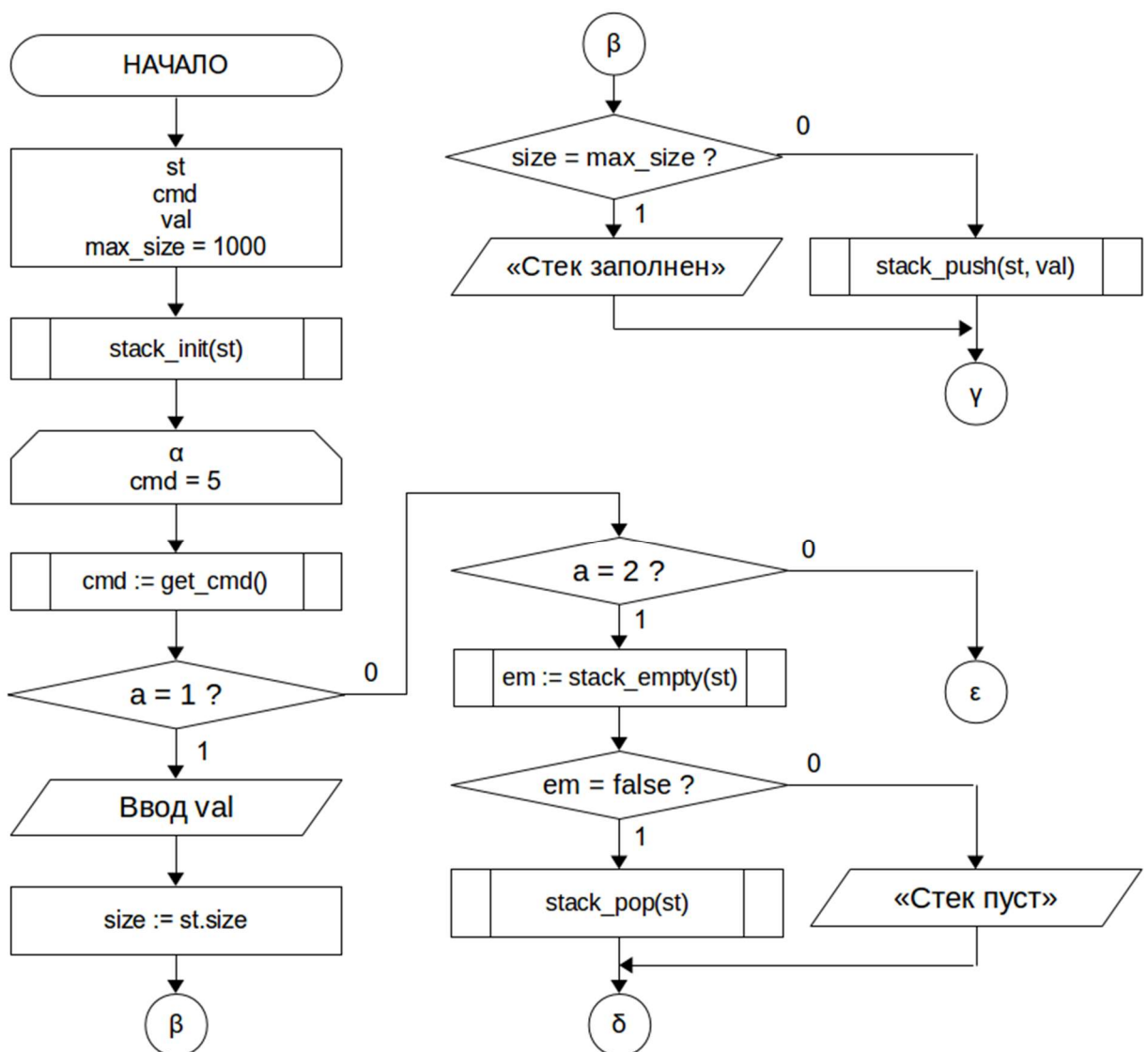
Киров 2015

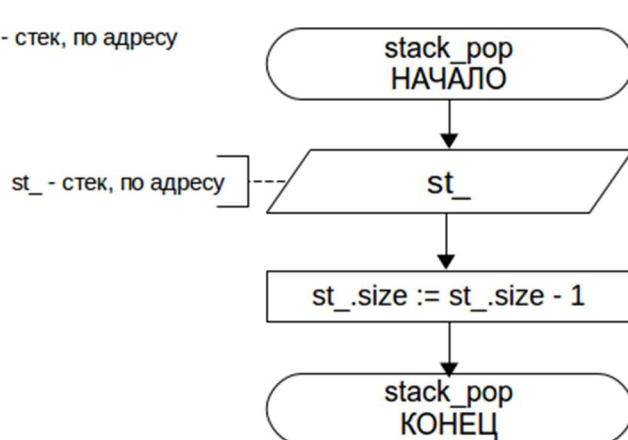
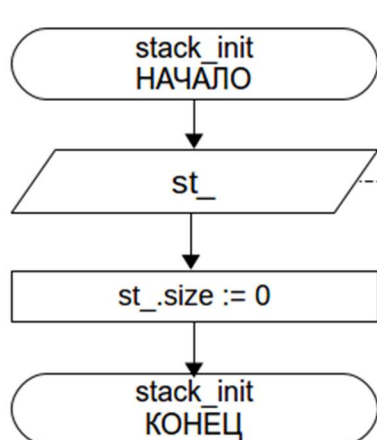
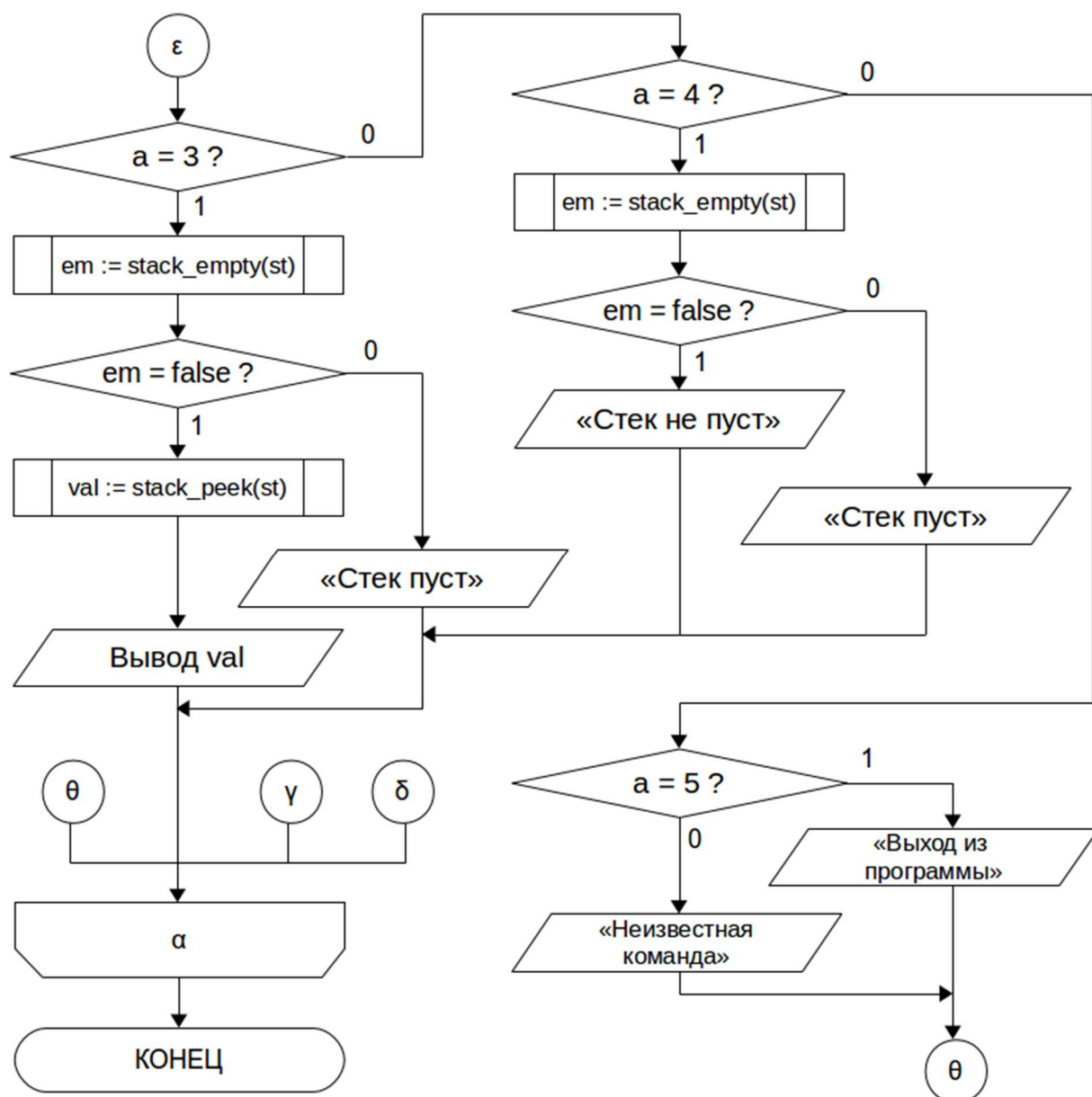
Цель работы: закрепить навыки по работе с массивами, освоить элементарные структуры данных, получить представление об организации текстового пользовательского меню.

