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, математики и электроники

Факультет информатики

Кафедра технической кибернетики

Отчет по лабораторной работе №6

Дисциплина: «Системное программирование и архитектура ЭВМ»

Выполнил: Абрамов Д.О.

Группа: 6208-010302D

Самара, 2021

**Код программы**

#include <iostream>

#include <iostream>

using namespace std;

/\*

(b + 5)/a + 2, a>b

x = -b, a=b

(a-b)/3, a<b

\*/

int main()

{

int error, status;

double a, b, res, res\_asm;

cout << "Please, enter a:\n";

cin >> a;

cout << "Please, enter b:\n";

cin >> b;

if (a > b && a == 0) {

cout << "Do not try to divide by zero\n";

system("pause");

return 0;

}

if (a > b) {

res = (b + 5) / a + 2;

}

else if (a == b) {

res = -b;

}

else {

res = (a - b) / 3;

}

const double c2 = 2, c3 = 3, c5 = 5, c1 = -1;

\_\_asm

{

finit

fld qword ptr[b] //st(1) = b

fld qword ptr[a] //st = a

fcom st(1) //сравниваем a и b

fstsw status //сохраняем регистр флагов сопроцессора

mov ah, byte ptr[status + 1] //сохраняем второй байт

sahf

ja l\_bigger // переход если a > b

jb l\_smaller // переход если a < b

fld qword ptr[c1]

fmulp st(1), st

fstp res\_asm

jmp EndOfCode

l\_bigger :

fld c5 // st = 5

faddp st(2), st // st = b + 5

fldz //Загрузка 0

fcomp st(1) //Вещественное сравнение a c 0 с выталкиванием

fnstsw ax //Считать слово состояния сопроцессора в память

fwait

sahf //Записать содержимое AH в регистр FLAGS

jz error\_zf //Обработка ошибки "знаменатель равен 0"

fdivp st(1), st //st = (b + 5)/a

fld qword ptr[c2] // st = 2

faddp st(1), st // st = (b + 5)/a + 2

fstp res\_asm // (b + 5)/a + 2

jmp EndOfCode // переход на конец программы

l\_smaller :

fld qword ptr[a] //st(1) = b

fld qword ptr[b] //st = a

fsubp st(1), st // st = a - b

fld qword ptr[c3]

fdivp st(1), st // st = (a - b) / 3

fstp res\_asm // res\_asm = (a - b) / 3

jmp EndOfCode

error\_zf :

mov error, 1

jmp EndOfCode

EndOfCode:

}

if (error == 1) { //errror\_zf

cout << "Do not try to divide by zero\n";

system("pause»null");

return 0;

}

cout << "res c++ : " << res << endl;

cout << "res asm : " << res\_asm << endl;

return 0;

}

