

REPORTE

PRIMERA PARTE

1. ¿Se imprimen en el orden esperado?

Se debería imprimir primero el "Hello it's me?", primero se ejecuta el proceso hijo, luego el padre"

2. ¿Siempre se imprimen en el mismo orden?

Si lo hace

3. ¿A sus compañeros les sucede lo mismo?

Si

```
c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$ make
gcc -o practica-01-fork-01 practica-01-fork-01.c
./practica-01-fork-01

HELLOHello, it's me

HELLOHello from the other side

c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$
```

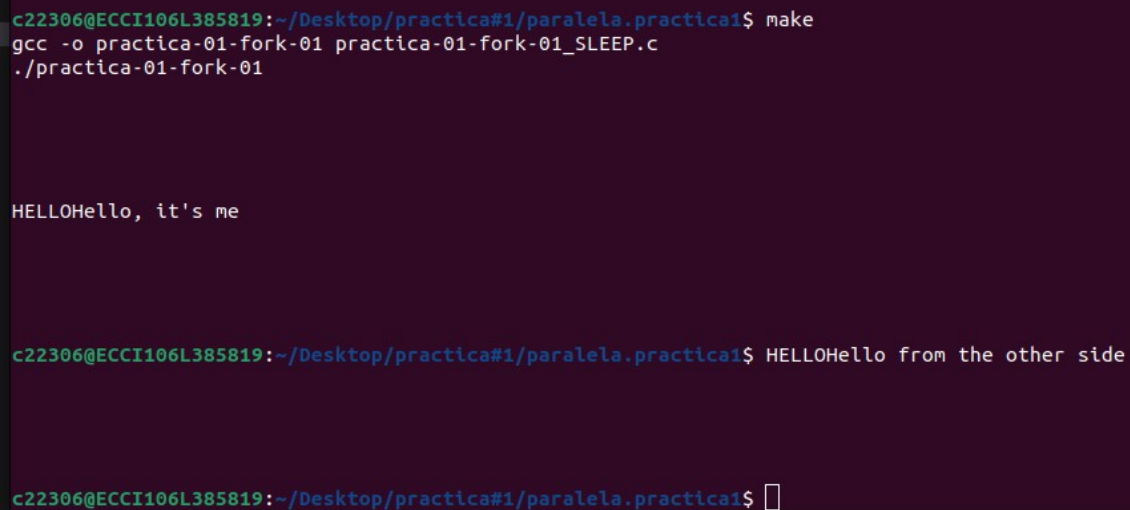
SEGUNDA PARTE

1. ¿Se imprimen en el orden esperado?

Ahora si imprime en el orden esperado, imprime primero "Hello it's me"

2. ¿Cómo imprime cada proceso?

Si se pone el comando `sleep(5);` en el inicio de un mensaje y no en el otro se termina la ejecución y una vez pasa el tiempo de sleep se despierta la ejecución, algo así:



```
c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$ make
gcc -o practica-01-fork-01 practica-01-fork-01_SLEEP.c
./practica-01-fork-01

HELLOHello, it's me

c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$ HELLOHello from the other side

c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$
```

3. ¿Se ejecuta el proceso hijo antes o después del proceso padre?

Depende de donde pongamos el `sleep(5);`

PD: Creo que el SLEEP me congela la terminal

TERCERA PARTE

[WaitPID documentation](#)

1. ¿Se imprimen en el orden esperado?

Si, usando el waitpid() si se ejecuta como yo quiero jeje

2. ¿Cómo imprime cada proceso?

Se imprime bien porque se usa el waitPID entonces to good

3. ¿Se ejecuta el proceso hijo antes o después del proceso padre?

Depende de donde ponga yo el WaitPID()

```
c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$ make
gcc -o practica-01-fork-01 practica-01-fork-01_waitpid.c
./practica-01-fork-01
Hello from the other side
I'm your father hello, it's me
c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$
```

CUARTA PARTE

1. ¿Cuántos procesos hijos se crean?

5 procesos hijos

2. ¿En qué orden se crean los procesos hijos?

En orden numérico

3. ¿En qué orden se ejecutan los procesos hijos? ¿Siempre es el mismo orden?

Se ejecutan en orden de 0 a 4, no obstante, el "DONE" que aparece al final del código se imprime dónde se le pega la gana

4. ¿Cómo sabe el proceso hijo cual es su número?

