Daniyal Khan

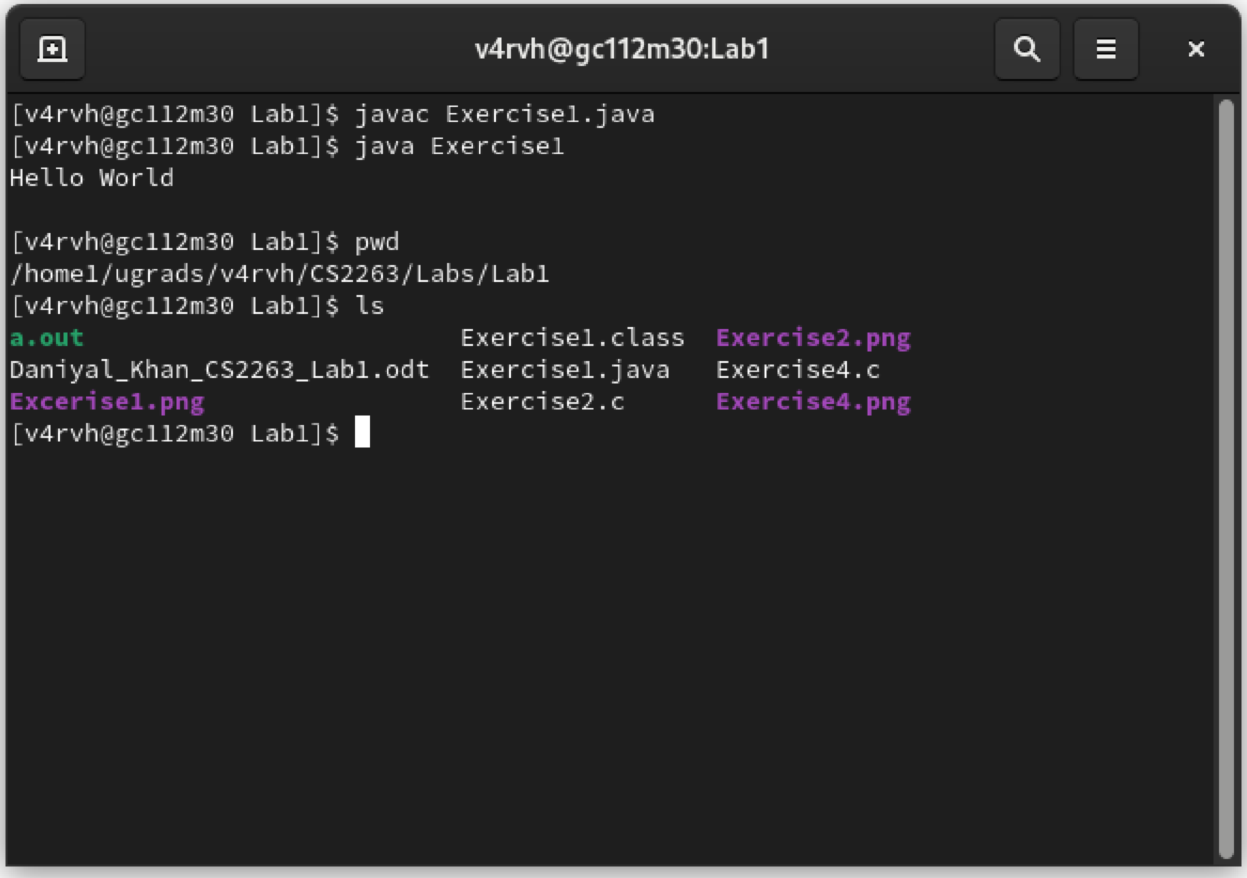
3765942

CS2263

Lab 1

Exercise 1:

```
class Exercise1 {
    public static void main (String[] args) {
        System.out.println("Hello World\n");
    }
}
```



