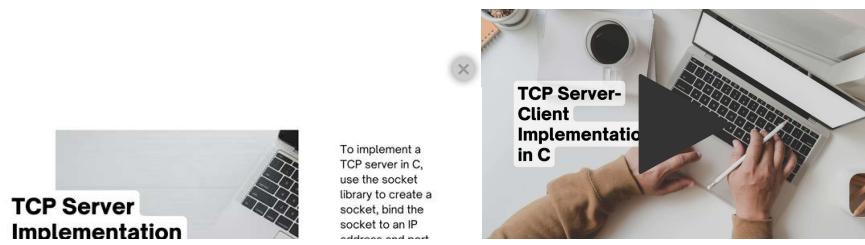


Multilevel Paging | Paging | Practice Problems





TCP Server-Client implementation in C | TCP (Transmission Co...

Operating System

Multilevel Paging in OS-

Before you go through this article, make sure that you have gone through the previous article on [Multilevel Paging](#).

We have discussed-

- Multilevel Paging is a paging scheme where there exists a hierarchy of [Page Tables](#).
- Multilevel paging is needed when a page table can not be stored in a single frame due to its large size.

In this article, we will discuss practice problems based on multilevel paging.

Also Read- [Paging Important Formulas](#)

PRACTICE PROBLEMS BASED ON MULTILEVEL PAGING-

Problem 01:

Consider a system using multilevel paging scheme. The page size is 1 MB. The memory is byte addressable and virtual address is 64 bits long. The page table entry size is 4 bytes.

Find-

1. How many levels of page table will be required?
2. Give the divided physical address and virtual address.

Solution-

■ ■ ■

Given-

- Virtual Address = 64 bits
- Page size = 1 MB
- Page table entry size = 4 bytes

Number of Bits in Frame Number-

Page table entry size

= 4 bytes

= 32 bits

Thus, Number of bits in frame number = 32 bits

Number of Frames in Main Memory:-

We have, Number of bits in frame number = 32 bits

Thus,

Number of frames in main memory

= 2^{32} frames

Size of Main Memory:-

Size of main memory

= Total number of frames x Frame size

= $2^{32} \times 1 \text{ MB}$

$$= 2^{52} \text{ B}$$

Thus, Number of bits in physical address = 52 bits

Number of Bits in Page Offset-

We have,

Page size

$$= 1 \text{ MB}$$

$$= 2^{20} \text{ B}$$

Thus, Number of bits in page offset = 20 bits

Alternatively,

Number of bits in page offset

$$= \text{Number of bits in physical address} - \text{Number of bits in frame number}$$

$$= 52 \text{ bits} - 32 \text{ bits}$$

$$= 20 \text{ bits}$$

Process Size-

Number of bits in virtual address = 64 bits

Thus,

Number of Pages of Process-

Number of pages the process is divided

$$= \text{Process size} / \text{Page size}$$

$$= 2^{64} \text{ B} / 1 \text{ MB}$$

$$= 2^{64} \text{ B} / 2^{20} \text{ B}$$

$$= 2^{44} \text{ pages}$$

Inner Page Table Size-

Inner page table keeps track of the frames storing the pages of process.

Inner page table size

$$= \text{Number of entries in inner page table} \times \text{Page table entry size}$$

$$= \text{Number of pages the process is divided} \times \text{Page table entry size}$$

$$= 2^{44} \times 4 \text{ bytes}$$

$$= 2^{46} \text{ bytes}$$

Now, we can observe-

- The size of inner page table is greater than the frame size (1 MB).
- Thus, inner page table can not be stored in a single frame.
- So, inner page table has to be divided into pages.

Number of Pages of Inner Page Table-

$$= 2^{46} \text{ B} / 1 \text{ MB}$$

$$= 2^{46} \text{ B} / 2^{20} \text{ B}$$

$$= 2^{26} \text{ pages}$$

Now, these 2^{26} pages of inner page table are stored in different frames of the main memory.

