PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

SISTEMAS OPERATIVOS
3ra práctica (tipo a)
(Primer semestre de 2014)

Horario 0781: prof. V. Khlebnikov

Duración: 1 h. 50 min.
Nota: No se puede usar ningún material de consulta.
La presentación, la ortografía y la gramática influirán en la calificación.
Puntaje total: 20 puntos

Pregunta 1 (2 puntos – 10 min.) (*Fred Kuhns*) Given memory partitions of 100K, 500K, 200K, 300K, and 600K (in order), how would each of the First-fit, Best-fit, and Worst-fit algorithms place processes of 212K, 417K, 112K, and 426K (in order)? Which algorithm makes the most efficient use of memory?

Pregunta 2 (2 puntos – 10 min.) (*Fred Kuhns*) Consider a logical address space of eight pages of 1024 addressable words each, mapped onto a physical memory of 32 frames. How many bits are there in the logical address? How many bits are there in the physical address?

Pregunta 3 (2 puntos – 10 min.) (*Fred Kuhns*) Consider a paging system with the page table (1 level) stored in memory.

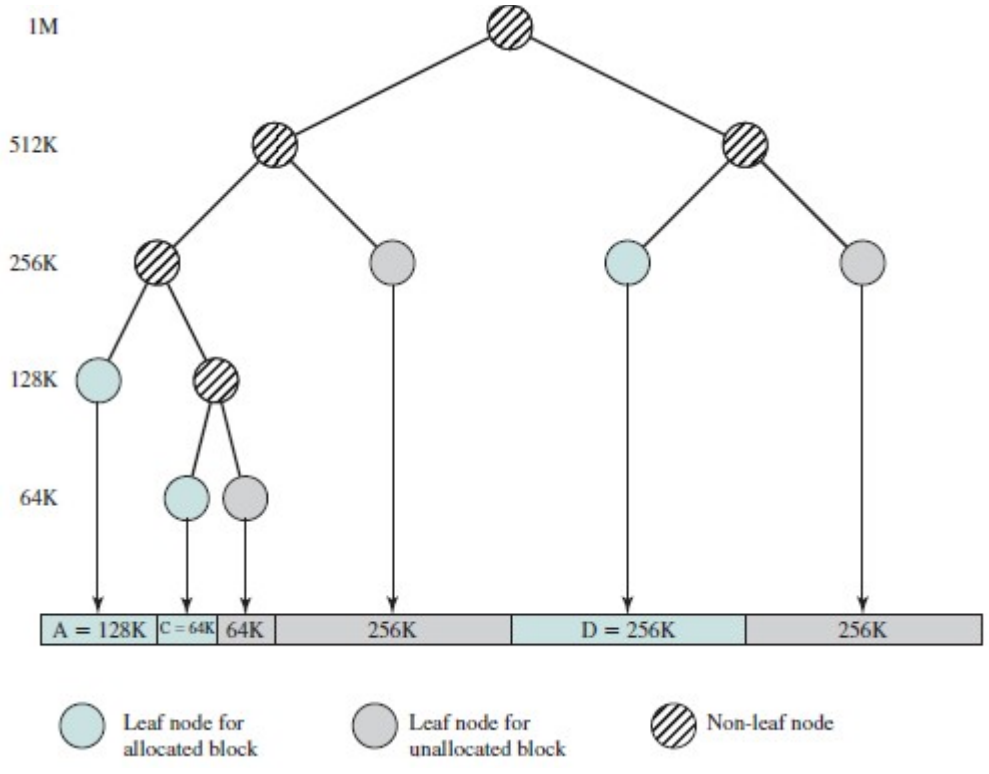
a. If a memory reference takes 200 nanoseconds, how long does a paged memory reference take?
b. If we add associative registers, and 75 percent of all page-table references are found in the associative registers, what is the effective memory reference time? (*Assume that finding a page-table entry in the associative registers takes zero time, if the entry is there.*)

Pregunta 4 (4 puntos – 20 min.) (*Fred Kuhns*) For a demand-paged memory subsystem with the page table held in registers, memory access times are as follows:

- *8 msec* to service a page fault if an empty page is available or the replaced page is not modified,
- *20 msec* if the replaced page is modified,
- *100 nsec* memory access time,
- 70% of the time the page to be replaced is modified.

What is the maximum acceptable page fault rate (P) for an *effective access time* of no more than 200 nsec?

Pregunta 5 (3 puntos – 15 min.) (*William Stallings – OSIDP/7E*) Para todos los bloques presentados en la figura, indique las direcciones de sus primer y el último bytes en hexadecimal.



Pregunta 6 **(4 puntos – 20 min.)** (*Buddy memory allocation*) Un bloque de memoria, usado por el *buddy system*, tiene la dirección `0xb0000 (1011 0000 0000 0000 0000)`. En este sistema no se asignan los bloques menores que 8K y mayores que 1G.

a) (2 puntos) ¿De que tamaño puede ser este bloque?

b) (1 punto) Si este es el bloque de tamaño máximo posible, ¿cuál es la dirección de su *buddy*? Explique por qué.

c) (1 punto) El artículo en la Wikipedia dice (con referencia a Donald Knuth), que "*... The "buddy" of each block can be found with an exclusive OR of the block's address and the block's size.*" Presente este cálculo para el bloque dado.

Pregunta 7 **(1 punto – 5 min.)** (*A.S.Tanenbaum*) A computer with a 32-bit address uses a two-level page table. Virtual addresses are split into a 9-bit top-level page table field, an 11-bit second-level page table field, and an offset. How large are the pages and how many are there in the address space?

Pregunta 8 **(2 puntos – 10 min.)** (*A.S.Tanenbaum*) A computer has 32-bit virtual addresses and 4-KB pages. The program and data together fit in the lowest page (0-4095). The stack fits in the highest page. How many entries are needed in the page table if traditional (one-level) paging is used? How many page table entries are needed for two-level paging, with 10 bits in each part?

La práctica ha sido preparada por VK
en Linux Mint 16 con LibreOffice Writer.

Profesor del curso: (0781) V. Khlebnikov

Pando, 27 de mayo de 2014