Question on Fork System Call | Operating System – M01 P09

This is a multipart blog article series, and in this series I am going to explain you the concepts of operating system. This article series is divided into multiple modules and this is the first module which consists of 12 articles.

In this article we will see a question on fork system call, and by that question I will try to explain you the concept of fork system call.

So, in the previous article I had explained about fork system call, it is recommended that first you go through that article for getting a basic understanding of the topic, Fork system call (<https://programmingport.hashnode.dev/fork-system-call-or-operating-system-m01-p08>) (<https://dev.to/rahulmishra05/fork-system-call-operating-system-m01-p08-27hp>)

I am going to use a C program to explain you about fork system call.

#include <stdio.h>

#include <unistd.h>

Int main() {

If(fork () && fork())

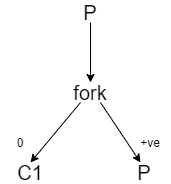
Fork()

Printf(“Hello”);

Return 0;

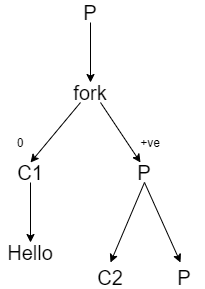
}

Now, the compiler will execute the code and when the “if” statement come, the first “fork” command will be executed, and a child process will be created, and both the process (child process and parent process) will run parallel, as you can see in the below diagram.



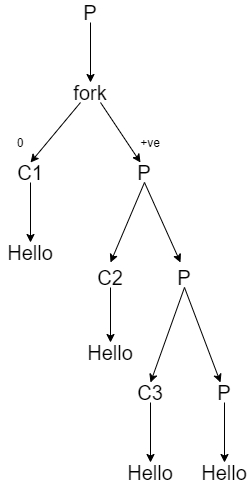
Now the child process “C1” will print “Hello”.

As you can see that AND operator is written after first “fork()” so it will also get executed, so another child process from parent process will get created. As you can see in the diagram below.



This C2 will also print “Hello”

Now the “if” statement is completely executed, and now the “fork()” statement written inside the “if” statement will get executed. And again a child process of parent process will be created. As you can see in the diagram below.



This child process and parent process will print “Hello”.

Here, in total fork is done 3 times, and in total 4 times “Hello” will be printed.

Hope you learned something. If you have any question, query, doubt or just want to share something with me, then please feel free to contact me.