

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МОЭВМ

ОТЧЕТ
по лабораторной работе №8
по дисциплине «Организация ЭВМ и систем»
ТЕМА: Обработка вещественных чисел. Программирование
математического сопроцессора
Вариант 3

Студент гр. 1303

Беззубов Д.В.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

Цель работы.

Составить программу для вычисления значения гиперболического синуса на языке ассемблера с использованием математического сопроцессора.

Задание.

Разработать подпрограмму на языке Ассемблера, обеспечивающую вычисление заданной математической функции с использованием математического сопроцессора. Подпрограмма должна вызываться из головной программы, разработанной на языке С. При этом должны быть обеспечены заданный способ вызова и обмен параметрами. Альтернативный вариант реализации: разработать на языке Ассемблера фрагмент программы, обеспечивающий вычисление заданной математической функции с использованием математического сопроцессора, который включается по принципу in-line в программу, разработанную на языке С.

Вариант 3:

* function

Name sinh - hyperbolic sine function

Usage double sinh(double x);

Prototype in math.h

Description sinh computes the hyperbolic sine of the input value.

$$\sinh(x) = (\exp(x) - \exp(-x)) / 2$$

Выполнение работы.

Выполнение программы начинается с ввода x , данное значение будет в качестве аргумента гиперболического синуса. В начале ассемблерной вставки сравниваем x с 0, в случае, если он отрицательный, получаем абсолютное значение переменной, в переменную *flag* записываем -1. Абсолютное значение будет расположено в регистре *st(0)* стека сопроцессора.

Для того, чтобы вычислить экспоненту, воспользуемся следующим преобразованием:

$$e^x = 2^{x \cdot \log_2 e}$$

Для этого последовательно вызываем команды FYL2X и F2XM1. Вычисляем требуемое значение, используя команды fdiv и fadd, записываем результат с вершины стека с помощью команды FSTP.

Вычисляем значение функции по формуле:

$$\sinh(x) = \left(e^x - \frac{1}{e^x} \right) / 2$$

В конце умножаем полученное значение на значение переменной *flag* и выводим его на экран. Так же выводим разницу между вычисленным значением со значением, полученным с помощью стандартной функции *sinh(x)*.

Исходный код программы см. в приложении А.

Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	0.1	Введите значение x: 0.1 Вычисленное значение $\sinh(x)$: 0.100167 Абсолютная погрешность вычисления: 0.00000000000000006939	
2.	-0.5	Введите значение x: -0.5 Вычисленное значение $\sinh(x)$: -0.521095 Абсолютная погрешность вычисления: 0.00000000000000000000	
3.	0.5	Введите значение x: 0.5 Вычисленное значение $\sinh(x)$: 0.521095 Абсолютная погрешность вычисления: 0.00000000000000000000	

Таблица фиксации изменения стека и регистров при $x = 0.1$

Команда	До выполнения:	После выполнения:
fld qword ptr[x]	ST0 = +0.0000000000000000e+0000	ST0 = +1.0000000000000000e-0001
fldZ	ST0 = +1.0000000000000000e-0001 ST1 = +0.0000000000000000e+0000	ST0 = +0.0000000000000000e+0000 ST1 = +1.0000000000000000e-0001
FCOMP	ST0 = +0.0000000000000000e+0000 ST1 = +1.0000000000000000e-0001	ST0 = +1.0000000000000000e-0001 ST1 = +0.0000000000000000e+0000
jge cont		
fld qword ptr[e]	ST0 = +1.0000000000000000e-0001 ST1 = +0.0000000000000000e+0000	ST0 = +2.7182818284590450e+0000 ST1 = +1.0000000000000000e-0001
FYL2X	ST0 = +2.7182818284590450e+0000 ST1 = +1.0000000000000000e-0001	ST0 = +1.4426950408889634e-0001 ST1 = +0.0000000000000000e+0000
F2XM1	ST0 = +1.4426950408889634e-0001	ST0 = +1.0517091807564762e-0001
fld1	ST0 = +1.0517091807564762e-0001 ST1 = +0.0000000000000000e+0000	ST0 = +1.0000000000000000e+0000 ST1 = +1.0517091807564762e-0001
fadd st, st(1)	ST0 = +1.0000000000000000e+0000	ST0 = +1.1051709180756477e+0000
fld st	ST0 = +1.1051709180756477e+0000 ST1 = +1.0517091807564762e-0001 ST2 = +0.0000000000000000e+0000	ST0 = +1.1051709180756477e+0000 ST1 = +1.1051709180756477e+0000 ST2 = +1.0517091807564762e-0001
fld1	ST0 = +1.1051709180756477e+0000 ST1 = +1.1051709180756477e+0000 ST2 = +1.0517091807564762e-0001 ST3 = +0.0000000000000000e+0000	ST0 = +1.0000000000000000e+0000 ST1 = +1.1051709180756477e+0000 ST2 = +1.1051709180756477e+0000 ST3 = +1.0517091807564762e-0001
fdivrp st(1), st	ST0 = +1.0000000000000000e+0000 ST1 = +1.1051709180756477e+0000 ST2 = +1.1051709180756477e+0000 ST3 = +1.0517091807564762e-0001	ST0 = +9.0483741803595951e-0001 ST1 = +1.1051709180756477e+0000 ST2 = +1.0517091807564762e-0001 ST3 = +0.0000000000000000e+0000

fsubp st(1), st	ST0 = +9.0483741803595951e-0001 ST1 = +1.1051709180756477e+0000 ST2 = +1.0517091807564762e-0001	ST0 = +2.0033350003968819e-0001 ST1 = +1.0517091807564762e-0001 ST2 = +0.0000000000000000e+0000
fdiv qword ptr[two]	ST0 = +2.0033350003968819e-0001	ST0 = +1.0016675001984409e-0001
fmul qword ptr[flag]		
FSTP qword ptr[res]	ST0 = +1.0016675001984409e-0001 ST1 = +1.0517091807564762e-0001	ST0 = +1.0517091807564762e-0001 ST1 = +0.0000000000000000e+0000

Выводы.

В ходе выполнения лабораторной работы были получены навыки программирования математического сопроцессора. Разработана программа позволяющая получить значение гиперболического синуса от введенного числа.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: source.asm

```
#include <math.h>
#include <iostream>

long double x;
long double e = exp(1);
long double const two = 2;
long double neg = -1;
long double flag = 1;
long double res;

int main() {

    system("chcp 1251 > nul");
    setlocale(LC_CTYPE, "rus");

    std::cout << "Введите значение x: \n";
    std::cin >> x;

    __asm {

        fld qword ptr[x]
        fldz
        FCOMP
        jge cont
        fabs
        fld qword ptr[neg]
        fstp qword ptr[flag]

        cont:
        fld qword ptr[e]
        FYL2X
        F2XM1
        fld1
        fadd st, st(1)
        fld st
        fld1
        fdivrp st(1), st
        fsubp st(1), st
        fdiv qword ptr[two]
        fmul qword ptr[flag]
        FSTP qword ptr[res]
    }
    printf("Вычисленное значение sinh(x): %lf\n", res);
    printf("Абсолютная погрешность вычисления: %.20lf\n", abs(res -
sinh(x)));
    return 0;
}
```