**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**

**ІКНІ**

Кафедра **ПЗ**

### ЗВІТ

до лабораторної роботи № 6

**з дисципліни:** *“Архітектура комп’ютера”*

**на тему:***“Програмування арифметичного співпроцесора мікропроцесорів х86”*

**Лектор:**

доц. каф. ПЗ

Крук О.Г.

**Виконав:**

ст. гр. ПЗ-22

Солтисюк Д.А.

**Прийняв:**

доц. каф. ПЗ

Крук О.Г.

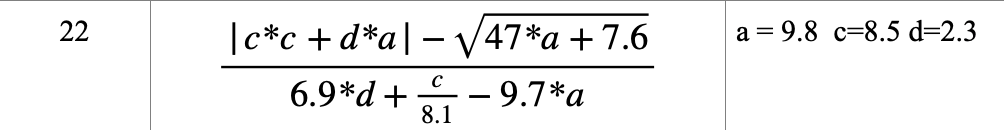
« \_\_\_\_ » \_\_\_\_\_\_\_\_ 2022 р.

∑= \_\_\_\_\_ .

Львів – 2022

**Тема роботи:** Програмування арифметичного співпроцесора мікро- процесорів х86.

**Мета роботи:** розвинути навики складання програми для арифметичного співпроцесора мовою асемблера для обчислення математичного виразу, відтранслювати і виконати в режимі відлагодження програму, складену відповідно до свого варіанту, обчислити заданий вираз в програмі мовою С та порівняти результати.

**Варіант:** 22

**Теоретичні відомості**

Арифметичний процесор або співпроцесор - це цифровий пристрій, призначений для апаратного виконання арифметичних операцій над дійсними (з плаваючою комою) числами. Наявність співпроцесора дозволяє значно прискорити роботу програм, що виконують обчислення з високою точністю, тригонометричні розрахунки та опрацювання інформації, яка повинна бути подана у вигляді дійсних чисел. В перших моделях мікропроцесорів Intel співпроцесора не було, він виготовлявся у вигляді окремої інтегральної мікросхеми і входив в склад комп’ютерів як опція. Починаючи з моделі i486DX співпроцесор розміщується на тому ж кристалі, що і основний процесор. .

Співпроцесор має вісім 80-розрядних регістрів R0 - R7 для зберігання чисел з плаваючою комою, організованих у вигляді кільцевого стека. Номер регістра, який на даний момент перебуває на вершині стека, вказується в 3-бітовому полі ТОР, що міститься в слові стану. При написанні програм, в яких використовуються команди з плаваючою комою, до вершини стека можна звернутися за допомогою операнда ST(0) (або просто ST). В командах можна також використовувати відносні до вершини стека операнди SТ(1) ... ST(7). Абсолютні імена регістрів типу R0, R1, ... R7 використовувати не можна.

При виконанні команд з плаваючою комою їх операнди зберігаються в десятибайтових регістрах у розширеному форматі з подвійною точністю. При збереженні результату арифметичної операції в пам'яті співпроцесор автоматично перетворює його з розширеного формату в ціле або довге ціле число, а також в коротке або довге дійсне число.

Основний процесор і співпроцесор можуть обмінюватися значеннями з плаваючою комою тільки через оперативну пам'ять. Тому перед викликом команди співпроцесора її операнд завжди повинен міститися в пам'яті. При цьому співпроцесор завантажує число з пам'яті в свій стек регістрів і виконує над ним арифметичну операцію.

Мнемоніки команд з плаваючою комою завжди починаються з літери F/f, щоб їх можна було відрізнити від інших команд основного процесора. Друга літера в мнемоніці (зазвичай це B/b або I/i) визначає спосіб інтерпретації операнда, що міститься в пам'яті. Літера В свідчить про те, що оператор поданий в двійково-десятковому коді (Binary-Coded Decimal, або BCD). Літера І говорить про те, що оператор поданий у вигляді цілого значення. Якщо ці літери не вказані, то передбачається, що оператор міститься в пам'яті в одному з форматів чисел із плаваючою комою. До прикладу, команда FBLD оперує з двійково-десятковими числами (BCD-числами), команда FILD - з цілими числами, а FLD - з дійсними, поданими в форматі з плаваючою комою.