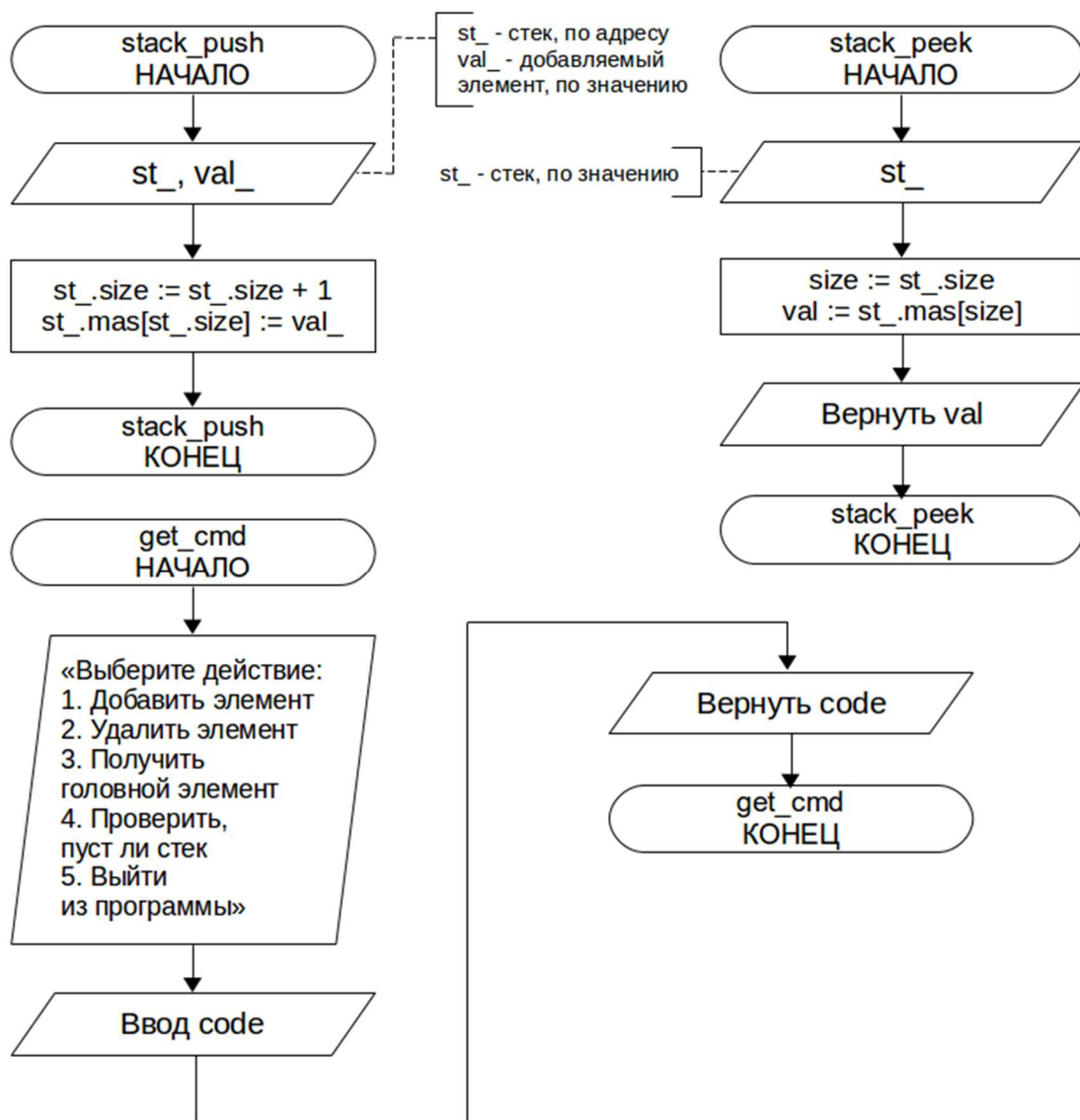
Задание:

1. Написать программу для работы со структурой данных "Стек".
2. Структура данных должна быть реализована на основе статических массивов.
3. Работа со структурой должна осуществляться с помощью текстового пользовательского меню.