Number of Page Table Entries in One Page of Inner Page Table-

Number of page table entries in one page of inner page table

$$= \text{Page size} / \text{Page table entry size}$$

$$= 1 \text{ MB} / 4 \text{ B}$$

$$= 2^{20} \text{ B} / 2^2 \text{ B}$$

$$= 2^{18} \text{ entries}$$

Number of Bits Required to Search an Entry in One Page of Inner Page Table-

One page of inner page table contains 2^{18} entries.

Thus,

Number of bits required to search a particular entry in one page of inner page table = 18 bits

■ ■ ■

Outer Page Table-1 Size-

Outer page table-1 is required to keep track of the frames storing the pages of inner page table.

Outer page table-1 size

= Number of entries in outer page table-1 x Page table entry size

= Number of pages the inner page table is divided x Page table entry size

= 2^{26} x 4 bytes

= 2^{28} bytes

= 256 MB

Now, we can observe-

- The size of outer page table-1 is greater than the frame size (1 MB).
- Thus, outer page table-1 can not be stored in a single frame.
- So, outer page table-1 has to be divided into pages.

Number of Pages of Outer Page Table-1

Number of pages the outer page table-1 is divided

= Outer page table-1 size / Page size

= 256 MB / 1 MB

= 256 pages

Now, these 256 pages of outer page table-1 are stored in different frames of the main memory.

Number of Page Table Entries in One Page of Outer Page Table-1

Number of page table entries in one page of outer page table-1

= Page size / Page table entry size

= 1 MB / 4 B

= 2^{20} B / 2^2 B

$= 2^{18}$ entries

Number of Bits Required to Search an Entry in One Page of Outer Page Table-1

One page of outer page table-1 contains 2^{18} entries.

Thus,

Number of bits required to search a particular entry in one page of outer page table-1 = 18 bits

Outer Page Table-2 Size-

Outer page table-2 is required to keep track of the frames storing the pages of outer page table-1.

Outer page table-2 size

$= \text{Number of entries in outer page table-2} \times \text{Page table entry size}$

$= \text{Number of pages the outer page table-1 is divided} \times \text{Page table entry size}$

$= 256 \times 4 \text{ bytes}$

$= 1 \text{ KB}$

Now, we can observe-

- The size of outer page table-2 is less than the frame size (16 KB).

- Page Table Base Register (PTBR) will store the base address of the outer page table-2.

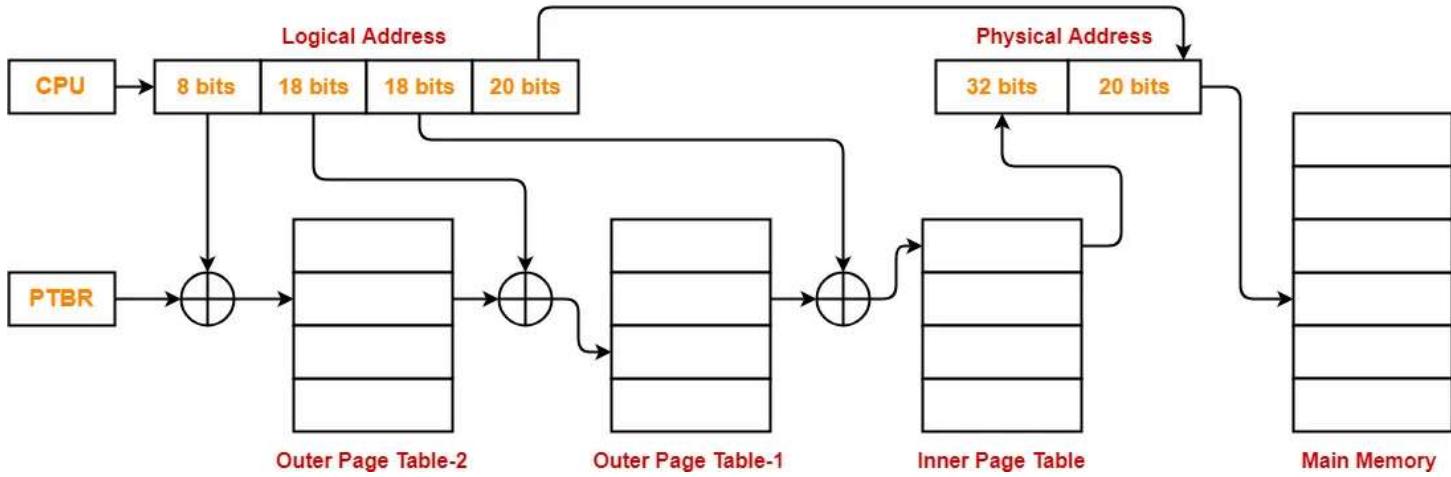
Number of Bits Required to Search an Entry in Outer Page Table-2