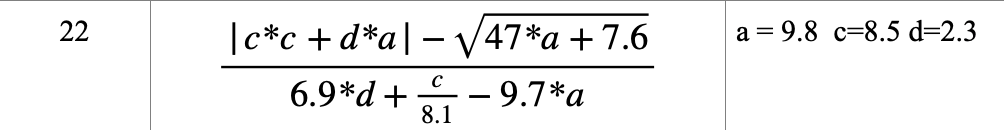
У командах з плаваючою комою можна вказати максимум два оператори, причому один з них - це ім'я одного з регістрів з плаваючою комою. Безпосередньо задані операнди не використовуються. Як операнди не можна також використовувати імена регістрів загального призначення основного процесора, таких як АХ або ЕВХ. Не дозволені також операції типу "пам’ять-пам’ять".

**Індивідуальне завдання**

1. Складіть програму обчислення виразу за допомогою команд співпроцесора для WINDOWS.

2. Перевірте результат роботи асемблерної програми, порівнявши його з результатом програми мовою Сі.

3. У звіті наведіть текст програми, копії вікон з результатами.

4. Зробіть висновки про виконану роботу.

**Хід виконання**

1. Склав програму обчислення виразу за допомогою команд співпроцесора.
2. Склав програму обчислення виразу на мові Сі.
3. Перевірив результат роботи асемблерної програми порівнявши його з результатом програми мовою Сі. .

pasted-image.png

*Результат роботи програми на мові Асемблера*

*pasted-image.png*

*Результат роботи програми на мові С*

**Код програми на мові Асемблера**

*; vim: ft=masm*

.686

.model flat,stdcall

.stack

.data

A REAL4 9.8

B REAL4 8.5*; C variable*

D REAL4 2.3

C1 REAL4 47.0

C2 REAL4 7.6

C3 REAL4 6.9

C4 REAL4 8.1

C5 REAL4 9.7

TOP REAL4 ?

BOT REAL4 ?

RES REAL4 ?

.code

main:

**finit**

**fld** D

**fmul** A *; st0 = D \* A*

**fld** B

**fmul** B *; st0 = C \* C, st1 = D \* A*

**fadd**

**fabs** *; st0 = abs(C \* C + D \* A)*

**fld** C1 *; st0 = C1, st1 = abs(C \* C + D \* A)*

**fmul** A *; st0 = C1 \* A, st1 = abs(C \* C + D \* A)*

**fadd** C2 *; st0 = C1 \* A + C2, st1 = abs(C \* C + D \* A)*

**fsqrt** *; st0 = sqrt(C1 \* A + C2), st1 = abs(C \* C + D \* A)*

**fsubp**

**fst** TOP

**fld** D

**fmul** C3

**fld** B

**fdiv** C4

**fld** C5

**fmul** A

**fsubp**

**faddp**

**fst** BOT

**fld** TOP

**fdiv** BOT

**fst** RES

**RET**

**END** main

**Код програми на мові C**

#include <math.h>

#include <stdio.h>

int main() {

double a = 9.8, c = 8.5, d = 2.3;

double res = (fabs(c \* c + d \* a) - sqrt(47 \* a + 7.6)) /

(6.9 \* d + c / 8.1 - 9.7 \* a);

printf("The result is: %f\n\n\n", res);

**return** 0;

}

**Висновки**

На цій лабораторній роботі я розвинув навики складання програми для арифметичного співпроцесора мовою асемблера для обчислення математичного виразу, відтранслював і виконав в режимі відлагодження програму, складену відповідно до свого варіанту, відлагодив та перевірив виконання тесту.