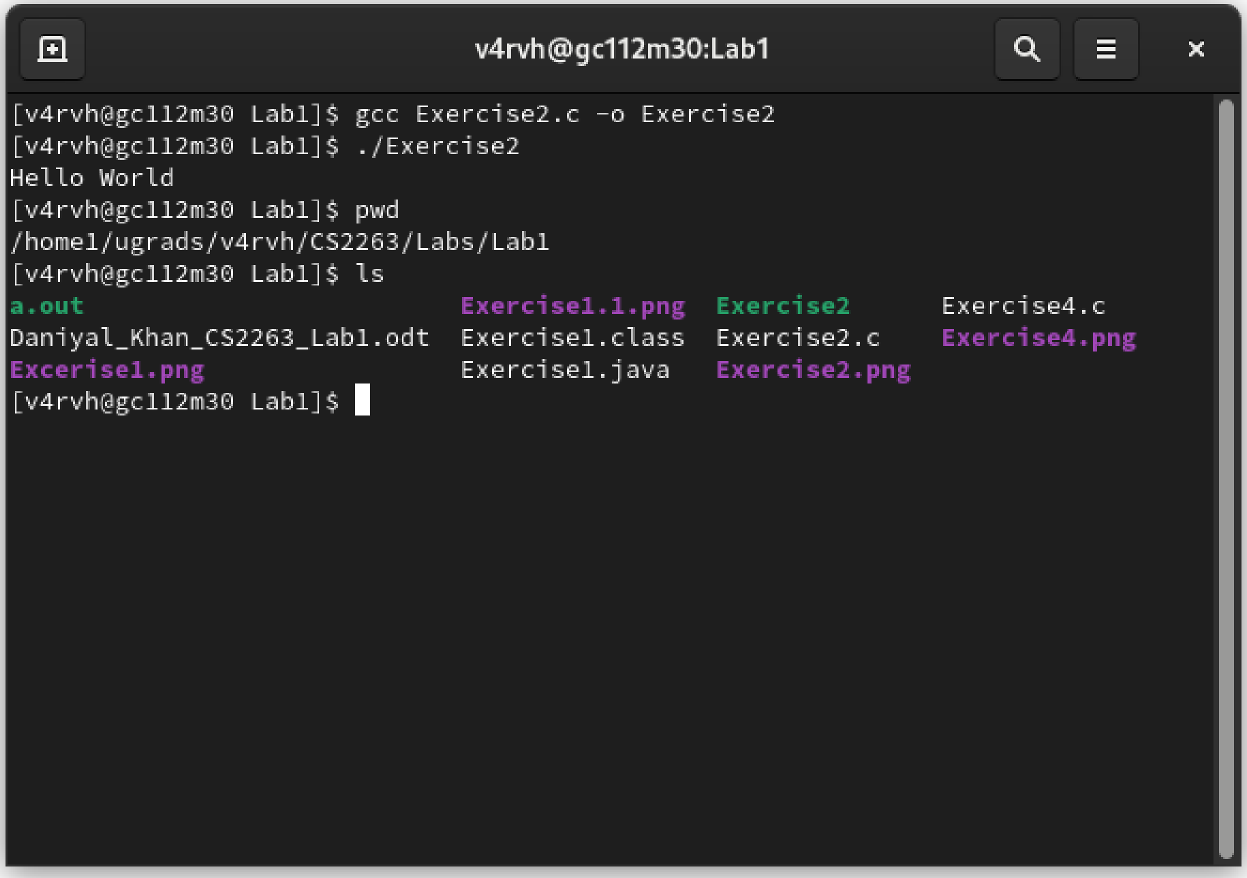
```
[v4rvh@gc112m30 Lab1]$ javac Exercise1.java
[v4rvh@gc112m30 Lab1]$ java Exercise1
Hello World

[v4rvh@gc112m30 Lab1]$ pwd
/home1/ugrads/v4rvh/CS2263/Labs/Lab1
[v4rvh@gc112m30 Lab1]$ ls
a.out                         Exercise1.class  Exercise2.png
Daniyal_Khan_CS2263_Lab1.odt  Exercise1.java   Exercise4.c
Excerise1.png                 Exercise2.c      Exercise4.png
[v4rvh@gc112m30 Lab1]$
```

Exercise 2:

```
#include <stdio.h>

int main() {
printf("Hello World\n");
return 0;
}
```



```
[v4rvh@gc112m30 Lab1]$ gcc Exercise2.c -o Exercise2
[v4rvh@gc112m30 Lab1]$ ./Exercise2
Hello World
[v4rvh@gc112m30 Lab1]$ pwd
/home1/ugrads/v4rvh/CS2263/Labs/Lab1
[v4rvh@gc112m30 Lab1]$ ls
a.out                         Exercise1.1.png  Exercise2      Exercise4.c
Daniyal_Khan_CS2263_Lab1.odt  Exercise1.class  Exercise2.c    Exercise4.png
Excerise1.png                 Exercise1.java   Exercise2.png
[v4rvh@gc112m30 Lab1]$
```

Exercise 3:

The java compilation model takes human written source code as input and emits not machine code but something called “byte code”. Byte code cannot be directly executed on a machine. Instead, it needs to be translated once again by another compiler to machine code. Byte code can run on any platform that has either a byte code compiler or virtual machine written for it. While in C language there is not such protability implied by the machine code.

To answer the question, ***javac*** compiles the code similar to the ***gcc*** command for c and c++. But the difference is that gcc produces an executable file whereas ***javac*** produces a bytecode which we run by the ***java*** command.

