

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**  
**FACULTAD DE CIENCIAS E INGENIERÍA**

**SISTEMAS OPERATIVOS**

**Examen 1**

**(Segundo semestre de 2020)**

Horario 0781: prof. V. Khlebnikov

Horario 0782: prof. F. Solari A.

Duración: 3 horas

Nota: **La presentación, la ortografía y la gramática influirán en la calificación.**

Puntaje total: 20 puntos

**Pregunta 1 (5 puntos – 30 min.)**

El archivo de su respuesta debe estar en la carpeta **Buzón 1** del **Examen 1** en PAIDEIA **antes de las 08:45**. Por cada 3 minutos de retardo son -2 puntos.

El nombre de su archivo debe ser `<su_código_de_8_dígitos>_11.txt`. Por ejemplo, `20202912_11.txt`.

Responda en forma breve, concisa y justificada, las siguientes cuestiones:

**a) (1.0 punto)** El *shell* puede ser gráfico o de línea de comando, pero, ¿es el *shell* el sistema operativo? ¿un módulo del mismo? ¿algo diferente a lo anterior? Explique brevemente.

**b) (1.0 punto)** ¿Qué problema del *hardware* pretende resolver el Sistema Operativo, y con qué concepto corresponde?

**c) (0.5 puntos)** ¿Cómo termina un proceso antes de llegar al final del código? Mencione al menos dos casos.

**d) (0.5 puntos)** ¿Cuándo sale un proceso del estado *BLOCKED* (en espera) al estado *READY* (listo para ejecutar)?

**e) (2 puntos)** Explique la ejecución (los procesos creados y su interacción) **(1 punto)**, y el resultado **(1 punto)** de:

```
$ find /var -name "*.log" | head -10 | cat -n
find: '/var/tmp/systemd-private-9a52e6c4dea84c5fac99b14b94c5eb4c-ntp.service-5c6oXi': Permission denied
find: '/var/tmp/systemd-private-9a52e6c4dea84c5fac99b14b94c5eb4c-systemd-logind.service-grnDyf': Permission
denied
find: '/var/tmp/systemd-private-9a52e6c4dea84c5fac99b14b94c5eb4c-ModemManager.service-MZo8Zh': Permission
denied
find: '/var/tmp/systemd-private-9a52e6c4dea84c5fac99b14b94c5eb4c-systemd-resolved.service-gmNLHh': Permission
denied
find: '/var/tmp/systemd-private-9a52e6c4dea84c5fac99b14b94c5eb4c-upower.service-tCixCg': Permission denied
1 /var/log/mintsystem.log
2 /var/log/fontconfig.log
3 /var/log/ubuntu-system-adjustments-adjust-grub-title.log
4 /var/log/boot.log
5 /var/log/ubuntu-system-adjustments-stop.log
6 /var/log/kern.log
7 /var/log/vboxadd-install.log
8 /var/log/ubuntu-system-adjustments-start.log
9 /var/log/Xorg.1.log
10 /var/log/gpu-manager.log
find: '/var/log/speech-dispatcher': Permission denied
```

## Pregunta 2 (5 puntos – 30 min.)

El archivo de su respuesta debe estar en la carpeta **Buzón 2** del **Examen 1** en PAIDEIA **antes de las 09:30**. Por cada 3 minutos de retardo son -2 puntos.

El nombre de su archivo debe ser `<su_código_de_8_dígitos>_12.txt`. Por ejemplo, `20202912_12.txt`.

Implementaremos la siguiente idea en el programa con nombre “`_0_2020-2.c`” en el lenguaje C:

```
int
main(int argc, char **argv) {
...
}
```

Primero compararemos la cadena del argumento 0 con la cadena “`./_0_2020-2`”. Estas deben coincidir y el resultado será 0. En la implementación del lenguaje C que usamos el resultado también puede ser negativo o positivo y el valor del resultado (en la variable `r`, por ejemplo) será el valor de la diferencia, por ejemplo, comparando “B” con “A” el resultado será 1 y comparando “C” con “A” será 2, o sea la diferencia en el orden lexicográfico.