Outer page table-2 contains $256 = 2^8$ entries.

Thus,

Number of bits required to search a particular entry in outer page table-2 = 8 bits

The paging system will look like as shown below-



Problem-02:

Consider a system using multilevel paging scheme. The page size is 1 GB. The memory is byte addressable and virtual address is 72 bits long. The page table entry size is 4 bytes.

Find-

1. How many levels of page table will be required?
2. Give the divided physical address and virtual address.

Given-

- Virtual Address = 72 bits
- Page size = 1 GB
- Page table entry size = 4 bytes

Number of Bits in Frame Number-

We have,

Page table entry size

= 4 bytes

= 32 bits

Thus, Number of bits in frame number = 32 bits

Number of Frames in Main Memory-

We have, Number of bits in frame number = 32 bits

Thus,

Number of frames in main memory

= 2^{32} frames

Size of Main Memory-

Size of main memory

= Total number of frames x Frame size

= $2^{32} \times 1 \text{ GB}$

Number of Bits in Page Offset-

We have,

Page size

= 1 GB

= 2^{30} B

Thus, Number of bits in page offset = 30 bits

Alternatively,

Number of bits in page offset

= Number of bits in physical address – Number of bits in frame number

= 62 bits – 32 bits

= 30 bits

Process Size-

Number of bits in virtual address = 72 bits

Thus,

Process size

= 2^{72} bytes

Number of Pages of Process-

Number of pages the process is divided

= Process size / Page size

$$= 2^{42} \text{ pages}$$

Inner Page Table Size-

Inner page table keeps track of the frames storing the pages of process.

Inner page table size

$$= \text{Number of entries in inner page table} \times \text{Page table entry size}$$

$$= \text{Number of pages the process is divided} \times \text{Page table entry size}$$

$$= 2^{42} \times 4 \text{ bytes}$$

$$= 2^{44} \text{ bytes}$$

Now, we can observe-

- The size of inner page table is greater than the frame size (1 GB).
- Thus, inner page table can not be stored in a single frame.
- So, inner page table has to be divided into pages.

Number of Pages of Inner Page Table-

Number of pages the inner page table is divided

$$= \text{Inner page table size} / \text{Page size}$$

$$= 2^{44} \text{ B} / 1 \text{ GB}$$

$$= 2^{44} \text{ B} / 2^{30} \text{ B}$$

$$= 2^{14} \text{ pages}$$

Now, these 2^{14} pages of inner page table are stored in different frames of the main memory.

Number of page table entries in one page of inner page table

$$= \text{Page size} / \text{Page table entry size}$$

$$= 1 \text{ GB} / 4 \text{ B}$$

$$= 2^{30} \text{ B} / 2^2 \text{ B}$$

$$= 2^{28} \text{ entries}$$

Number of Bits Required to Search an Entry in One Page of Inner Page Table-

One page of inner page table contains 2^{28} entries.

Thus,

Number of bits required to search a particular entry in one page of inner page table = 28 bits

Outer Page Table Size-

Outer page table is required to keep track of the frames storing the pages of inner page table.

