

Part 1: Java and C

Java:

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello world!");  
    }  
}
```

```
[ishoebot@id414m10 Java]$ ls  
HelloWorld.class HelloWorld.java
```

```
File Edit View Search Terminal Help  
[ishoebot@id414m10 Java]$ javac HelloWorld.java  
[ishoebot@id414m10 Java]$ java Hello  
HelloWorld.class HelloWorld.java  
[ishoebot@id414m10 Java]$ java HelloWorld.class  
Error: Could not find or load main class HelloWorld.class  
Caused by: java.lang.ClassNotFoundException: HelloWorld.class  
[ishoebot@id414m10 Java]$ java HelloWorld  
Hello world!  
[ishoebot@id414m10 Java]$
```

C:

```
#include <stdio.h>  
  
int main() {  
    printf("Hello world!\n");  
    return 0;  
}
```

```
[ishoebot@id414m10 C]$ ls  
HelloWorld HelloWorld.c
```

```
File Edit View Search Terminal Help  
[ishoebot@id414m10 C]$ gcc -o HelloWorld HelloWorld.c  
[ishoebot@id414m10 C]$ ./HelloWorld  
Hello world!  
[ishoebot@id414m10 C]$
```

Explain the difference:

The difference between a C program and a Java program is as follows: Java is traditionally compiled JIT, so its program files are stored in an intermediate format known as bytecode. This lets the java developers write a native VM for each supported platform and this bytecode can run independently of operating system. C on the other hand is compiled to assembly for the processor architecture, as well as the operating systems executable headers/format. These executables are executed straight from the processor with not virtual machine involved and are often platform dependant. The reason java programs need to be executed with a java command beforehand is that you need to tell the native java VM to read this bytecode. The system does not natively understand java bytecode.

(Source: <https://stackoverflow.com/a/1326084>)

Part 2: countOnes:

```
[ishoebot@id414m10 Lab1]$ cat countOnes.c
#include <stdio.h>
int main(){
    int value;
    int iErr;

    printf("Value to examine: ");

    iErr = scanf("%d",&value);

    if(iErr != 1){
        printf("Unable to read the value\n");
        return 0;
    }
    if(value <= 0) {
        printf("Value must be positive\n");
        return 0;
    }
    int binaryArray[32];
    int arrayLength = 0;
    while(value > 0) {
        binaryArray[arrayLength] = value % 2;
        value = value/2;
        arrayLength++;
    }
    int counter = 0;
    for(int i = arrayLength; i >= 0; i--) {
        if (binaryArray[i] == 1) {
            counter++;
        }
    }
    printf("Number of ones: %d\n", counter);
    return 0;
}
```

```
[ishoebot@id414m10 CountOnes]$ ls
countOnes  countOnes.c  countOnes.exe
```

```
ishoebot@id414m10:Lab1
File Edit View Search Terminal Help
[ishoebot@id414m10 Lab1]$ gcc -o count0nes -std=c99 count0nes.c
[ishoebot@id414m10 Lab1]$ ./count0nes
Value to examine: 1
Number of ones: 1
[ishoebot@id414m10 Lab1]$ ./count0nes
Value to examine: 52
Number of ones: 3
[ishoebot@id414m10 Lab1]$ ./count0nes
Value to examine: 5645645645
Number of ones: 14
[ishoebot@id414m10 Lab1]$
```