1. Each entry in a translation lookaside buffer (TLB) consists of _____

a) key

b) value

c) bit value

d) constant

2. If a page number is not found in the TLB, then it is known as a _____

a) TLB miss

b) Buffer miss

c) TLB hit

d) All of the mentioned

3. An _____ uniquely identifies processes and is used to provide address space protection for that process.

a) address space locator

b) address space identifier

c) address process identifier

d) none of the mentioned

4. The percentage of times a page number is found in the TLB is known as _____

a) miss ratio

b) hit ratio

c) miss percent

d) none of the mentioned

5. Memory protection in a paged environment is accomplished by _____

a) protection algorithm with each page

b) restricted access rights to users

c) restriction on page visibility

d) protection bit with each page

6. When the valid – invalid bit is set to valid, it means that the associated page _____

a) is in the TLB

b) has data in it

c) is in the process's logical address space

d) is the system's physical address space

7. Illegal addresses are trapped using the _____ bit.

a) error

b) protection

c) valid – invalid

d) access

8. When there is a large logical address space, the best way of paging would be _____

a) not to page

b) a two level paging algorithm

c) the page table itself

d) all of the mentioned

10. The size of the pages in the paging scheme is
a) variable
b) fixed
c) both variable and fixed
d) none

11. Physical memory is broken into fixed-sized blocks called _____

a) frames

b) pages

c) backing store

d) none of the mentioned

View Answer

12. Logical memory is broken into blocks of the same size called _____

a) frames

b) pages

c) backing store

d) none of the mentioned

View Answer

13. Every address generated by the CPU is divided into two parts. They are _____

a) frame bit & page number

b) page number & page offset

c) page offset & frame bit

d) frame offset & page offset

View Answer

14. The _____ is used as an index into the page table.

a) frame bit

b) page number

c) page offset

d) frame offset

View Answer

15. The _____ table contains the base address of each page in physical memory.

a) process

b) memory

c) page

d) frame

View Answer

16. The size of a page is typically _____

a) varied

b) power of 2

c) power of 4

d) none of the mentioned

View Answer

17. If the size of logical address space is 2 to the power of m, and a page size is 2 to the power of n addressing units, then the high order _____ bits of a logical address designate the page number, and the ____ low order bits designate the page offset.

a) m, n

b) n, m

c) m – n, m

d) m – n, n

View Answer

18. With paging there is no _____ fragmentation.

a) internal

b) external

c) either type of

d) none of the mentioned

View Answer

19. The operating system maintains a _____ table that keeps track of how many frames have been allocated, how many are there, and how many are available.

