



Министерство науки и высшего образования Российской Федерации
Калужский филиал
федерального государственного бюджетного
образовательного учреждения высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(КФ МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУК «Информатика и управление»

КАФЕДРА ИУК4 «Программное обеспечение ЭВМ, информационные технологии»

ЛАБОРАТОРНАЯ РАБОТА №9

**«Программирование математического сопроцессора.
Вызов ассемблерных функций из программ на C/C++»**

ДИСЦИПЛИНА: «Машинно-зависимые языки программирования»

Выполнил: студент гр. ИУК4-31Б _____ (Отрошенко Т. В.)
(Подпись)

Проверил: _____ (Амеличева К. А.)
(Подпись)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:
- Оценка:

Калуга, 2021

Цель работы: практическое овладение навыками разработки программного кода на языке Ассемблер. Изучить особенности программирования математического сопроцессора. Научиться стыковать модули, написанные с помощью языка Ассемблер и C++.

Задача: разработка программы, использующей технологию совместного программирования на языках C/C++ и Ассемблер. В программе использовать арифметические операции сопроцессора.

1. Написать на базовом алгоритмическом языке C++ программу корректного ввода исходных данных (с контролем допустимого диапазона), вычисления арифметического выражения и вывода результата.
2. Написать модуль вычисления на языке Ассемблера.
3. Встроить вызов `asm` – модуля в программу на базовом алгоритмическом языке.
4. Произвести тестовые проверки и сделать анализ результатов (найдите значение выражения, используя MS Excel)
5. Рассмотреть, как располагаются значения в стеке в процессе решения задачи.

Вариант 10

Задание 1

$$a = \frac{2}{\operatorname{tg} x^2} + \frac{2^{x \sin \sqrt{3x^2+1}}}{\sin x \cos x \log_2 \sqrt{x^2+1}}$$

Задание 2

$$f(x) = \begin{cases} \cos^4 x, & \text{если } x < 0 \\ 2^x - 7, & \text{если } 0 \leq x \leq 0,5 \\ (x^2 + 1)(x - 1), & \text{если } x > 0,5 \end{cases}$$

C++ модуль

```
#include <iostream>
#include <limits.h>
#include <conio.h>
#include <cmath>
#define pi = 3.14
using namespace std;
```

```

extern "C"

{
    float prog1(float);
    float prog2(float);
}

extern "C" float result;

float task1(float x)
{
    float b = 0.0;

    b = (2 / (tan(pow(x, 2))) + ((pow(2, (x * sin(sqrt(3 * pow(x, 2) + 1)))) / (sin(x) * cos(x) * log2(sqrt(pow(x, 2) + 1)))));

    return b;
}

float task2(float x)
{
    float f;

    if (x < 0)
    {
        f = pow(cos(x), 4);
    }

    if (x <= 0.5 and x >= 0)
    {
        f = pow(2, x) - 7;
    }

    if (x > 0.5)
    {
        f = (pow(x, 2) + 1) * (x - 1);
    }
}

```

```

        return f;
    }

int main()
{
    setlocale(LC_ALL, "rus");

    int selector;

    float x;

    do
    {
        cout << "1 - Первое задание" << endl;
        cout << "2 - Второе задание" << endl;
        cout << "0 - Выход" << endl;

        cin >> selector;

        switch (selector)
        {
            case 1:
                cout << "Введите x: ";
                cin >> x;

                cout << "Задание 1 на c++: " << task1(x) << endl;
                cout << "Задание 1 на assembler: " << prog1(x) << endl;
                break;

            case 2:
                cout << "Введите x: ";
                cin >> x;

                cout << "Задание 2 на c++: " << task2(x) << endl;
                cout << "Задание 2 на assembler: " << prog2(x) << endl;
                break;

            case 0:
                break;
        }
    }
}

```

```

        default:

            break;

        }

    } while (selector != 0);
}

```

Tasm модуль

```

.386

.model flat, C

.data

half      dd 0.5
one       dd 1.0
two       dd 2.0
three     dd 3.0
six       dd 6.0
perem     dd 0.0
num       dd 0.0
save1     dd 0.0
save2     dd 0.0

.code

Public C prog1
Public C prog2

prog1 proc C x: dword

    finit

    ; (2 / (pow(tan(x), 2))) + ((pow(2, (x * sin(sqrt(3 * pow(x, 2) +
1)))) / (sin(x) * cos(x) * log2(sqrt(pow(x, 2) + 1))))

    fld x                                ; ST(0)=x