Схема алгоритма:







Листинг кода:

```

uses crt;

const max_size = 1000;

type value_type = Integer;

type seq = array[1..max_size] of value_type;

type stack = record
    size : Integer;
    container : seq;
end;

procedure stack_init(var stack_ : stack);
begin
    stack_.size := 0;
  
```

```

end;

procedure stack_push(var stack_ : stack; value_ : value_type);
begin
    inc(stack_.size);
    stack_.container[stack_.size] := value_;
end;

procedure stack_pop(var stack_ : stack);
begin
    dec(stack_.size);
end;

function stack_peek(stack_ : stack): value_type;
begin
    stack_peek := stack_.container[stack_.size];
end;

function stack_empty(stack_ : stack): Boolean;
begin
    if (stack_.size = 0) then
        stack_empty := True
    else
        stack_empty := False;
    end;
end;

function get_cmd(): Integer;
var code : Integer;
begin
    writeln('What do you want to do:', chr(10), chr(13),
        '1. Push element in the stack', chr(10), chr(13),
        '2. Pop element from the stack', chr(10), chr(13),
        '3. Get top element of the stack', chr(10), chr(13),
        '4. Check the stack for empty', chr(10), chr(13),
        '5. Exit');
    readln(code);
    get_cmd := code;
end;

var astack : stack;
    cmd : Integer;
    value : value_type;
begin
    stack_init(astack);
    cmd := 0;
    while (cmd <> 5) do
        begin
            clrscr;
            cmd := get_cmd();
            case cmd of
                1 :
                    begin
                        writeln('Enter a value');
                        readln(value);

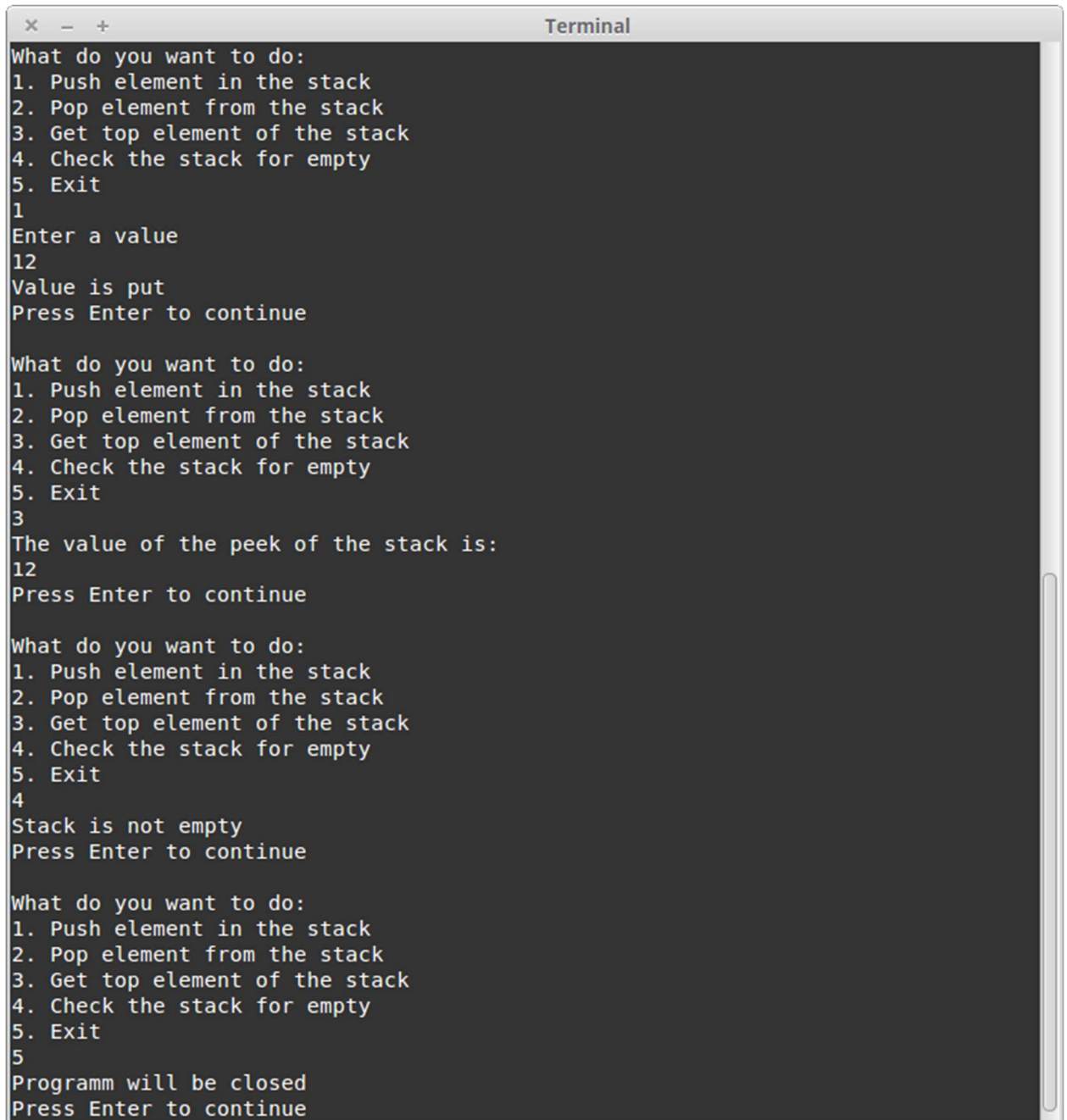
```

```

        if (astack.size = max_size) then
            writeln('Stack is completely filled')
        else
            begin
                stack_push(astack, value);
                writeln('Value is put');
            end;
        end;
    2 :
    begin
        if (stack_empty(astack)) then
            writeln('Stack is already empty')
        else
            begin
                stack_pop(astack);
                writeln('Element popped');
            end;
        end;
    3 :
    begin
        if (stack_empty(astack)) then
            writeln('Stack is already empty')
        else
            begin
                writeln('The value of the peek of the stack is: ');
                value := stack_peek(astack);
                writeln(value);
            end;
        end;
    4 :
    begin
        if (stack_empty(astack)) then
            writeln('Stack is empty')
        else
            writeln('Stack is not empty');
        end;
    5 :
        writeln('Programm will be closed');
    else
        writeln('Unknown command!');
    end;
    writeln('Press Enter to continue');
    readln;
end;
end.

