Week2 Report in Class (Fri56)

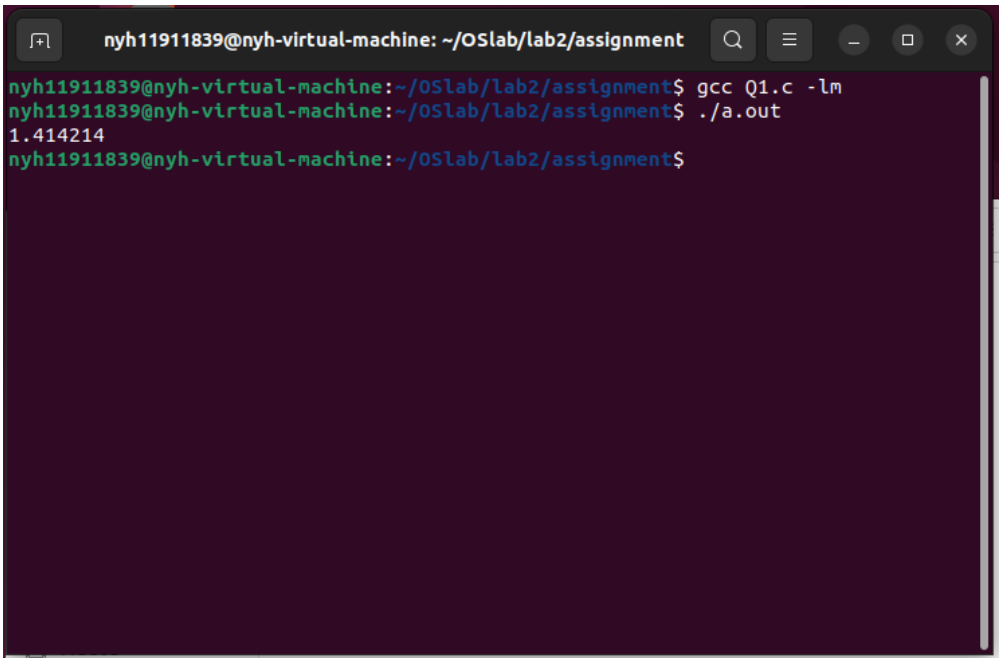
> 11911839 聂雨荷

Q1

> 编写c代码存在Q1.c中(截图)，代码中使用math库中的sqrt函数，对2进行开方，并打印结果。用gcc命令生成可执行文件Q1并执行(截图)。

```c
#include <math.h>
#include <stdio.h>


int main(){
    float num = 2;
    float num_sqrt = sqrt(num);
    printf("%f\n", num_sqrt);
    return 0;
}
```



Q2

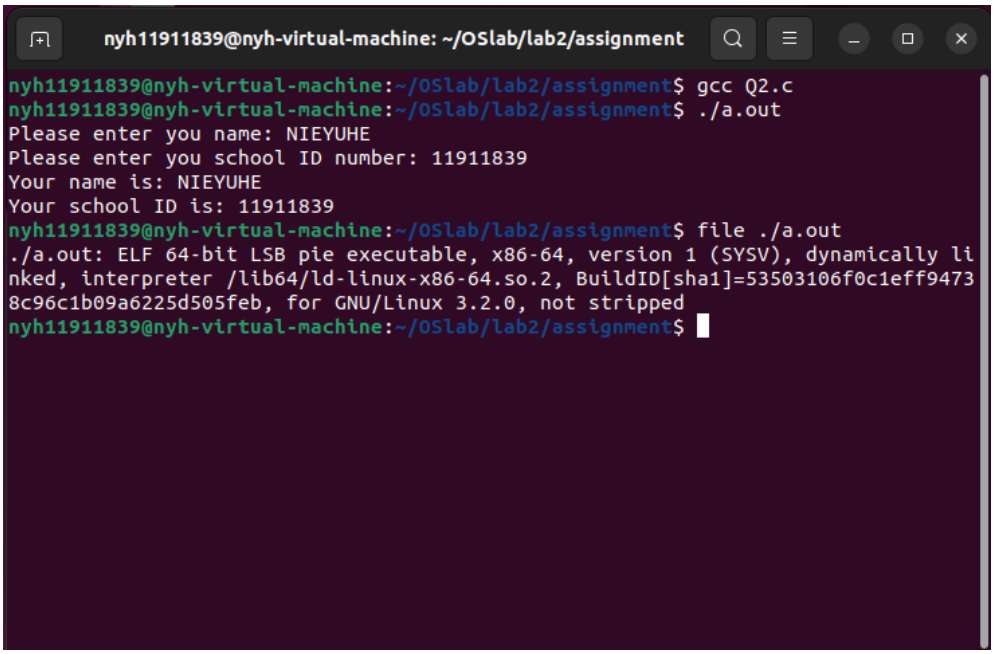
> 编写c代码存在Q2.c中,请实现一个可以打印自己姓名和学号的C语言程序(截图)。用gcc生成目标文件Q2.o，然后生成可执行文件Q2并执行。用file命令查看文件类型(截图)。

```c
#include <stdio.h>

int main(){
    char name_str[20];
    char id_str[20];

    printf("Please enter you name: ");
    scanf("%s", name_str);
    printf("Please enter you school ID number: ");
    scanf("%s", id_str);

    printf("Your name is: %s\n", name_str);
    printf("Your school ID is: %s\n", id_str);
    return 0;
}
```