Porque el print está dentro del for y en teoría deben imprimir los hijos en orden

```
Main process
I'm child process number 0
I'm child process number 1
I'm child process number 2
DONE
I'm child process number 3
I'm child process number 4
c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$ make
gcc -o practica-01-fork-01 practica-01-fork-04.c
./practica-01-fork-01
Main process
I'm child process number 0
I'm child process number 1
I'm child process number 2
DONE
I'm child process number 3
I'm child process number 4
c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$ make
gcc -o practica-01-fork-01 practica-01-fork-04.c
./practica-01-fork-01
Main process
I'm child process number 0
I'm child process number 1
I'm child process number 2
DONE
I'm child process number 4
I'm child process number 3
c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$ make
gcc -o practica-01-fork-01 practica-01-fork-04.c
./practica-01-fork-01
Main process
I'm child process number 0
I'm child process number 1
DONE
I'm child process number 2
I'm child process number 3
I'm child process number 4
c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$
```

QUINTA PARTE

1. ¿Ahora se imprimen en el orden esperado?

Si se imprimen en el orden esperado

2. ¿El proceso padre espera a que todos los procesos hijos terminen antes de terminar?

Si todo bien

```
Main process
I'm child process number 0
I'm child process number 1
I'm child process number 2
I'm child process number 3
I'm child process number 4
DONE
c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$ make
gcc -o practica-01-fork-01 practica-01-fork-05.c
./practica-01-fork-01
Main process
I'm child process number 0
I'm child process number 1
I'm child process number 2
I'm child process number 3
I'm child process number 4
DONE
c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$ make
gcc -o practica-01-fork-01 practica-01-fork-05.c
./practica-01-fork-01
Main process
I'm child process number 0
I'm child process number 1
I'm child process number 2
I'm child process number 3
I'm child process number 4
DONE
c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$ make
gcc -o practica-01-fork-01 practica-01-fork-05.c
./practica-01-fork-01
Main process
I'm child process number 0
I'm child process number 1
I'm child process number 2
I'm child process number 3
I'm child process number 4
DONE
c22306@ECCI106L385819:~/Desktop/practica#1/paralela.practica1$
```

SEXTA PARTE

1. ¿Qué valor tiene la variable msg al final del programa?

El vaor que tiene es "I'm the parent process!!!"

2. ¿Por qué el valor de msg cambia o no?

Porque el msg que se imprime al inicio y al final no es el mismo ya

3. ¿Porqué los hijos pueden imprimir el mensaje correcto?

Porque al hacer el fork se hace una copia de la memoria, se puede decir que comparten una misma memoria, claro, como lo ha explicado el profe, utilizan memoria virtual, o sea, que comparten memoria hasta que vayan a escribir, ahí si hacen una copia

4. Al final, qué valor tiene la variable msg? ¿Porqué?

Tiene el valor que se le dió al inicio del programa porque la variable "msg" se cambió dentro de los hilos, o sea, que se hizo una copia de esa variable y la original no se modificó, la original se creó en el proceso padre y ahí sigue

5. ¿Qué pasó con la asignación de memoria de msg realizada en los hijos?

Se hace una copia para cada mensaje creado dentro de los hijos, ya que están queriendo explicar y por lo anterior mencionado el sistema lo que hace es crear una copia de la variable original y modificarla ya que los hijos no tienen acceso a editar esa variable

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.practical$ make
gcc -o practica-01-fork-01 practica-01-fork-06.c
./practica-01-fork-01
Process start
I'm the parent process!!!
I'm child process number 0
I'm child process number 1
I'm child process number 2
I'm child process number 3
I'm child process number 4
I'm child process number 5
I'm child process number 6
I'm child process number 7
I'm child process number 8
I'm child process number 9
The program ends
The final value of msg is: I'm the parent process!!!

mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.practical$
```

SÉPTIMA PARTE

1. ¿La dirección de memoria de msg es la misma en el proceso padre y en los procesos hijos?

Si es la misma

2. ¿Porqué?

Porque se está cambiando solo un dato, entonces se hace una copia pero solo de esa parte de la cadena y el sistema sigue usando el inicio de la cadena la cual no hace falta copiar

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.practical$ make
gcc -o practica-01-fork-01 practica-01-fork-07.c
./practica-01-fork-01
Process start
[Parent] The address of msg is: 0x559cf545e6b0 | Value is: I'm the parent process!!!
[Child 0] The address of msg is: 0x559cf545e6b0 | Value is: hello from child 0
[Child 1] The address of msg is: 0x559cf545e6b0 | Value is: hello from child 1
[Child 2] The address of msg is: 0x559cf545e6b0 | Value is: hello from child 2
[Child 3] The address of msg is: 0x559cf545e6b0 | Value is: hello from child 3
[Child 4] The address of msg is: 0x559cf545e6b0 | Value is: hello from child 4
[Child 5] The address of msg is: 0x559cf545e6b0 | Value is: hello from child 5
[Child 6] The address of msg is: 0x559cf545e6b0 | Value is: hello from child 6
[Child 7] The address of msg is: 0x559cf545e6b0 | Value is: hello from child 7
[Child 8] The address of msg is: 0x559cf545e6b0 | Value is: hello from child 8
[Child 9] The address of msg is: 0x559cf545e6b0 | Value is: hello from child 9
[Parent] The program ends
[Parent] The final value of msg is: I'm the parent process!!!
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.practical$
```

OCTAVA PARTE

1. ¿Cuánta memoria gasta el programa (incluyendo cada proceso)?