Outer page table size

$$= \text{Number of entries in outer page table} \times \text{Page table entry size}$$

$$= \text{Number of pages the inner page table is divided} \times \text{Page table entry size}$$

$$= 2^{14} \times 4 \text{ bytes}$$

$$= 2^{16} \text{ bytes}$$

$$= 64 \text{ KB}$$

Now, we can observe-

- The size of outer page table is less than the frame size (1 GB).

- So, for given system, we will have two levels of page table.
- Page Table Base Register (PTBR) will store the base address of the outer page table.

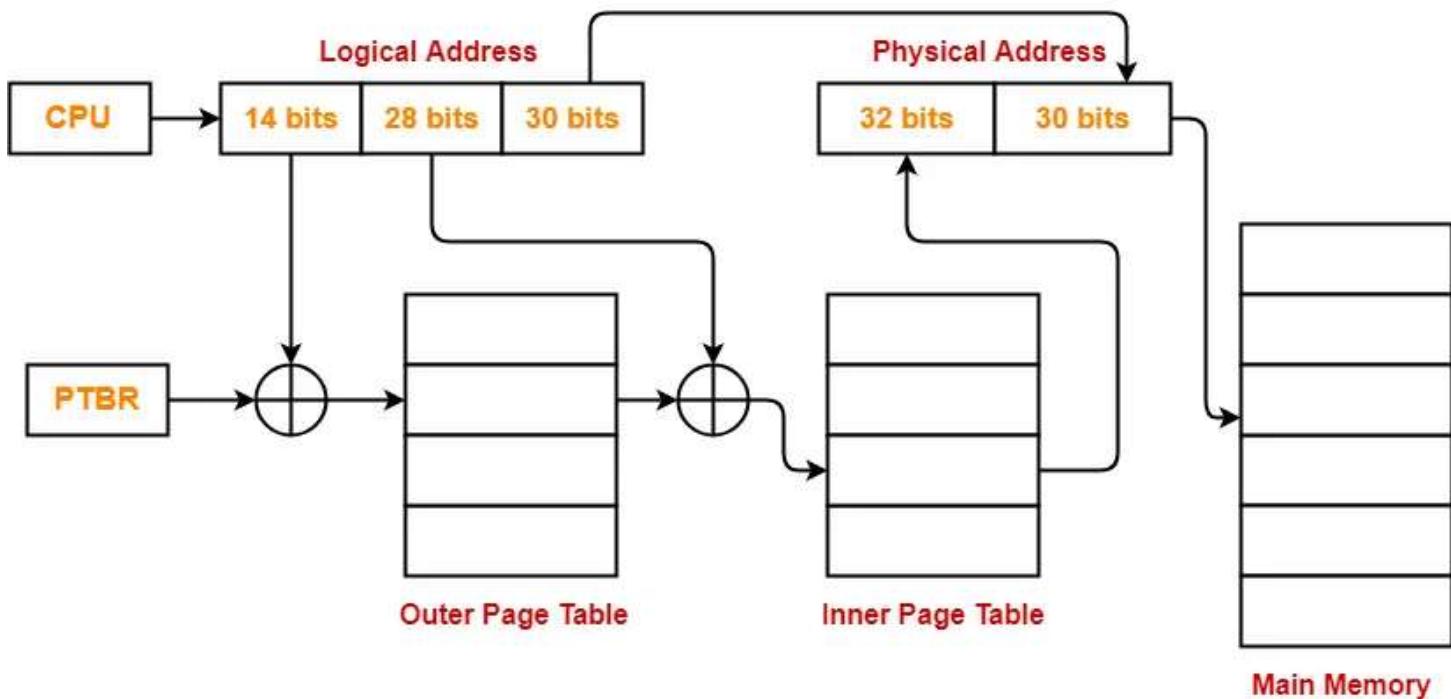
Number of Bits Required to Search an Entry in Outer Page Table-

Outer page table contains 2^{14} entries.

