|  |  |  |
| --- | --- | --- |
| Министерство науки и высшего образования  Российской Федерации | | |
| Федеральное государственное бюджетное  образовательное учреждение высшего образования | | |
| «Новосибирский государственный технический университет» | | |
|  | | |
|  | | |
|  | | |
| Лабораторная работа № 3 «Интерфейс с языком С++» | | |
| по дисциплине «Низкоуровневое программирование» | | |
|  | | |
| **кафедра теоретической и прикладной информатики** | | |
|  | | |
|  | Факультет: | ПМИ |
| Группа: | ПМИ-12 |
|  |  |
| Студенты: | Швадченко А. В.  Михайловский М. А. |
| Вариант: | 8 |
|  |  |
| Преподаватели: | Сивак Мария Алексеевна |
|  |  |
|  | | |
| Новосибирск, 2023 | | |

1. Цель работы

Изучить и приобрести практические навыки использования процедур на языке Ассемблера и их интерфейса с языком С++.

1. Ход работы
   1. Написать программу, находящую максимальный и минимальный символы в исходной строке. Для передачи данных процедуре на языке Ассемблера использовать аргументы функции. Для вызова процедуры использовать сdecl.
   2. Алгоритм:
      1. Ввод строки в консоль с помощью C++
      2. Вызов функции из ассемблера
      3. Помещение строки в буфер
      4. Проверка входных данных
      5. Посимвольное сравнение элементов, поиск максимума и минимума с помощью цепочечных команд
      6. Вывод максимума и минимума
   3. Текст разработанной программы

**Модуль assem.asm**

.386

.MODEL FLAT

.DATA

MAX DW 0

MIN DW 1000

LENS DD ?

BUF DB 200 dup (?)

.CODE

\_find PROC

PUSH EBP

MOV EBP, ESP

ADD EBP, 8

MOV ECX, [EBP]+4

XOR EDI, EDI

PUSH ECX

MOV ESI, OFFSET BUF

BUFER:

MOV EAX, [EBP] ; считываем символ

MOV AL, [EAX + EDI] ; получаем адрес текущего символа

MOV AH, 0

;заполнение буфера

CMP AX, 48 ;проверка входных данных

JB ERROR

CMP AX, 57

JA ERROR

PUSH AX

XOR EAX, EAX

POP AX

MOV [ESI], EAX

INC ESI

INC EDI

MOV EDX, [EBP] + 8

LOOP BUFER

;поиск максимума и минимума

CLD

POP ECX

LEA ESI, OFFSET BUF

PROG:

LODS BUF

CMP AX, MIN

JAE M

MINIM:

MOV MIN, AX

M:

CMP AX, MAX

JBE EXIT

MAXIM:

MOV MAX, AX

EXIT:

LOOP PROG

XOR EBX, EBX

MOV BX, MAX

MOV EAX, [EBP] + 8

MOV [EAX], EBX

XOR EBX, EBX

MOV BX, MIN

MOV EAX, [EBP] + 12

MOV [EAX], EBX

POP EBP

RET

;обработка при ошибке

ERROR:

POP ECX

POP EBP

MOV EAX, 1

RET

\_find ENDP

END

**Модуль main.cpp**

#include <conio.h>

#include <locale>

#include <iostream>

extern "C" int find(char\* source, int len, int\*max, int\*min);

void main()

{

char \*line = new char[200];

int out;

int max, min;

std::cout << "Enter the line: ";

std::cin >> line;

out = find(line, strlen(line), &max, &min);

if (out != 1)

{

printf("Maximim: ");

printf(" %d", max-48);

printf("\nMinimum: ");

printf(" %d", min-48);

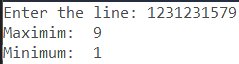
}

else

printf("Invalid character!");

}

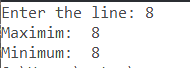
* 1. Тесты
     1. Стандартный тест



* + 1. Одинаковые элементы



* + 1. Один элемент



* + 1. Ввод неверного символа



* 1. Новые используемые средства
     1. CLD – очищение флага направления
     2. LEA – загрузка исполнительного адреса
     3. LODS – загрузка элемента цепочки в регистр
     4. RET – возврат из процедуры в место её вызова