*En términos de memoria, 1 gigabyte (GB) es igual a 1024 megabytes (MB), y cada MB es igual a 1024 kilobytes (KB), lo que significa que se está asignando $1024 * 1024$ bytes de memoria, que equivale a 1 GB.*

2. ¿Estamos gastando 100GB de memoria o no?

No

```
[Child 66] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 66
[Child 67] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 67
[Child 68] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 68
[Child 69] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 69
[Child 70] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 70
[Child 71] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 71
[Child 72] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 72
[Child 73] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 73
[Child 74] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 74
[Child 75] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 75
[Child 76] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 76
[Child 77] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 77
[Child 78] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 78
[Child 79] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 79
[Child 80] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 80
[Child 81] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 81
[Child 82] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 82
[Child 83] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 83
[Child 84] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 84
[Child 85] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 85
[Child 86] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 86
[Child 87] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 87
[Child 88] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 88
[Child 89] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 89
[Child 90] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 90
[Child 91] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 91
[Child 92] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 92
[Child 93] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 93
[Child 94] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 94
[Child 95] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 95
[Child 96] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 96
[Child 97] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 97
[Child 98] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 98
[Child 99] The address of msg is: 0x7fc993e7a010 | Value is: hello from child 99
[Parent] The program ends
[Parent] The final value of msg is: I'm the parent process!!!
```

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.practical$
```


NOVENA PARTE

1. ¿Cuánta memoria gasta el programa (incluyendo cada proceso)?

Gastó 8GB

2. ¿Ahora sí se gasta más memoria?

Si claro

3. ¿Porqué?

Por el memset() ya que ahora si se está dando la instrucción de llenar la memoria

4. ¿Colgó la computadora? ¿Porqué?

La mía no se colgó, solo se puso lenta, tal vez sí hubiese usado más memoria de la que tengo disponible si se hubiese colgado

Process Name	User	% CPU	ID	Memory	Disk read total	Disk write
practica-01-fork-01	mariocordero	0,00	106212	1,1 GB	N/A	
practica-01-fork-01	mariocordero	0,00	106214	1,1 GB	N/A	
practica-01-fork-01	mariocordero	0,00	106215	1,1 GB	N/A	
practica-01-fork-01	mariocordero	0,00	106216	1,1 GB	N/A	
practica-01-fork-01	mariocordero	1,54	106217	1,1 GB	N/A	
practica-01-fork-01	mariocordero	0,00	106218	1,1 GB	N/A	
practica-01-fork-01	mariocordero	0,00	106210	1,1 GB	N/A	
soffice.bin	mariocordero	0,00	5468	202,8 MB	181,2 MB	25,1 MB
code	mariocordero	0,00	5587	195,1 MB	31,9 MB	2,1 MB

```
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.practical$ make
gcc -o practica-01-fork-01 practica-01-fork-08.c
practica-01-fork-08.c: In function 'main':
practica-01-fork-08.c:15:5: warning: implicit declaration of function 'memset' [-Wimplicit-function-declaration]
   15 |     memset(msg, 8, 1024 * 1024 * 1024);
      |     ^~~~~~
practica-01-fork-08.c:9:1: note: include '<string.h>' or provide a declaration of 'memset'
     8 | #include <fcntl.h>
+++  |+#include <string.h>
     9 |
practica-01-fork-08.c:15:5: warning: incompatible implicit declaration of built-in function 'memset' [-Wbuiltin-declaration-mismatch]
   15 |     memset(msg, 8, 1024 * 1024 * 1024);
      |     ^~~~~~
practica-01-fork-08.c:15:5: note: include '<string.h>' or provide a declaration of 'memset'
./practica-01-fork-01
Process start
[Parent] The address of msg is: 0x7f650ea89010 | Value is: I'm the parent process!!!

[Child 0] The address of msg is: 0x7f650ea89010 | Value is:
[Child 3] The address of msg is: 0x7f650ea89010 | Value is:
[Child 6] The address of msg is: 0x7f650ea89010 | Value is:
[Child 7] The address of msg is: 0x7f650ea89010 | Value is:

[Parent] The program ends
[Parent] The final value of msg is: I'm the parent process!!!
mariocordero@mariocordero-B450M-AORUS-MONSTER:~/Desktop/progra_paralela/paralela.practical$
```