МИНИСТЕРСТВО НАУКИ И ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«Вятский государственный университет»**

**(ФГБОУ ВО «ВятГУ»)**

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Отчет по лабораторной работе 4

по дисциплине

«Технологии программирования»

Разработал студент группы ИВТб-21 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Седов М.Д./

(подпись)

Проверил преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Долженкова М.Л./

(подпись)

Киров 2019**Постановка задачи**

Используя язык Assembler реализовать функцию, которая находит минимальные расстояния между вершинами графа.

**Стекфрейм на момент вызова**

|  |  |
| --- | --- |
| Содержимое регистра EBP | EBP |
| Адрес возврата | EBP+4 |
| Адрес одномерного массива | EBP+8 |
| Количество элементов | EBP+12 |

**Листинг**

.586

.MODEL FLAT, C

.CODE

PUBLIC minDist

minDist PROC

push ebp

mob ebp, esp

sub esp, 12

mov ebx, [ebp+8]

mov esi, [ebx]

mov ecx, 0

L1:

;k

mov [ebp-12], cx

mov ecx, 0

L2:

;i

mov [ebp-4], cx

mov ecx, 0

L3:

;j

mov [ebp-8], cx

mov ax, [ebp-4]

mov ebx, [ebp+12]

mul ebx

add ax, [ebp-8]

mov ebx, 64h

cmp [esi+4\*eax], ebx

je @not

;W[[ebp-4] \* count + [ebp-8]] = min(W[[ebp-4] \* count + [ebp-8]], W[[ebp-4] \* count + [ebp-12]] + W[[ebp-12] \* count + [ebp-8]]);

mov ebx, eax

xor eax, eax

mov ax, [ebp-4]

mov edx, [ebp+12]

mul edx

add ax, [ebp-12]

mov eax, [esi+4\*eax]

push eax

xor eax, eax

mov ax, [ebp-12]

mov edx, [ebp+12]

mul edx

add ax, [ebp-8]

mov eax, dword ptr [esi+4\*eax]

pop edx

add eax, edx

cmp dword ptr [esi+4\*ebx], eax

jle @endif

mov dword ptr [esi+4\*ebx], eax

jmp @endif

@not:

;W[[ebp-4] \* count + [ebp-8]] = W[[ebp-4] \* count + [ebp-12]] + W[[ebp-12] \* count + [ebp-8]]

mov ebx, eax

xor eax, eax

mov ax, [ebp-4]

mov edx, [ebp+12]

mul edx

add ax, [ebp-12]

mov eax, [esi+4\*eax]

push eax

xor eax, eax

mov ax, [ebp-12]

mov edx, [ebp+12]

mul edx

add ax, [ebp-8]

mov eax, dword ptr [esi+4\*eax]

pop edx

add eax, edx

mov dword ptr [esi+4\*ebx], eax

@endif:

inc ecx

cmp ecx, [ebp+12]

je @no1

jmp L3

@no1:

mov cx, [ebp-4]

inc ecx

cmp ecx, [ebp+12]

je @no2

jmp L2

@no2:

mov cx, [ebp-12]

inc ecx

cmp ecx, [ebp+12]

je @no3

jmp L1

@no3:

mov esp, ebp

pop ebp

ret

minDist ENDP

END

**Вывод**

В ходе выполнения данной лабораторной работы были получены навыки работы с языком программирования Assembler и особенности написания программ с его использованием.