Thus,

Number of bits required to search a particular entry in outer page table = 14 bits

The paging system will look like as shown below-



Problem-03:

Consider a system using multilevel paging scheme. The page size is 256 MB. The memory is byte addressable and virtual address is 72 bits long. The page table entry size is 4 bytes.

2. Give the divided physical address and virtual address.

Solution-

Given-

- Virtual Address = 72 bits
- Page size = 256 MB
- Page table entry size = 4 bytes

Number of Bits in Frame Number-

We have,

Page table entry size

= 4 bytes

= 32 bits

Thus, Number of bits in frame number = 32 bits

Number of Frames in Main Memory-

We have, Number of bits in frame number = 32 bits

Thus,

Number of frames in main memory

= 2^{32} frames

Size of Main Memory-

$$= 2^{32} \times 256 \text{ MB}$$

$$= 2^{60} \text{ B}$$

Thus, Number of bits in physical address = 60 bits

Number of Bits in Page Offset-

We have,

Page size

$$= 256 \text{ MB}$$

$$= 2^{28} \text{ B}$$

Thus, Number of bits in page offset = 28 bits

Alternatively,

Number of bits in page offset

$$= \text{Number of bits in physical address} - \text{Number of bits in frame number}$$

Ad removed. [Details](#)

= 60 bits – 32 bits

= 28 bits

Process Size-

Number of bits in virtual address = 72 bits

Thus,

Process size

= 2^{72} bytes

Number of Pages of Process-

Number of pages the process is divided

= Process size / Page size

= 2^{72} B / 256 MB

= 2^{72} B / 2^{28} B

= 2^{44} pages

Inner Page Table Size-

Inner page table keeps track of the frames storing the pages of process.

Inner page table size

= Number of entries in inner page table x Page table entry size

= Number of pages the process is divided x Page table entry size

= 2^{44} x 4 bytes

Now, we can observe-

- The size of inner page table is greater than the frame size (256 MB).
- Thus, inner page table can not be stored in a single frame.
- So, inner page table has to be divided into pages.

Number of Pages of Inner Page Table-

Number of pages the inner page table is divided

$$= \text{Inner page table size} / \text{Page size}$$

$$= 2^{46} \text{ B} / 256 \text{ MB}$$

$$= 2^{46} \text{ B} / 2^{28} \text{ B}$$

$$= 2^{18} \text{ pages}$$

Now, these 2^{18} pages of inner page table are stored in different frames of the main memory.

Number of Page Table Entries in One Page of Inner Page Table-

Number of page table entries in one page of inner page table

$$= \text{Page size} / \text{Page table entry size}$$

$$= 256 \text{ MB} / 4 \text{ B}$$

$$= 2^{28} \text{ B} / 2^2 \text{ B}$$

$$= 2^{26} \text{ entries}$$

Number of Bits Required to Search an Entry in One Page of Inner Page Table-

Number of bits required to search a particular entry in one page of inner page table = 26 bits

Outer Page Table Size-

Outer page table is required to keep track of the frames storing the pages of inner page table.

Outer page table size

= Number of entries in outer page table x Page table entry size

= Number of pages the inner page table is divided x Page table entry size

= 2^{18} x 4 bytes

= 2^{20} bytes

= 1 MB

Now, we can observe-

- The size of outer page table is less than the frame size (256 MB).
- Thus, outer page table can be stored in a single frame.
- In fact, outer page table will not completely occupy one frame and some space will remain vacant.
- So, for given system, we will have two levels of page table.
- Page Table Base Register (PTBR) will store the base address of the outer page table.