Entonces, cuando nuestras cadenas coinciden, aseguraremos que tenemos el argumento 1 proporcionado al programa para su ejecución y convertiremos su cadena en un entero `n`. En la variable `np` tendremos la cadena del argumento 0. Ahora haremos `n` repeticiones de lo siguiente:

```
incrementaremos np[2] en 1,
unlink(np),
link del argumento 0 con np,
hacer fork y si la negación del resultado de fork es cierto, entonces
    execl(np,np,argv[1],(char *)NULL);
    write(1,"Oops...\n",8);
    exit(EXIT_FAILURE);
```

Pero si la cadena del argumento 0 es diferente de la cadena “`./_0_2020-2`”, entonces decrementaremos la diferencia `r` en 1 y si ahora la diferencia se evalúa como cierto, dormiremos 1 segundo. En seguida,

```
execl(a[r][0],a[r][1],(char *)NULL);
write(1,"Oops...\n",8);
exit(EXIT_FAILURE);
```

donde la variable `a` está declarada como

```
char a[4][2][20] = {
    { "/bin/ps", "ps" }
    { "/bin/pwd", "pwd" }
    { "/bin/date", "date" }
    { "/usr/bin/ncal", "ncal" }
};
```

Independiente del resultado de la comparación de las cadenas del argumento 0 y “`./_0_2020-2`”, acabamos el programa con

```
while (waitpid(-1,NULL,0) != -1);
exit(EXIT_SUCCESS);
```

Cuando logramos compilar el programa sin errores y lo ejecutamos como

```
$ ./_0_2020-2 4
```

la ejecución será sin errores y se obtendrá la salida deseada.

**a) (2 puntos)** ¿Qué se pretende lograr con la ejecución indicada del programa descrito? Explíquelo describiendo los procesos, la relación entre ellos, los programas que ejecutan y usando otros conceptos vistos en este curso.

**b) (1 punto)** Describa con detalles el primer bloque de impresión en la salida de la ejecución indicada del programa.

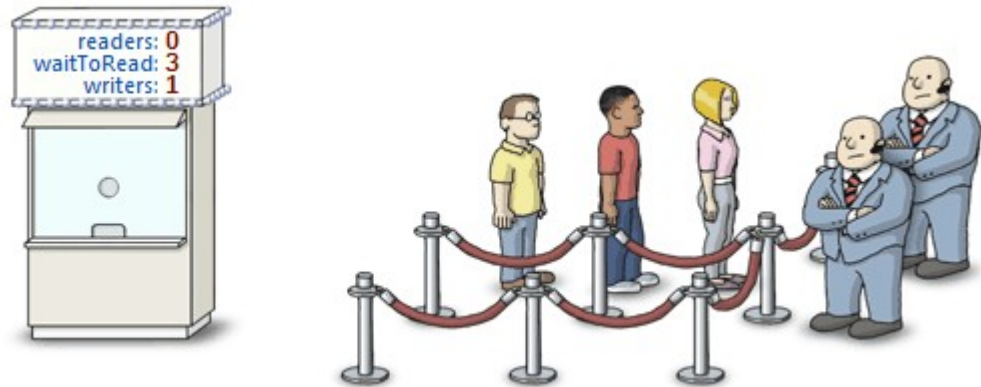
**c) (1 punto)** Describa la impresión total (y su orden) en la salida de la ejecución indicada del programa.

**d) (1 punto)** ¿Cuál será el resultado del mandato “`ls -li`” (listar el contenido del directorio en formato largo con los números de índices de cada archivo) ejecutado (en el directorio donde está nuestro programa) después de ejecutar el programa de manera indicada?

**Pregunta 3 (5 puntos – 30 min.)** El problema clásico de los lectores-escritores.

El archivo de su respuesta debe estar en la carpeta **Buzón 3** del **Examen 1** en PAIDEIA **antes de las 10:15**. Por cada 3 minutos de retardo son -2 puntos.

El nombre de su archivo debe ser `<su_código_de_8_dígitos>_13.txt`. Por ejemplo, `20202912_13.txt`.