Q3

> 请总结C语言的编译过程。

1. Pre-process
    - extend `#define` `#include` into C code
2. Compile & Optimize

    - Check the grammar of the code, if there is no mistake, compile it and transform it into intermediate code (assembly language)
    - Optimize the code
    - Output `.s` files
3. Assemble
    - Transform the intermediate code into the target code (machine language)
    - Output `.o` files
4. Link
    - Link all the target code and the static/dynamic library
    - Produce executable file

Q4

> windows和ubuntu下的可执行文件的格式分别是什么?

- Windows: `.exe`
- Ubuntu: There is not suffix of the executable file in Ubuntu, while the The standard Linux executable format is named **Executable and Linking Format (ELF)**

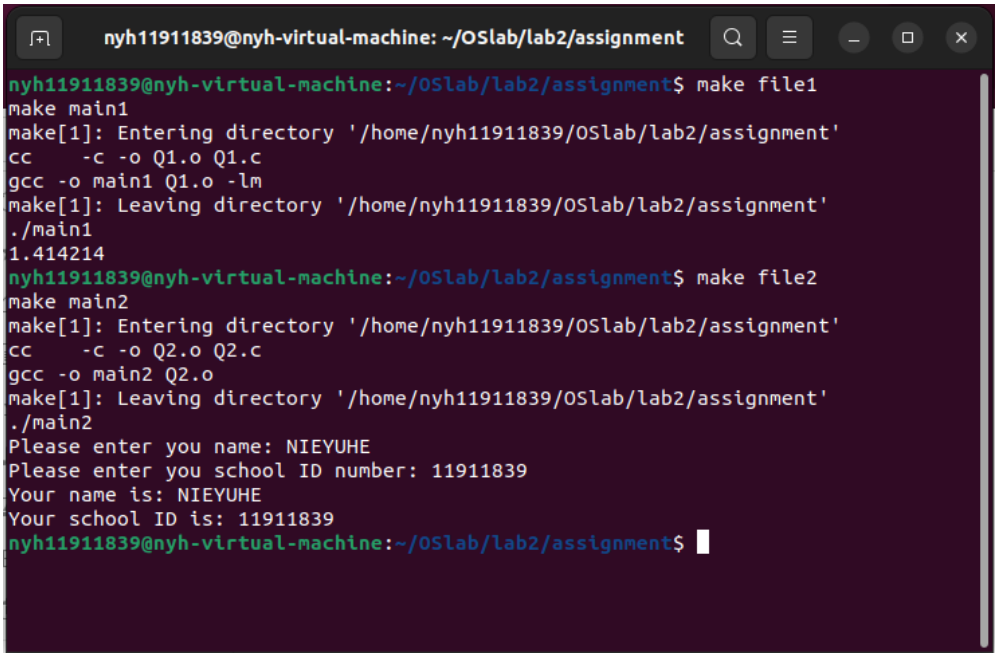
Q5

> 请编写makefile文件(截图)，执行make命令或者执行make file1时通过Q1打印2的开方结果(截图)，执行make file2时通过Q2打印自己的姓名学号(截图)。

```
main1: Q1.o
    gcc -o main1 Q1.o -lm
Q1.o: Q1.c
file1:
    make main1
    ./main1

main2: Q2.o
    gcc -o main2 Q2.o
Q2.o: Q2.c
file2:
    make main2
    ./main2

clean:
    rm Q1.o Q2.o
```

