

## Assignment 1

Link to github: [https://github.com/davidirvinne/David\\_Irvinne-SKJ-Lab/](https://github.com/davidirvinne/David_Irvinne-SKJ-Lab/)

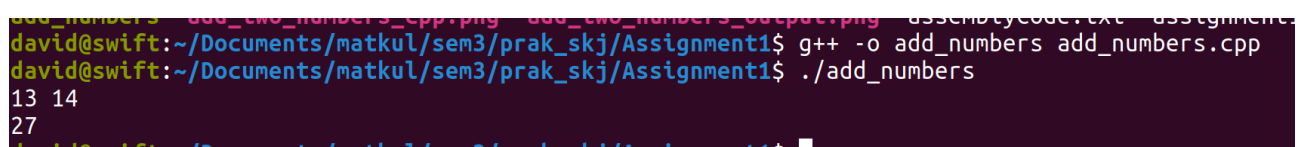
### First Task: C++ Code to Assembly

1. Write a simple C++ program that adds two integers



```
1  #include<iostream>
2  using namespace std;
3
4  int main(){
5      int a, b;
6      cin >> a >> b;
7      cout << a + b << '\n';
8      return 0;
9  }
```

2. Compile the code using g++



```
david@swift:~/Documents/matkul/sem3/prak_skj/Assignment1$ g++ -o add_numbers add_numbers.cpp
david@swift:~/Documents/matkul/sem3/prak_skj/Assignment1$ ./add_numbers
13 14
27
```

3. Disassemble the code

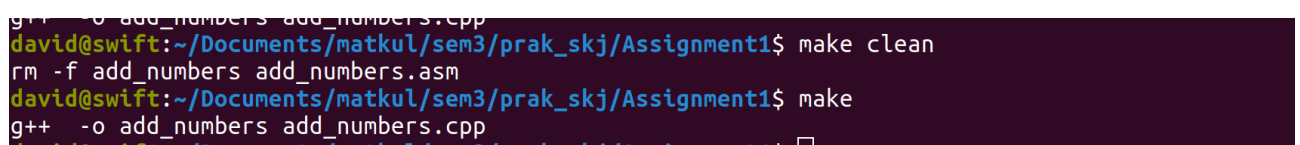
Terminal command:



```
david@swift:~/Documents/matkul/sem3/prak_skj/Assignment1$ subl assemblyCode.txt
david@swift:~/Documents/matkul/sem3/prak_skj/Assignment1$ objdump -d add_numbers > assemblyCode.txt
```

The code is available in the github repository, named *assemblyCode.txt*.

4. Write a Makefile



```
g++ -o add_numbers add_numbers.cpp
david@swift:~/Documents/matkul/sem3/prak_skj/Assignment1$ make clean
rm -f add_numbers add_numbers.asm
david@swift:~/Documents/matkul/sem3/prak_skj/Assignment1$ make
g++ -o add_numbers add_numbers.cpp
```

The code is available in the github repository, named *Makefile*.

## Second Task: Assembly to C++

### 1. Analyze the provided Assembly code

```
1  section .data
2      num1 dw 5
3      num2 dw 10
4      result dw 0
5
6  section .text
7      global _start
8
9  _start:
10     mov ax, [num1]
11     imul ax, [num2]
12     mov [result], ax
13
14     ; Exit the program
15     mov eax, 1
16     xor ebx, ebx
17     int 0x80
```

Penjelasan:

Line 1:

Menunjukkan tempat variabel-variabel disimpan. Bagian ini digunakan untuk mendeklarasikan data statis yang digunakan dalam program.

Line 2 – 4:

*num1 dw 5* mendeklarasikan variabel *num1* dengan tipe data *dw* (define word, 2 byte) dan menginisialisasinya dengan nilai 5.

*num2 dw 10* mendeklarasikan variabel *num2* dengan tipe data *dw* (define word, 2 byte) dan menginisialisasinya dengan nilai 10.

*result dw 0* mendeklarasikan variabel *result* dengan tipe data *dw* (define word, 2 byte) dan menginisialisasinya dengan nilai 0.

Line 6 – 7:

*section .text* mendefinisikan segmen teks tempat kode eksekusi program berada.

*global \_start* menandakan bahwa label *\_start* adalah titik masuk program, dan akan digunakan oleh linker sebagai titik mulai eksekusi.

Line 9 – 12:

*\_start* sebagai label titik mulai eksekusi program

*mov ax, [num1]* memindahkan nilai dari *num1* (yaitu 5) ke register *ax*

*imul ax, [num2]* melakukan perkalian nilai di *ax* dengan nilai dari *num2*. Hasilnya disimpan di *ax*.

*mov [result], ax* menyimpan nilai dari register *ax* ke variabel *result*.

Line 14 – 17:

; *Exit the program* komentar yang menjelaskan bahwa bagian ini digunakan untuk keluar dari program.

*mov eax, 1* mengatur register *eax* ke 1, yang merupakan nomor panggilan sistem (*system call number*) untuk keluar dari program pada sistem operasi Unix/Linux.

*xor ebx, ebx* mengatur register *ebx* ke 0 dengan melakukan operasi XOR ke diri nya sendiri. Ini menunjukkan status keluar (*exit status*) dari program.

*int 0x80* menghasilkan *interrupt 0x80* yang digunakan untuk memanggil layanan kernel Linux, yaitu keluar dari program.

Kesimpulannya, kode *assembly* diatas melakukan operasi perkalian dua bilangan, yaitu 5 dan 10 lalu menyimpan hasilnya ke variabel *result*.

2. Write the equivalent C++ code

```
1  #include<iostream>
2  using namespace std;
3
4  int main(){
5      int num1 = 5, num2 = 10;
6      int result = 5 * 10;
7      return 0;
8  }
```

3. Write a Makefile

```
david@swift:~/Documents/matkul/sem3/prak_skj/Assignment1/second_task$ make clean
rm -f mul_numbers mul_numbers.asm
david@swift:~/Documents/matkul/sem3/prak_skj/Assignment1/second_task$ make
g++ -o mul_numbers mul_numbers.cpp
david@swift:~/Documents/matkul/sem3/prak_skj/Assignment1/second_task$ ls
a.asm Makefile mul_numbers mul_numbers.cpp mul.png second_task_code.png
david@swift:~/Documents/matkul/sem3/prak_skj/Assignment1/second_task$
```

The code is available in the github repository, named *Makefile*.