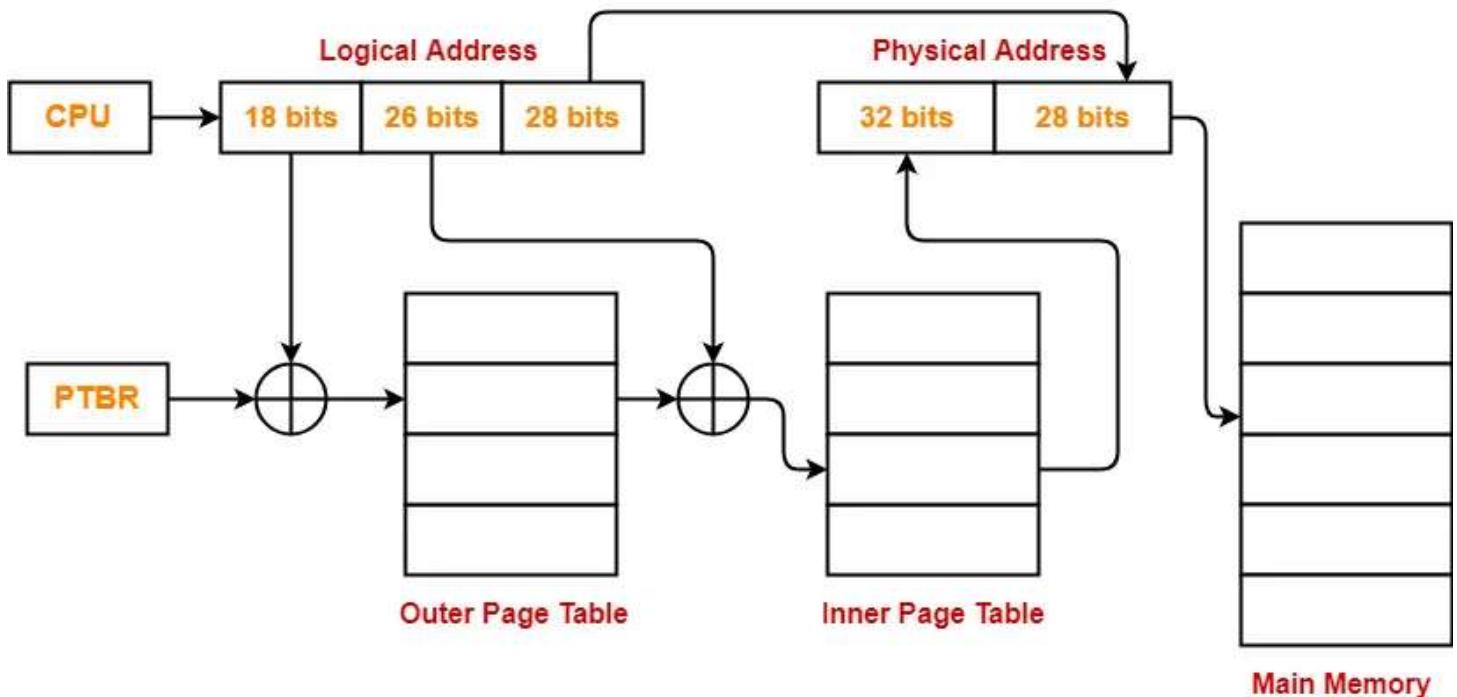
Number of Bits Required to Search an Entry in Outer Page Table-

Outer page table contains 2^{18} entries.

Thus,

Number of bits required to search a particular entry in outer page table = 18 bits

The paging system will look like as shown below-



Problem-04:

Consider a system using multilevel paging scheme. The page size is 16 MB. The memory is byte

1. How many levels of page table will be required?
2. Give the divided physical address and virtual address.

Solution-

Given-

- Virtual Address = 72 bits
- Page size = 16 MB
- Page table entry size = 4 bytes