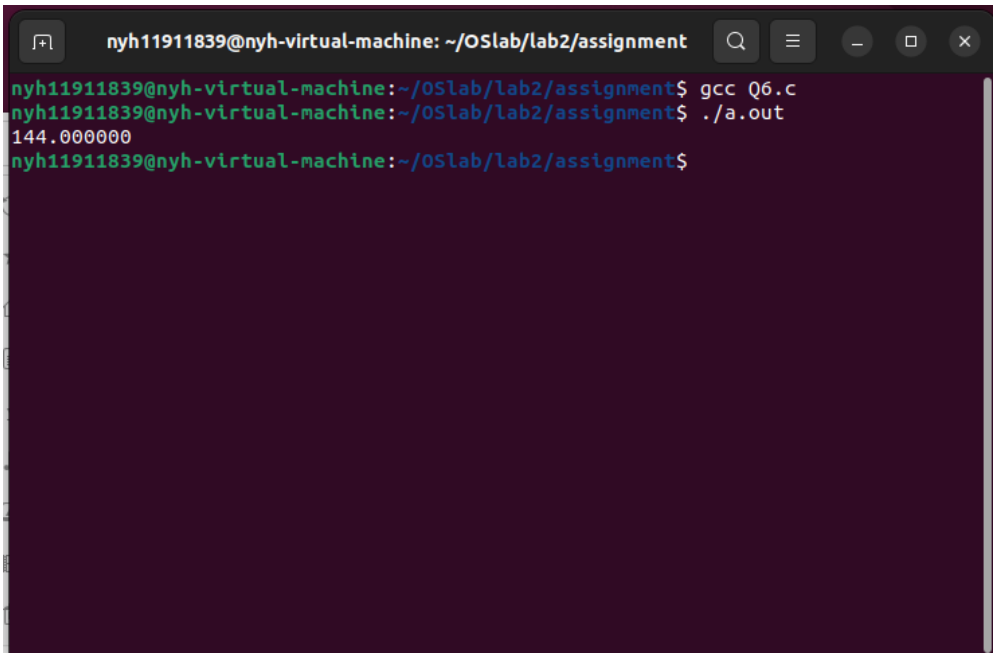


Q6

> 给定宏定义 #define MUL(x) (x)*(x) , MUL(9+3)的计算结果是什么?

```c
#define MUL(x) (x)*(x)
#include <stdio.h>

int main(){
    float result = MUL(9+3);
    printf("%f\n", result);
    return 0;
}
```



The result is 144.

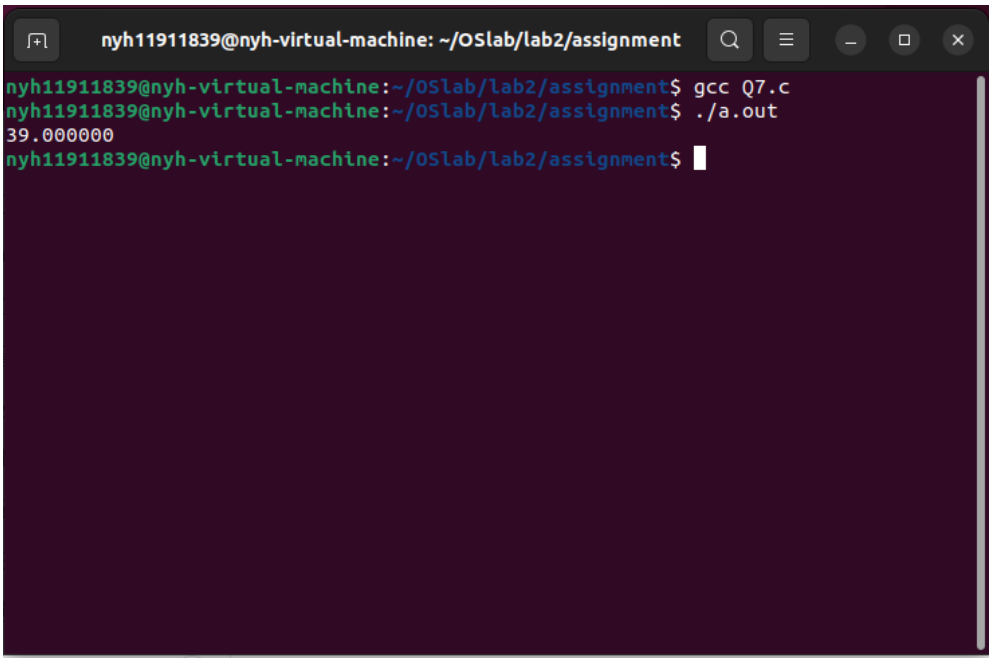
- (9+3) * (9+3) = 144

Q7

> 给定宏定义 #define MUL(x) x*x , MUL(9+3)的计算结果是什么?

```c
#define MUL(x) x*x
#include <stdio.h>

int main(){
    float result = MUL(9+3);
    printf("%f\n", result);
    return 0;
}
```



The result is 39.

- 9 + 3*9 + 3 =9+27+3=39