Branch : CSE & IT Batch : English

Operating Systems

Memory Management

DPP 06

[MCQ]

1. A computer system has a physical address of 128 bits and a page size of 32 KB. Each page table entry contains 4 valid and 2 dirty bits along with the translation bits. If the maximum size of the page table of a process is 60MB. Calculate the size of logical address space supported by the system (in GB)?

(a) 512 (b) 32
(c) 128 (d) 64

[NAT]

2. Consider a system with 512 KB page size and each page table entry requires 8 bytes. The level of paging required to map a 30bit logical address if every page table fits into a single page are_____.

[NAT]

3. A computer system implements a 36 bits virtual address. Page size to 64KB and the size of physical memory is 34 bits. The approximate size of page table in the system is_____MB. [Assume memory is byte addressable]

[MCQ]

4. If page size is 4096 Bytes, in a paging system. A process needs 20 frames. What is the maximum possible internal fragmentation size_____.

(a) 4096 byts
(b) 4095 Bytes
(c) 2048 Bytes
(d) No internal fragmentation

[MCQ]

5. Consider a virtual address space of eight pages of 2048 words each mapped onto a physical memory of 32 frames. How many bits are there in the logical address.

(a) 13 (b) 14
(c) 10 (d) 11

[NAT]

6. Consider a virtual memory system with physical memory of 8GB. A page size of 8KB and 46-bit virtual address. Assume every page table exactly fits into a single page. If page table entry size is 4 bytes the how many levels of page tables would be required?

[NAT]

7. Consider the following statements:

I. Overlays are used to increase the size of physical memory.
II. The size of virtual memory depends on the size of main memory.
III. Aging is used to keep track of number of times a page is referenced.

How many of the above are correct statements?

[NAT]

8. A processor can support a maximum memory of 8 GB, where the memory is word addressable (each word consists 4 bytes). The size of address bus of the process is at least _____bits.

[MSQ]

9. Which of the following statement is/are correct regarding paging?

(a) Paging helps solve the issue external fragmentation
(b) Page size has no impact on internal fragmentation
(c) Paging incurs memory overheads.
(d) Multi-level paging is necessary to support pages of different sizes.

[MCQ]

10. What is basic objective of hierarchical paging?

(a) Reduce Internal fragmentation.
(b) Reduce External fragmentation.
(c) To reduce context -switch overhead.
(d) Reduce page table size overhead in memory.

Answer Key

1. **(c)**
2. **(1)**
3. **(3 to 3)**
4. **(b)**
5. **(b)**
6. **(3)**
7. **(0)**
8. **(31)**
9. **(a, c)**
10. **(d)**

Hints & Solutions

1. (c)

Given:

Page size = 32 KB

Physical address = 128 bits

$$\text{Number of frames} = \frac{\text{Physcial address sapce}}{\text{Page size}}$$

$$= \frac{2^{128}}{2^{15}} = 2^{113}$$

Frame bits = $\log_2 (2^{113}) = 113$ bits

Size of page table entry = Frame bits + Extra bits (Valid + Dirty)

= 113 + (4 + 2)

= 113 + 6

= 119 bits

Size of PTE = 15 bytes.

Now, Page table is of size = 60 MB.

$$\text{Page table entries} = \frac{\text{Page table size}}{\text{PTE size}} = \frac{60\text{MB}}{15\text{B}} = 4\text{ M}$$

Hence, Logical address space = No. of page table entry × page size

= 4 M × 32 KB

= 128 GB

Therefore, option 'C' is correct.

2. (1)

Given,

Page size = 512 KB

Page table entry size = 8 byte

First level page table size = (Number of pages in page table) × (PTE size)

$$= \frac{2^{30}\text{B}}{2^{19}\text{B}} \times 2^{8}\text{B}$$

$= 2^{19}$ B

= 512 KB

$\because$ 1st level page table size =Page size.

So, 1st level page table easily fits into a single page.

3. (3 to 3)

- Virtual address = 36 bits

  $\therefore$ Virtual address space = 2^{36} bytes

- Physical address = 34 bits

  $\therefore$ Physical address space = 2^{34} Bytes

- Number of frames = $\frac{2^{34}}{2^{16}} = 2^{18}$ Bytes

  Page table entry size = $\log_2 (2^{18}) = 18$ bits

$\therefore$ Page table size = (Number of pages in page table) × (PTE size)

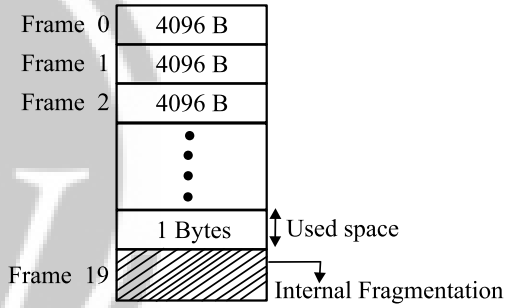
$$= \frac{2^{36}}{2^{16}} \times 18 \text{ bits}$$

$= 2^{20} \times 3$ bits

= 3 MB

4. (B)

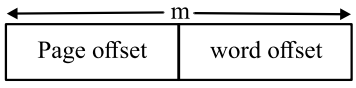
- When memory is divided into fixed sized partition then internal fragmentation may occur.
- Page size is equal to frame size so, in worst case process may store data in all allocated frame completely but still left with 1 byte of data.



$\therefore$ Maximum possible Internal fragmentation = 4096 –1 = 4095 bytes.

5. (b)

Virtual address space:



- Page size × word size = 2048 words

So, word offset = $\log_2 2048 = 11$ bits

- VAS = Number of page × page size
  = 8 × 2048
  = 2^{14} words

$\therefore$ Virtual Address = $\log_2 2^{14} = 14$ bits

6. (3)

Each page size = 8KB = 2^{13} bytes

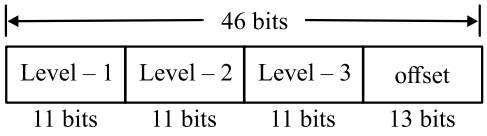
Page table entry size = 4 bytes

If page table exactly fits into a page, page table size = Page size

So, Number of page table entries

$$= \frac{\text{Page size}}{\text{Page table entry size}} = \frac{2^{13}}{4} = 2^{11}$$

So,



Therefore, there level are required.

7. (0)

Overlays are not used to increase the size of main memory. Size of virtual memory does not depends on size of main memory.

Aging is used to solve problem of saturation.

So, no statement is correct.

8. (31)

Size of memory = Number of words × Number of bits per word.

$\Rightarrow 2^{33}$ B = Number of words (address)× 4B

Number of words = $\frac{2^{33}\text{B}}{2^{2}\text{B}} = 2^{31}$ B.

So, 31 bits are required.

9. (A, C)

Page are divided into fixed size slots, so no external fragmentation, but application smaller than page size can cause internal fragmentation.

Page tables requires extra pages in memory. Therefore it is overhead to memory.

Therefore, (a) and (c) are correct.

10. (d)

Hierarchical paging or inverted paging is primarily designed to associate smaller page table with processes. In inverted paging, only one page table will be maintained for all the process avoiding the overhead of maintaining the page table for every process and reducing page table size overhead in memory.