

#### Práctica 4: Creación de procesos con Fork y gestión de procesos con htop

Sistemas Operativos NRC 18149

Natalia Arboleda Arboleda ID 000165655

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

hijo[1] --> pid = 32635 y ppid = 32631, acumulado = 0
hijo[1] --> pid = 32635 y ppid = 32631, acumulado = 1
hijo[1] --> pid = 32635 y ppid = 32631, acumulado = 3
hijo[1] --> pid = 32635 y ppid = 32631, acumulado = 6
hijo[1] --> pid = 32635 y ppid = 32631, acumulado = 10
hijo[1] --> pid = 32635 y ppid = 32631, acumulado = 15

hijo[2] --> pid = 32636 y ppid = 32631, acumulado = 0
hijo[2] --> pid = 32636 y ppid = 32631, acumulado = 1
hijo[2] --> pid = 32636 y ppid = 32631, acumulado = 3
hijo[2] --> pid = 32636 y ppid = 32631, acumulado = 6
hijo[2] --> pid = 32636 y ppid = 32631, acumulado = 10
hijo[2] --> pid = 32636 y ppid = 32631, acumulado = 15

hijo[3] --> pid = 32637 y ppid = 32631, acumulado = 0
hijo[3] --> pid = 32637 y ppid = 32631, acumulado = 1
hijo[3] --> pid = 32637 y ppid = 32631, acumulado = 3
hijo[3] --> pid = 32637 y ppid = 32631, acumulado = 6
hijo[3] --> pid = 32637 y ppid = 32631, acumulado = 10
hijo[3] --> pid = 32637 y ppid = 32631, acumulado = 15

padre --> pid = 32631, acumulado = 0
padre --> pid = 32631, acumulado = 1
padre --> pid = 32631, acumulado = 3
padre --> pid = 32631, acumulado = 6
padre --> pid = 32631, acumulado = 10
padre --> pid = 32631, acumulado = 15

[1] + Done                                "/usr/bin/gdb" --interpreter=mi --tty=$
root@DESKTOP-K51BEFA:~/practica4#
```

## Código

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/wait.h>
#include <sys/types.h>

int main()
{
    int pid, pid1, pid2, i;
    int acumulado = 0;
    pid = fork();

    if (pid == 0) {
        for(i=0; i<=5; i++){
            acumulado += i;
            printf("hijo[1] --> pid = %d y ppid = %d, acumulado = %d\n",
getpid(), getppid(), acumulado);
        }
        printf("\n");
    }

    else {

        wait(NULL);
        pid1 = fork();

        if (pid1 == 0) {
            for(i=0; i<=5; i++){
                acumulado += i;
                printf("hijo[2] --> pid = %d y ppid = %d, acumulado = %d\n",
getpid(), getppid(), acumulado);
            }
            printf("\n");
        }

        else {

            wait(NULL);
            pid2 = fork();

            if (pid2 == 0) {
                for(i=0; i<=5; i++){
                    acumulado += i;
```

```
        printf("hijo[3] --> pid = %d y ppid = %d, acumulado  
= %d\n", getpid(), getppid(), acumulado);  
    }  
    printf("\n");  
}  
  
else {  
  
    wait(NULL);  
    for(i=0; i<=5; i++){  
        acumulado += i;  
        printf("padre --> pid = %d, acumulado = %d\n", getpid(),  
acumulado);  
    }  
    printf("\n");  
}  
}  
}  
  
return 0;  
}
```