```

```

fmul st(0), st(0)          ;ST(0)=x^2

fptan                      ;ST(0)=1, ST(1)=result

fld st(1)                  ;ST(1) to ST(0)

fld two                    ;ST(0)=two

fdiv st(0), st(1)          ;ST(0)=2/tg(x^2)

fstp perem                 ;save perem


fld x                      ;ST(0)=x

fmul st(0), st(0)          ;ST(0)=x^2

fld three                  ;ST(0)=3, ST(1)=x^2

fmul st(0), st(1)          ;ST(0)=3*x^2

fadd one                   ;ST(0)=3*x^2+1

fsqrt                     ;ST(0)=sqrt(3*x^2+1)

fsin                       ;ST(0)=sin(sqrt(3*x^2+1))

fld x                      ;ST(0)=x, ST(1)=sin(sqrt(3*x^2+1))

fmul st(0), st(1)          ;ST(0)=x*sin(sqrt(3*x^2+1))

f2xm1 ;2^x - 1            ;ST(0)=2^(x*sin(sqrt(3*x^2+1)))-1

fadd one                   ;ST(0)=2^(x*sin(sqrt(3*x^2+1)))

fstp num                   ;save num


fld x                      ;ST(0)=x

fmul st(0), st(0)          ;ST(0)=x^2

fadd one                   ;ST(0)=x^2+1

fsqrt                     ;ST(0)=sqrt(x^2+1)

fstp save1                 ;save in save1

fld x                      ;ST(0)=x

fcos                       ;ST(0)=cos(x)

fld save1                  ;ST(0)=save1(=sqrt(x^2+1))

fyl2x                      ;ST(0)=cos(x)*log(sqrt(x^2+1)) (y*log(x))

fld x                      ;ST(0)=x, ST(1)=cos(x)*log(sqrt(x^2+1))

```

```

    fsin                    ;ST(0)=sin(x), ST(1)=cos(x)*log(sqrt(x^2+1))

    fmul st(0), st(1)       ;ST(0)=sin(x)*cos(x)*log(sqrt(x^2+1))

    fld num                 ;ST(0)=num

    fdiv st(0), st(1)       ;ST(0)=num/sin(x)*cos(x)*log(sqrt(x^2+1))

    fadd perem
;ST(0)=2^(x*sin(sqrt(3*x^2+1)))/sin(x)*cos(x)*log(sqrt(x^2+1))+2/tg(x^2)

    fwait

    ret

prog1 endp

prog2 proc C x: dword

    finit

    fld x                   ;ST(0)=x

    ftst                   ;compare ST(0) and 0

    FSTSW AX

    SAHF

    jnb case_1

    fld half

    fld x                   ;<0

    fcomp st(1)             ;compare

    FSTSW AX

    SAHF

    ja case_3               ;>0,5

    jmp case_2              ;else

case_1:

    fld x                   ;ST(0)=x

    fcos                   ;ST(0)=cos(x)

    fmul st(0), st(0)       ;ST(0)=(cos(x))^2

    fmul st(0), st(0)       ;ST(0)=(cos(x))^4

```

```

    jmp fin

case_2:
    fld x                ;ST(0)=x
    f2xm1                ;ST(0)=2^x-1
    fld six              ;ST(0)=6
    fld st(1)            ;ST(1)=6
    fsub st(0), st(1)    ;ST(0)=2^x-7
    jmp fin

case_3:
    fld x                ;ST(0)=x
    fmul st(0), st(0)    ;ST(0)=x^2
    fadd one             ;ST(0)=x^2+1
    fstp save2          ;save in save2
    fld one              ;ST(0)=1
    fld x                ;ST(0)=x, ST(1)=1
    fsub st(0), st(1)    ;ST(0)=x-1
    fld save2            ;ST(0)=x^2+1
    fmul st(0), st(1)    ;ST(0)=(x^2+1)(x-1)
    jmp fin

fin:
    ret
    fwait

prog2 endp
end

```


Результаты выполнения:

Выполнение первой задачи:

```
1 - Первое задание
2 - Второе задание
0 - Выход
1
Введите x: 1
Задание 1 на c++: 9.54608
Задание 1 на assembler: 9.54608
```

		x	1
задание 1			tan
		промежуточные вычисления	1,28419
		b	9,54608

Выполнение второй задачи:

```
Введите x: -1
Задание 2 на c++: 0.0852211
Задание 2 на assembler: 0.0852211
```

```
Введите x: 0.1
Задание 2 на c++: -5.92823
Задание 2 на assembler: -5.92823
```

```
Введите x: 10
Задание 2 на c++: 909
Задание 2 на assembler: 909
```

задание 2	x	условие	f
	-1	$x < 0$	0,08522
	0,1	else	-5,92823
	10	$x > 0,5$	909

Вывод: в ходе выполнения лабораторной работы были приобретены навыки написания программ с циклами на языке Ассемблер.