

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

SISTEMAS OPERATIVOS

3ra práctica (tipo a)
(Primer semestre de 2021)

Horario 0781: prof. V. Khlebnikov

Duración: 1 h. 50 min.

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

Puntaje total: 20 puntos

Pregunta 1 (6 puntos – 30 min.) Su respuesta debe estar en la carpeta **INF239_0781_P3_P1_Buzón** de la **Práctica 3** en PAIDEIA **antes de las 09:40**. Por cada 3 minutos de retardo son -2 puntos.

El nombre de su archivo debe ser `<su_código_de_8_dígitos>_31.txt`. Por ejemplo, `20171903_31.txt`.

En memoria particionada el sistema operativo ocupa siempre una zona de la memoria, mientras que el resto está disponible para los procesos que van llegando al sistema. La zona de memoria que está a disposición de los procesos se puede dividir en un número fijo o variable de particiones. Cuando el número de particiones es fijo, hablamos de un sistema de gestión de memoria con número fijo de particiones o MFT (Multiprogramación con número Fijo de Tareas) que está presentado en las diapositivas 43 y 44 de la clase 9 del nuestro curso.

En este caso la zona de memoria reservada para los procesos se divide en un número fijo de particiones; el tamaño de cada partición es invariable una vez que se ha establecido. El número de particiones y el tamaño de cada una se define en el momento de arrancar la máquina. El resultado es un sistema multiprogramado con número fijo de tareas. Las particiones pueden ser todas del mismo tamaño o de distinto tamaño.

El algoritmo de ubicación debe decidir en qué partición albergar al nuevo proceso. La ubicación de un proceso en memoria es trivial si todas las particiones tienen el mismo tamaño. Es indiferente en qué partición se ubique un proceso mientras haya particiones libres. Si no hay particiones libres, los procesos que deban cargarse en memoria tendrán que esperar a que se libere una partición.

Cuando las particiones de memoria son de distinto tamaño se pueden aplicar varios algoritmos de ubicación. Una política de ubicación es asignar la primera partición disponible en la que quepa el proceso. Otra política de ubicación consiste en asignar al proceso la partición libre más pequeña en la que quepa. En el primer caso, se utilizará una única cola. En el segundo caso, se empleará una cola para cada partición. Si existen varias particiones de igual tamaño, se asigna una cola a todas las particiones de igual tamaño.

El particionamiento de la memoria en zonas estáticas de igual tamaño plantea dos problemas: (1) se produce fragmentación interna, (2) un programa puede ser demasiado grande y no caber en ninguna de las particiones.

Tenemos un sistema con gestión de memoria MFT compuesto por cuatro particiones:

Base de la partición	Tamaño de la partición	Estado
0 K	150 K	Asignada al S.O.
150 K	100 K	Libre
250 K	150 K	Libre
400 K	100 K	Libre

El tamaño, el instante de llegada y la duración de los procesos que llegan al sistema:

Proceso	Instante llegada	Duración	Tamaño
A	0	3	130 K
B	1	3	90 K
C	2	2	80 K
D	3	2	70 K
E	5	4	110 K

Se pide determinar cómo se realiza la asignación de particiones y el tipo y tamaño de la fragmentación que se produce para los siguientes casos:

a) (3 puntos – 15 min.) Se emplea una cola de procesos por partición. El algoritmo de ubicación asigna al proceso la partición más pequeña en la que quepa. Los tiempos de carga y desalojo en/de memoria se consideran despreciables.

b) (3 puntos – 15 min.) Se emplea una cola única de procesos. El algoritmo de ubicación asigna la primera partición disponible en la que quepa. Los tiempos de carga y desalojo en/de memoria se consideran despreciables.



La práctica ha sido preparada por VK
con LibreOffice Writer en Linux Mint 20.1 “Ulyssa”

Profesor del curso: V. Khlebnikov

Lima, 18 de junio de 2021

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

SISTEMAS OPERATIVOS

3ra práctica (tipo a)
(Primer semestre de 2021)

Horario 0781: prof. V. Khlebnikov

Duración: 1 h. 50 min.

