Министерство науки и высшего образования Российской Федерации

федеральное государственное автономное   
образовательное учреждение высшего образования   
«Самарский национальный исследовательский университет   
имени академика С.П. Королева»

Институт информатики и кибернетики

Кафедра технической кибернетики

Отчет по лабораторной работе №2

Дисциплина: «Системное программирование и архитектура ЭВМ»

Выполнила: Иванова

Анна Алексеевна

Группа: 6206-010302D

№ варианта:4

Самара, 2024

**Задание на лабораторную работу**

1) В программе, разработанной в лабораторной работе 1 необходимо добавить

код на ассемблере реализующий:

1.1) Проверку делителя на равенство 0 — для устранения системной

ошибки деление на ноль.

1.2) Проверку делимого на переполнения частного — для устранения

системной ошибки «частное велико».

1.3) Проверку результата на переполнение.

2) В качестве комментария к каждой строке на ассемблере необходимо указать, какой промежуточный результат формируется в каком регистре.

Вариант 4: (6\*a - b + c/2)/(c + 3a/b)

**Текст программы**

#include<iostream>

#include<string>

using namespace std;

int main()

{

int a, b, c;

cout << "Input a: ";

cin >> a;

cout << "Input b: ";

cin >> b;

cout << "Input c: ";

cin >> c;

int res;

int zm = 0;

\_\_asm {

mov zm, 0

mov eax, a; кладем в eax значение a

mov ebx, b; кладем в ebx значение b

mov ecx, c; кладем в ecx значение c

imul eax, 3; 3 \* a

jo error\_of; ошибка переполнения

; готовимся к делению

cdq

idiv ebx; 3 \* a / b

add eax, ecx; < eax >= 3 \* a / b + c

jz error\_zf; ошибка деление на ноль

jo error\_of; ошибка переполнения

push eax; в стеке c + 3 \* a / b

mov eax, a

imul eax, 6; 6 \* a

jo error\_of; ошибка переполнения

sub eax, ebx; < eax >= 6 \* a - b

jo error\_of; ошибка переполнения

mov ebx, eax; < ebx >= 6 \* a - b

mov eax, c; eax = c

mov ecx, 2; ecx = 2

cdq

idiv ecx; eax = c / 2

add eax, ebx; < eax >= 6 \* a - b + c / 2

jo error\_of; ошибка переполнения

; готовимся к делению

pop ebx; < ebx >= c + 3 \* a / b

cdq

idiv ebx; < eax >= (6 \* a - b + c / 2) / (c + 3 \* a / b)

jmp m3

error\_zf :

mov zm, 1

jmp m3

error\_of :

mov zm, 2

jmp m3

m3 :

mov res, eax

}

if (zm == 0) cout << "Answer asm: " + to\_string(res);

if (zm == 1) cout << "division by zero";

if (zm == 2) cout << "owerflow";

return 0;

}







