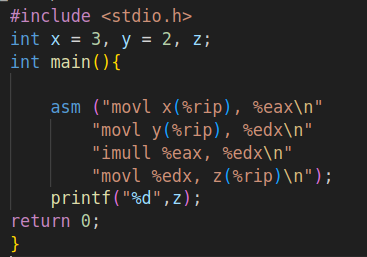
Лабораторная работа №1

Игнатьев П. А.

Б03-205

1. Программа с ассемблерной вставкой



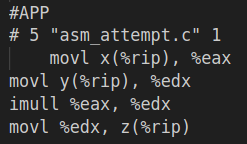
В ассемблерной вставке выполняется:

1) перенос значений переменных в нужные регистры

2) Происходит умножение двух регистров

3) Присвоение результата умножения переменной z

В ассемблерном листинге код выглядит следующим образом:



#APP – (imho) означает ассемблерную вставку из самого кода

Далее после этого идёт #NO\_APP и остальной код, не написанный “программистом”.

ASSembler’ные метки и переход к ним (наиболее встречаемые, если ты не “жёсткий прогер”):

Для начала, что такое **cmp**?

cmp – сравнение (О0О)

**jmp** – переходит на метку вне зависимости от условий

**je** - Jump if Equal

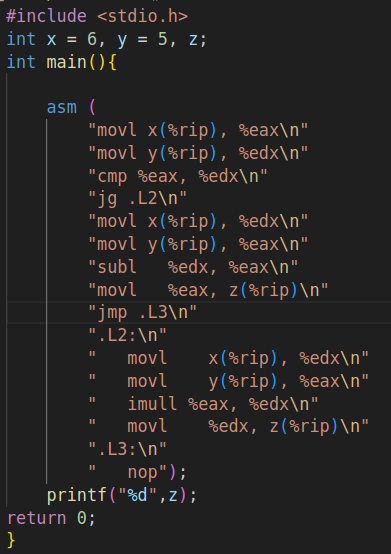
**jne** - Jump if Not Equal

**jg** - Jump if Greater or Equal

**jz** - Jump if Zero

**jnz** - Jump if Not Zero

Ассемблерная вставка с метками:



РАБОТАЕТ!!!

Если x < y: y – x;

иначе : y \* x;

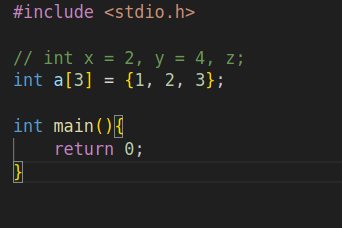
cmp – проверка

потом если %eax > %edx переходим на .L2

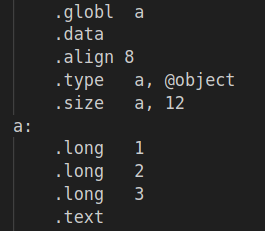
если нет, то идём дальше пока не встречаем .L3

в .L3 nop – значит нет оператора, т.е. за этим последует завершение программы

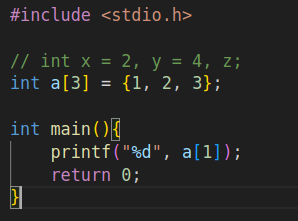
КРЧ массивы:



В ассемблерном листинге (очевидно там всё):



Вывод элемента массива:



В листинге:



Означает начало a (a.begin()) плюс 4 байта == (a.begin() + 4) or (a[1])

Вроде легко.