a) page

b) mapping

c) frame

d) memory

View Answer

20. Paging increases the _____ time.

a) waiting

b) execution

c) context – switch

d) all of the mentioned

View Answer

21. Smaller page tables are implemented as a set of _____

a) queues

b) stacks

c) counters

d) registers

View Answer

22. The page table registers should be built with _____

a) very low speed logic

b) very high speed logic

c) a large memory space

d) none of the mentioned

23. For larger page tables, they are kept in main memory and a _____ points to the page table.

a) page table base register

b) page table base pointer

c) page table register pointer

d) page table base

24. For every process there is a _____

a) page table

b) copy of page table

c) pointer to page table

d) all of the mentioned

25. In segmentation, each address is specified by _____

a) a segment number & offset

b) an offset & value

c) a value & segment number

d) a key & value

26. In paging the user provides only _____ which is partitioned by the hardware into _____ and _____

a) one address, page number, offset

b) one offset, page number, address

c) page number, offset, address

d) none of the mentioned

27. Each entry in a segment table has a _____

a) segment base

b) segment peak

c) segment value

d) none of the mentioned

28. The segment base contains the _____

a) starting logical address of the process

b) starting physical address of the segment in memory

c) segment length

d) none of the mentioned

29. The segment limit contains the _____

a) starting logical address of the process

b) starting physical address of the segment in memory

c) segment length

d) none of the mentioned

30. The offset ‘d’ of the logical address must be _____

a) greater than segment limit

b) between 0 and segment limit

c) between 0 and the segment number

d) greater than the segment number

31. If the offset is legal _____

a) it is used as a physical memory address itself

b) it is subtracted from the segment base to produce the physical memory address

c) it is added to the segment base to produce the physical memory address

d) none of the mentioned

32. When the entries in the segment tables of two different processes point to the same physical location _____

a) the segments are invalid

b) the processes get blocked

c) segments are shared

d) all of the mentioned

33. If there are 32 segments, each of size 1Kb, then the logical address should have _____

a) 13 bits

b) 14 bits

c) 15 bits

d) 16 bits

34. **Consider a set of n tasks with known runtimes r1, r2, … rn to be run on a uniprocessor machine. Which of the following processor scheduling algorithms will result in the maximum throughput? (GATE 2001)**
(a) Round-Robin
(b) Shortest-Job-First
(c) Highest-Response-Ratio-Next
(d) First-Come-First-Served

35-**Where does the swap space reside ? (GATE 2001)**
(a) RAM
(b) Disk
(c) ROM
(d) On-chip cache
36-Which of the following scheduling algorithms is non-preemptive? (GATE CS 2002)

a) Round Robin
b) First-In First-Out
c) Multilevel Queue Scheduling
d) Multilevel Queue Scheduling with Feedback

37-**Consider the following statements with respect to user-level threads and kernel supported threads**
i. context switch is faster with kernel-supported threads
ii. for user-level threads, a system call can block the entire process
iii. Kernel supported threads can be scheduled independently
iv. User level threads are transparent to the kernel
Which of the above statements are true? (GATE CS 2004)
a) (ii), (iii) and (iv) only
b) (ii) and (iii) only
c) (i) and (iii) only
d) (i) and (ii) only

38- A smaller page size leads to smaller page tables? T/F

39- A smaller page size leads to more TLB misses ? ? T/F

40- A blocking kernel-scheduled thread blocks all threads in the process? ? T/F

41- Threads are cheaper to context switch than processes? T/F

42- Different user-level threads of the same process can have different scheduling priorities? T/F

43-in case of TLB miss we totally ignore accessing TLB ? T/F

44-TLB always contain all page table entries? T/F

45-page table is kept in secondary memory ? T/F

46-changing page table requires changing only PTBR register ? T/F

47- Segmentation avoids external memory fragmentation ? T/F

48 – paging avoids external memory fragmentation ? T/F

49- Interrupts can be shut off by user processes.? T/F

50- The kernel uses the Process Control Block to keep track of bookkeeping information about processes, such as the program counter.? T/F

51- Virtual memory space is always smaller than physical memory space? T/F

52- If an OS designer does not like page size he/she just picks a new one? T/F

53- A system that uses segmentation and paging just means that the pages are grouped into logical statements? T/F

54- The medium term scheduler only runs in times of high resource contention, as when physical memory is full? T/F

55- Using a larger page size makes page tables larger? T/F

56-there is internal fragmentation in paging ? T/F

57- there is internal fragmentation in segmentation ? T/F

58-STBR contains the starting physical address of segments? T/F

59-STLR specifies the length of the segment ? T/F

60-its easier to place segments in memory than pages ? T/F

61-pages is fixed size? T/F

62-there can be more than one logical address space to be mapped to one address space in paging? T/F

63- there can be more than one logical address space to be mapped to one address space in segments? T/F

64-we don’t need a placement question in paging ? T/F

65-only one process can exist in one segment? T/F

66- only one process can exist in one page ? T/F

67-in paging number of frames used in physical memory indicate degree of multiprogramming ? T/F

68-paging with segmentation means paged segments ? T/F

69-in paging with segmentation the segment-table entry contains the base address of the segment? T/F

70- in segmentation the segment-table entry contains the base address of the segment? T/F

Consider a paging hardware with a TLB. Assume that the entire page table and all the pages are in the physical memory. It takes 10 milliseconds to search the TLB and 80 milliseconds to access the physical memory. If the TLB hit ratio is 0.6, the effective memory access time (in milliseconds) is _____.

A-120

B-122

C-124

D-118