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

Puntaje total: 20 puntos

Pregunta 2 (6 puntos – 30 min.) Su respuesta debe estar en la carpeta **INF239_0781_P3_P2_Buzón** de la **Práctica 3** en PAIDEIA **antes de las 10:20**. Por cada 3 minutos de retardo son -2 puntos.

El nombre de su archivo debe ser `<su_código_de_8_dígitos>_32.txt`. Por ejemplo, 20171903_32.txt.

a) (3 puntos – 15 min.) Nuestro sistema operativo realiza una gestión de memoria virtual. Tenemos un proceso cuyo espacio lógico de direcciones está formado por 100 páginas. Cada una de estas páginas tiene un tamaño de 2 K. También sabemos que el tamaño de la memoria física es de 1 M. Queremos conocer de cuántos bits se componen tanto la dirección física o real como la lógica o virtual. También queremos conocer el tamaño de la tabla de páginas de este proceso.

b) (3 puntos – 15 min.) Queremos traducir las direcciones virtuales, generadas por un proceso, a direcciones reales en un sistema que soporta memoria virtual paginada. Cada página tiene un tamaño de 2 K y la tabla de páginas del proceso es la siguiente (se indica solamente el campo del número de marco):

	#marco
4:	8
3:	10
2:	-
1:	4
0:	7

Las direcciones virtuales que queremos traducir son:

1. 0x1873
2. 0x2033
3. 0x0889



La práctica ha sido preparada por VK
con LibreOffice Writer en Linux Mint 20.1 “Ulyssa”

Profesor del curso: V. Khlebnikov

Lima, 18 de junio de 2021

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

SISTEMAS OPERATIVOS

3ra práctica (tipo a)
(Primer semestre de 2021)

Horario 0781: prof. V. Khlebnikov

Duración: 1 h. 50 min.

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

Puntaje total: 20 puntos

Pregunta 3 (8 puntos – 30 min.) Su respuesta debe estar en la carpeta **INF239_0781_P3_P3_Buzón** de la **Práctica 3** en PAIDEIA **antes de las 11:00**. Por cada 3 minutos de retardo son -2 puntos.

El nombre de su archivo debe ser `<su_código_de_8_dígitos>_33.txt`. Por ejemplo, `20171903_33.txt`.

a) (4 puntos) Nuestro sistema soporta memoria virtual paginada. Ha llegado un proceso que al cargarse necesita 50 K de memoria. El tamaño de la página es de 4 K y el de la memoria de 200 K. El mapa de bits tiene la siguiente información:

```

1 0 1 0 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1
1 1 1 0 0 1 0 0 0 0
0 1 0 1 0 1 0 1 0 0
1 0 0 0 0 1 0 0 0 0

```

Un bit a 1 significa que el marco correspondiente está ocupado y un bit a 0 significa que está libre. Debemos calcular:

a1) (2 puntos) cuál será el contenido de la tabla de páginas una vez que el proceso se ha cargado.

a2) (2 puntos) el estado del mapa de bits después de cargado el proceso.

b) (2 puntos) Suppose that a machine has 48-bit virtual addresses and 32-bit physical addresses. If pages are 4 KB, how many entries are in the page table if it has only a single level? Explain. ¿In the inverted page table? Presente los valores finales con el prefijo adecuado: K, M, ...

c) (2 puntos) Suppose that a machine has 38-bit virtual addresses and 32-bit physical addresses. What is the main advantage of a multilevel page table over a single-level one? **(1 punto)** With a two-level page table, 16-KB pages, and 4-byte entries, how many bits should be allocated for the top-level page table field and how many for the next-level page table field? Explain. **(1 punto más)**



La práctica ha sido preparada por VK
con LibreOffice Writer en Linux Mint 20.1 “Ulyssa”

Profesor del curso: V. Khlebnikov

Lima, 18 de junio de 2021