<https://preshing.com/20150316/semaphores-are-surprisingly-versatile/>

**(a) (1,5 puntos)** Dada, por orden de llegada, la siguiente secuencia de pedidos de acceso a una base de datos:

W1(0), W2(2), R1(3), R2(4), W3(5), R3(5), R4(6), R5(7), W4(8), R6(9),

donde  $W_i(t)$  indica al  $i$ -ésimo escritor,  $R_i(t)$  al  $i$ -ésimo lector, y  $t$  indica el momento de llegada del pedido.

Considerando que la atención a un escritor exige 5 unidades de tiempo, mientras que la atención a un lector exige 3 unidades de tiempo, ¿cuál será la secuencia de atención si la preferencia se da

- (1) a los lectores,
- (2) a los escritores, o
- (3) si la atención es equitativa por orden de llegada?

Considere que la ejecución de varios procesos al mismo tiempo no afecta a la duración de la atención requerida individualmente. Indique el momento de tiempo del inicio de atención de cada pedido.

Analice el siguiente algoritmo:

```
semaphore mutex = 1;
semaphore r = 0, w = 0;
int nr = 0, nw = 0, dr = 0, dw = 0;

void reader(void)
{
    down(&mutex);
    if (nw == 0) { nr++; up(&r); }
    else dr++;
    up(&mutex);
    down(&r);
    read_data_base();
    down(&mutex);
    nr--;
    if (nr == 0 && dw > 0) { dw--; nw++; up(&w); }
    up(&mutex);
    use_data();
}
```

```

void writer(void)
{
    prepare_data();
    down(&mutex);
    if (nr == 0 && nw == 0) { nw++; up(&w); }
    else dw++;
    up(&mutex);
    down(&w);
    write_data_base();
    down(&mutex);
    nw--;
    if (dw > 0) { dw--; nw++; up(&w); }
    else while (dr > 0) { dr--; nr++; up(&r); }
    up(&mutex);
}

```

**(b) (2 puntos)** Explique el papel de cada uno de los semáforos y de las variables indicando sus valores posibles. ¿Existe alguna preferencia en la atención de pedidos en este algoritmo?

**(c) (1 punto)** Considerando la secuencia de pedidos dada en el apartado **(a)** y la preferencia (o su ausencia) en atención de los pedidos del algoritmo presentado en el apartado **(b)**, ¿cuál será estado de cada proceso de los lectores y de los escritores para el **tiempo 11**? Si el estado es bloqueado, indique qué evento exactamente se espera.

**d) (0,5 puntos)** Considerando la secuencia de pedidos dada en el apartado **(a)** y la preferencia (o su ausencia) en atención de los pedidos del algoritmo presentado en el apartado **(b)**, ¿cuál es el valor de cada variable (las enteras simples y los semáforos) para el **tiempo 14**? No olvide considerar al semáforo como una estructura.

