

Programación Concurrente, Febrero 2023

Práctica 1

Martínez Buenrostro Jorge Rafael.
Universidad Autónoma Metropolitana
Unidad Iztapalapa, México
molap96@gmail.com

Respuestas ejercicio C

```
if ((cpid = fork()) == 0)
    printf("Hola mundo");
```

“Hola mundo” se escribe una vez, la cual la realiza el proceso hijo

```
if ((cpid = fork()) != 0)
    printf("Hola mundo");
```

“Hola mundo” se escribe una vez, la cual la realiza el proceso padre

```
for( i = 0; i < n; i++) {
    cpid = fork();
    if (cpid == 0) {
        printf("Hola mundo");
        exit();
    }
}
```

Cada hijo que se crea imprime un “Hola mundo”, por lo que se imprime 3 veces

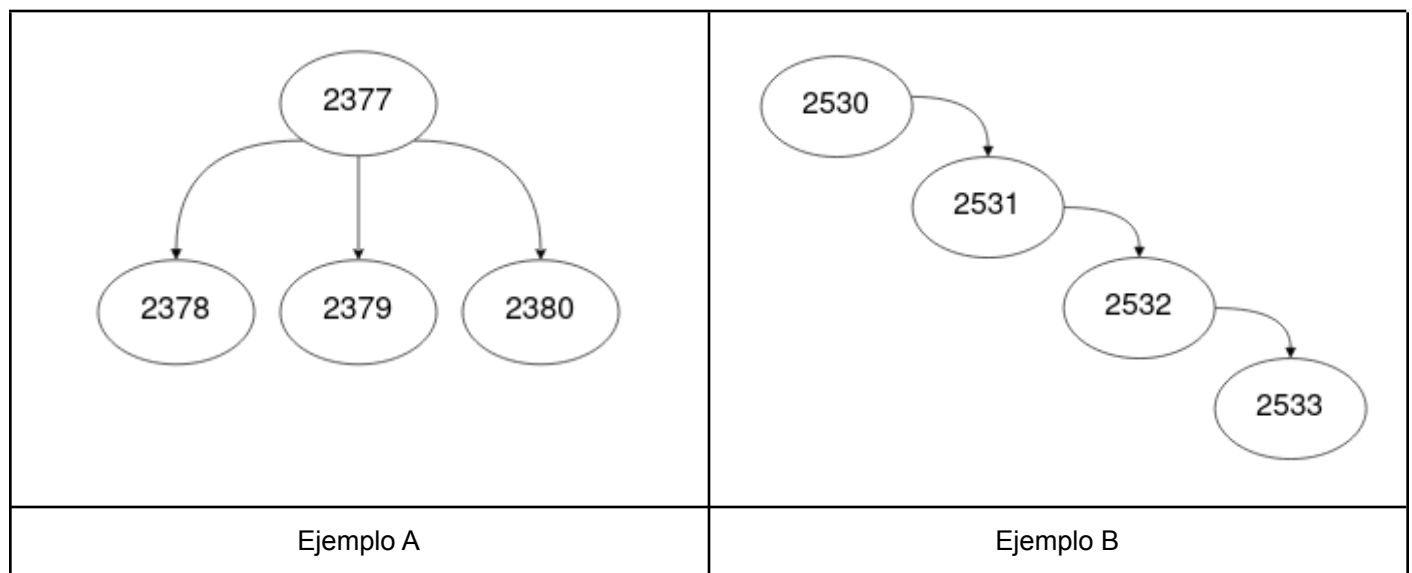
```

for( i = 0; i < n; i++) {
    cpid = fork();
    if (cpid != 0) {
        printf("Hola mundo");
        wait(----);
        exit(----);
    }
}

```

En cada iteración se crea un hijo, el padre de este hijo es el que imprime "Hola mundo". En la siguiente iteración el hijo que se creó se vuelve el nuevo padre. Por lo que "Hola mundo" se imprime 3 veces

Grafos inciso D



Enlace GDB online

- <https://onlinegdb.com/kWcsEBvru>

EjercicioA

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

int main() {
    printf("\n**** Ejercicio a ****\n");
    printf("\n\tC*x^n\n\n");

    pid_t pid;
    int valor_salida, c, x_base, x_resultado, n, i;
    //c*x^n

    printf("¿Cuál es el valor de x? ");
    scanf("%d", &x_base);
    x_resultado = x_base;
    printf("¿Cuál es el valor de n? ");
    scanf("%d", &n);
    printf("¿Cuál es el valor de c? ");
    scanf("%d", &c);
    printf("\n");

    for(i=0; i<n; i++) {
        pid = fork();
        if(pid != 0) {
            printf("\nProceso %d, padre %d\n", getpid(), getppid());
            printf("\tResultado del cálculo x^%d: %d\n", i+1, x_resultado);
            //printf("Soy %d, mi padre es %d y el resultado de mi calculo
es=%d\n", getpid(), getppid(), x_resultado);
            wait(&valor_salida);
            exit(WEXITSTATUS(valor_salida));
        }
        if(i != n-1) {
            x_resultado *= x_base;
        }
    }
}
```

```

}

printf("\nProceso %d, padre %d\n",getpid(),getppid());
printf("\tResultado del cálculo  $c \cdot x^n =$  %d\n",c*x_resultado);
return 0;
}

```

Ejercicio B

```

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

int menorDivisor(int);
int encuentraPrimos(int,int);

int main(){
    printf("\n**** Ejercicio b ****\n");
    printf("\n\t[1,M]\n\n");
    pid_t pid;
    int valor_salida, i, m,numeroHijos,sumaPrimos=0;

    printf("Ingrese el valor de M...");
    scanf("%d",&m);
    numeroHijos=menorDivisor(m);

    for( i = 0; i < numeroHijos; i++) {
        pid = fork();
        if (pid == 0) {
            sumaPrimos=encuentraPrimos((m/numeroHijos*i)+1,(m/numeroHijos)*(i+1));
            printf("\nProceso %d, padre %d.\n",getpid(),getppid());
            printf("\tEncontró %d primos en el rango
[%d,%d]\n",sumaPrimos,(m/numeroHijos*i)+1,(m/numeroHijos)*(i+1)),
            exit(sumaPrimos);

```

```

    }

}

for( i = 0; i < numeroHijos; i++) {
    pid = wait(&valor_salida);
    sumaPrimos+=WEXITSTATUS(valor_salida);
}

printf("\nProceso %d, padre %d\n",getpid(),getppid());
printf("\tLos primos que se encontraron entre [1,%d] es de %d\n",
m,sumaPrimos);

return 0;
}

int menorDivisor(int m){
    int i;
    for(i = 2; i <= m; i++) {
        if((m%i) == 0){
            return i;
        }
    }
}

int encuentraPrimos(int a, int b){
    int numeroPrimos=0,esPrimo, i,j;

    for(i=a;i<=b;i++){
        if(i==1 || i==0)
            continue;
        esPrimo = 1;
        for(j=2;j<= i/2;++j){
            if(i%j==0){
                esPrimo = 0;
                break;
            }
        }
        if (esPrimo == 1)

```

```
        numeroPrimos++;  
    }  
    return numeroPrimos;  
}
```