```

Экранная форма:

A screenshot of a terminal window titled "Terminal". The window has a dark background with light-colored text. The text shows a menu-driven program for a stack. The menu options are: 1. Push element in the stack, 2. Pop element from the stack, 3. Get top element of the stack, 4. Check the stack for empty, 5. Exit. The user has entered '1' to push the value '12'. The program confirms the push and asks to press Enter to continue. The user then enters '3' to peek at the stack, and the program displays 'The value of the peek of the stack is: 12' before asking to press Enter to continue. Next, the user enters '4' to check if the stack is empty, and the program responds 'Stack is not empty' before asking to press Enter to continue. Finally, the user enters '5' to exit, and the program displays 'Programm will be closed' before asking to press Enter to continue.

```
x - + Terminal
What do you want to do:
1. Push element in the stack
2. Pop element from the stack
3. Get top element of the stack
4. Check the stack for empty
5. Exit
1
Enter a value
12
Value is put
Press Enter to continue

What do you want to do:
1. Push element in the stack
2. Pop element from the stack
3. Get top element of the stack
4. Check the stack for empty
5. Exit
3
The value of the peek of the stack is:
12
Press Enter to continue

What do you want to do:
1. Push element in the stack
2. Pop element from the stack
3. Get top element of the stack
4. Check the stack for empty
5. Exit
4
Stack is not empty
Press Enter to continue

What do you want to do:
1. Push element in the stack
2. Pop element from the stack
3. Get top element of the stack
4. Check the stack for empty
5. Exit
5
Programm will be closed
Press Enter to continue
```

Вывод: В данной лабораторной работе были закреплены навыки по работе с массивами; освоена элементарная структура данных «стек», позволяющая выполнять операции добавления и удаления элемента, получения головного элемента, проверки стека на заполненность; освоен способ создания пользовательских типов данных с помощью записей.