#### **Pregunta 4 (5 puntos – 30 min.)**

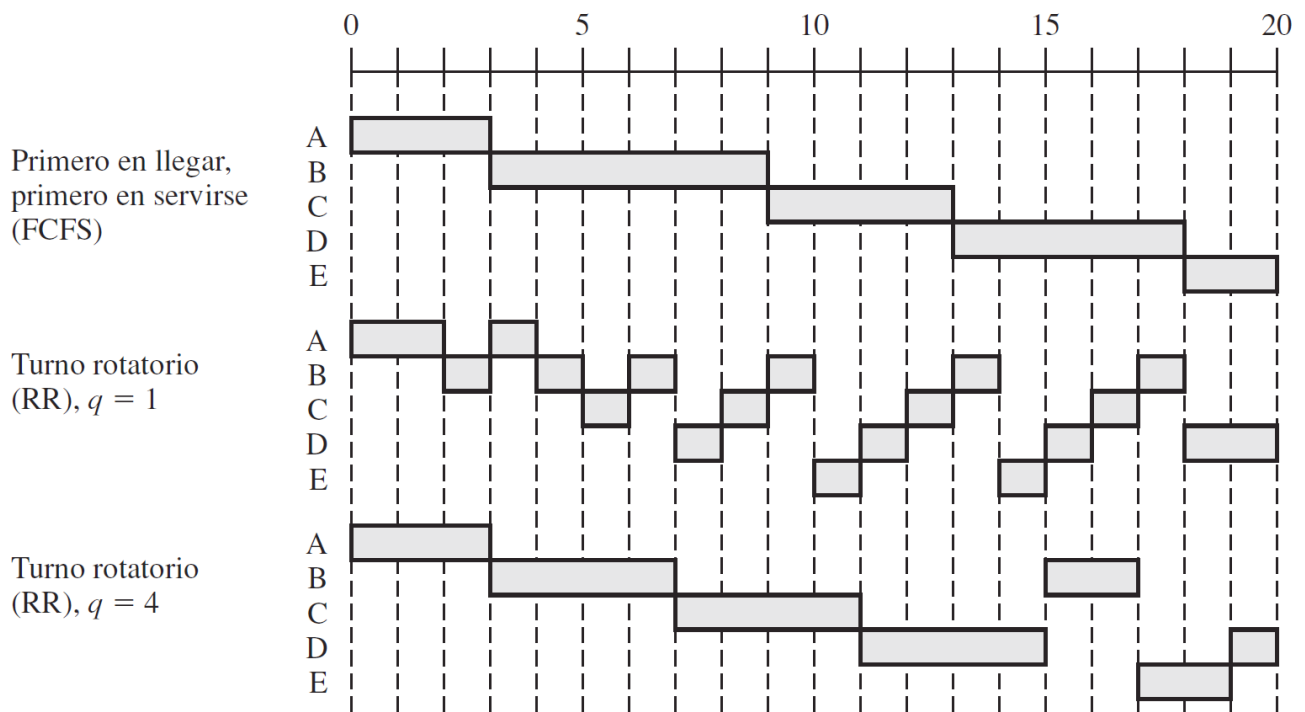
El archivo de su respuesta debe estar en la carpeta **Buzón 4** del **Examen 1** en PAIDEIA **antes de las 11:00 hrs.**

Por cada 3 minutos de retardo son -2 puntos.

El nombre de su archivo debe ser *<su\_código\_de\_8\_dígitos>\_14.txt*. Por ejemplo, 20202912\_14.txt.

Para el ejemplo de los 5 procesos dados en la siguiente tabla, y la gráfica Gantt de su ejecución según FCFS, y RR  $q=1$ , RR  $q=4$ :

Proceso	Tiempo de llegada	Tiempo de servicio
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2



Se tienen los siguientes resultados en la siguiente tabla, siendo “tiempo de estancia” el *turnaround time*, es decir, el tiempo de retorno que considera los tiempos de espera más el tiempo de servicio.

Proceso		A	B	C	D	E	Media
Tiempo de llegada		0	2	4	6	8	
Tiempo de servicio ( $T_s$ )		3	6	4	5	2	
FCFS	Tiempo de finalización	3	9	13	18	20	8.60 2.56
	Tiempo de estancia ( $T_e$ )	3	7	9	12	12	
	$T_e/T_s$	1.00	1.17	2.25	2.40	6.00	
RR $q = 1$	Tiempo de finalización	4	18	17	20	15	10.80 2.71
	Tiempo de estancia ( $T_e$ )	4	16	13	14	7	
	$T_e/T_s$	1.33	2.67	3.25	2.80	3.50	
RR $q = 4$	Tiempo de finalización						
	Tiempo de estancia ( $T_e$ )						
	$T_e/T_s$						

**a) (2 puntos)** Complete los cálculos en la tabla, para los recuadros en blanco, de los procesos A, B, C, D, E, justificando los mismos con los valores requeridos.

**b) (2 puntos)** Complete los cálculos para los valores medios, incluyendo el tiempo promedio de finalización, incluido para FCFS y RR  $q=1$ , justificando los mismos con los valores requeridos.

**c) (1 punto)** ¿Qué puede calcularse en cuanto a los cambios de contexto para RR  $q=1$  y RR  $q=4$ ? Si cada cambio de contexto toma 0.1 unidades de tiempo (10% de  $q=1$ ), ¿cuánto tiempo total significa en cada caso? Considere el cambio de contexto inicial en el instante 0.



Preparado por FS (1,4) y VK (2,3)  
con LibreOffice Writer en Linux Mint 20 “Ulyana”

Profesores del curso: (0781) V. Khlebnikov  
(0782) F. Solari A.

Lima, 28 de octubre de 2020