Referenced from this question on stackOverFlow:

https://stackoverflow.com/questions/8297832/how-does-c-differ-from-javas-compilation-model

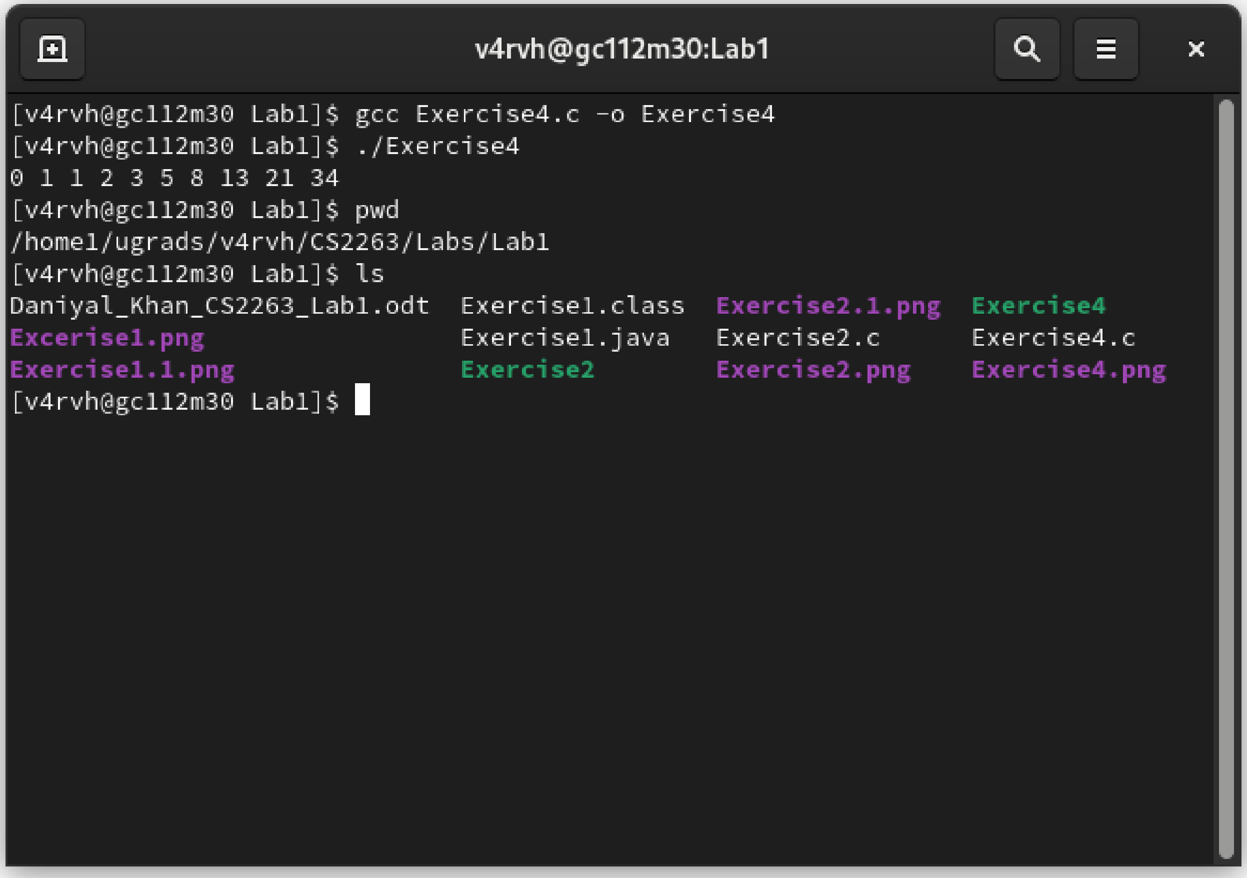
Exercise 4:

```
#include <stdio.h>
#include <stdlib.h>

void printFib(int n);

int main() {
printFib(10);
putchar('\n');
return EXIT_SUCCESS;
}

void printFib(int n) {
int a = 0, b = 1, temp;
if (n <= 0) {
return;
}
printf("%d ", a);
for (int i = 1; i < n; i++) {
printf("%d ", b);
temp = a + b;
a = b;
b = temp;
}
}
```



```
[v4rvh@gc112m30 Lab1]$ gcc Exercise4.c -o Exercise4
[v4rvh@gc112m30 Lab1]$ ./Exercise4
0 1 1 2 3 5 8 13 21 34
[v4rvh@gc112m30 Lab1]$ pwd
/home1/ugrads/v4rvh/CS2263/Labs/Lab1
[v4rvh@gc112m30 Lab1]$ ls
Daniyal_Khan_CS2263_Lab1.odt  Exercise1.class  Exercise2.1.png  Exercise4
Excerise1.png                 Exercise1.java   Exercise2.c      Exercise4.c
Exercise1.1.png               Exercise2        Exercise2.png    Exercise4.png
[v4rvh@gc112m30 Lab1]$ 
```