

PARCIAL 1ª EVALUACIÓN			
MÓDULO:	Programación de Servicios y Procesos	U.T.:	1
GRUPO:	CF GS DAM (2º)	FECHA:	05/11/2023

Nombre alumno: _____

RESULTADOS DE APRENDIZAJE Y CRITERIOS DE EVALUACIÓN RELACIONADOS

RA1. Desarrolla aplicaciones compuestas por varios procesos reconociendo y aplicando principios de programación paralela.

CE	Calificación
a) Se han reconocido las características de la programación concurrente y sus ámbitos de aplicación. b) Se han identificado las diferencias entre programación paralela y programación distribuida, sus ventajas e inconvenientes. c) Se han analizado las características de los procesos y de su ejecución por el sistema operativo. d) Se han caracterizado los hilos de ejecución y descrito su relación con los procesos. e) Se han utilizado clases para programar aplicaciones que crean subprocesos. f) Se han utilizado mecanismos para sincronizar y obtener el valor devuelto por los subprocesos iniciados. g) Se han desarrollado aplicaciones que gestionen y utilicen procesos para la ejecución de varias tareas en paralelo. h) Se han depurado y documentado las aplicaciones desarrolladas.	

Esta prueba tiene una puntuación de 0 a 10 puntos.

Ejercicio 1: (5 puntos)

Realiza en Java un programa PP que reciba entre 2 y 10 números por argumentos. (2 y 10 incluidos) y devuelva por pantalla un texto dependiendo del número de primos que se le hayan pasado.

Instrucciones:

- PP guardará los argumentos recibidos desde la entrada estándar en una lista de enteros (List<Integer>, por ejemplo) o alguna estructura de array o arrayList.
- PP debe comprobar si cada elemento de la lista es un número múltiplo de 5 o no. Para hacerlo, llamará a PH tantas veces como números se hayan pasado al programa.
- PP escribirá en stdin de PH el número en cuestión.
- PP leerá el stdout de PH, que será "SI" en caso de que el número sea múltiplo de 5; o "NO" si el número no lo es. (SI o NO se recibirán sin comillas).

- Finalmente, PP imprimirá por su stdout **“PARiendo cincos”** si en la lista hay un número par de números múltiplos de 5; o **“sIMPARTicos cincos”** si en la lista hay un número impar de números múltiplos de 5.

Nota:

- ✓ PH es programa de entrada/salida. Por su entrada recibe un número entero y por su salida saca “SI” o “NO”, dependiendo si el número es primo o no.

Se entregarán los ficheros java con la solución.

Ejercicio 2: (2,5 puntos)

Partiendo del ejercicio anterior, modifícalo para que PP recoja “SI” o “NO”, dependiendo si el número es primo o no por parte del hijo y crea y llama a otro proceso hijo diferente, PH2, que reciba por su entrada estándar, stdin, la salida de PH y devuelva por salida estándar los mensajes (“PARiendo primos” y “sIMPARTicos primos tengo”) que en un principio sacaba PP y que ahora recogerá de PH2 para mostrarlos por consola.

Se entregarán los ficheros java con la solución.

Ejercicio 3: (2,5 puntos)

Escribe un programa Aleatorios que haga lo siguiente:

- Cree un programa (futuro proceso hijo) que se encargue de generar números pares aleatorios. Este proceso hijo escribirá en su salida estándar un número aleatorio del 0 al 20 cada vez que reciba una petición de ejecución por parte del padre.
- El proceso padre lee líneas de la entrada estándar y por cada línea que lea solicitará al hijo que le envíe un número aleatorio, lo leerá y lo imprimirá en pantalla.
- Cuando el proceso padre reciba la palabra "fin", finalizará la ejecución del hijo y procederá a finalizar su ejecución.
- Ejemplo de ejecución:

```
ab (enter)
7
```

```
abcdef (enter)
1
```

```
Pepe (enter)
6
```

```
fin (enter)
```

Se debe entregar el código fuente (del padre y del hijo) y el fichero ejecutable del hijo.