Number of Bits in Frame Number-

We have,

Page table entry size

= 4 bytes

= 32 bits

Thus, Number of bits in frame number = 32 bits

Number of Frames in Main Memory-

We have, Number of bits in frame number = 32 bits

Thus,

Number of frames in main memory

= 2^{32} frames

Size of Main Memory-

= Total number of frames x Frame size

$$= 2^{32} \times 16 \text{ MB}$$

$$= 2^{56} \text{ B}$$

Thus, Number of bits in physical address = 56 bits

Number of Bits in Page Offset-

We have,

Page size

$$= 16 \text{ MB}$$

$$= 2^{24} \text{ B}$$

Thus, Number of bits in page offset = 24 bits

Alternatively,

Number of bits in page offset

$$= \text{Number of bits in physical address} - \text{Number of bits in frame number}$$

$$= 56 \text{ bits} - 32 \text{ bits}$$

$$= 24 \text{ bits}$$

Process Size-

Number of bits in virtual address = 72 bits

Thus,

Process size

$$= 2^{72} \text{ bytes}$$

Number of pages the process is divided

= Process size / Page size

= 2^{72} B / 16 MB

= 2^{72} B / 2^{24} B

= 2^{48} pages

Inner Page Table Size-

Inner page table keeps track of the frames storing the pages of process.

Inner page table size

= Number of entries in inner page table x Page table entry size

= Number of pages the process is divided x Page table entry size

= $2^{48} \times 4$ bytes

= 2^{50} bytes

Now, we can observe-

- The size of inner page table is greater than the frame size (16 MB).
- Thus, inner page table can not be stored in a single frame.
- So, inner page table has to be divided into pages.

Number of Pages of Inner Page Table-

Number of pages the inner page table is divided

= Inner page table size / Page size

= 2^{26} pages

Now, these 2^{26} pages of inner page table are stored in different frames of the main memory.

Number of Page Table Entries in One Page of Inner Page Table-

Number of page table entries in one page of inner page table

= Page size / Page table entry size

= 16 MB / 4 B

= 2^{24} B / 2^2 B

= 2^{22} entries

Number of Bits Required to Search an Entry in One Page of Inner Page Table-

One page of inner page table contains 2^{22} entries.

Thus,

Number of bits required to search a particular entry in one page of inner page table = 22 bits

Outer Page Table-1 Size-

Outer page table-1 is required to keep track of the frames storing the pages of inner page table.

Ad removed. [Details](#)

Outer page table-1 size

$$\begin{aligned} &= \text{Number of entries in outer page table-1} \times \text{Page table entry size} \\ &= \text{Number of pages the inner page table is divided} \times \text{Page table entry size} \\ &= 2^{26} \times 4 \text{ bytes} \\ &= 2^{28} \text{ bytes} \\ &= 256 \text{ MB} \end{aligned}$$

Now, we can observe-

- The size of outer page table-1 is greater than the frame size (16 MB).
- Thus, outer page table-1 can not be stored in a single frame.
- So, outer page table-1 has to be divided into pages.

Number of Pages of Outer Page Table-1

Number of pages the outer page table-1 is divided

$$\begin{aligned} &= \text{Outer page table-1 size} / \text{Page size} \\ &= 256 \text{ MB} / 16 \text{ MB} \\ &= 16 \text{ pages} \end{aligned}$$

Number of Page Table Entries in One Page of Outer Page Table-1

Number of page table entries in one page of outer page table-1

$$= \text{Page size} / \text{Page table entry size}$$

$$= 16 \text{ MB} / 4 \text{ B}$$

$$= 2^{24} \text{ B} / 2^2 \text{ B}$$

$$= 2^{22} \text{ entries}$$

Number of Bits Required to Search an Entry in One Page of Outer Page Table-1

One page of outer page table-1 contains 2^{22} entries.

Thus,

Number of bits required to search a particular entry in one page of outer page table-1 = 22 bits

Outer Page Table-2 Size-

Outer page table-2 is required to keep track of the frames storing the pages of outer page table-1.

Outer page table-2 size

$$= \text{Number of entries in outer page table-2} \times \text{Page table entry size}$$

$$= \text{Number of pages the outer page table-1 is divided} \times \text{Page table entry size}$$

$$= 16 \times 4 \text{ bytes}$$

$$= 64 \text{ bytes}$$

Now, we can observe-

- The size of outer page table-2 is less than the frame size (16 MB).

- Page Table Base Register (PTBR) will store the base address of the outer page table-2.

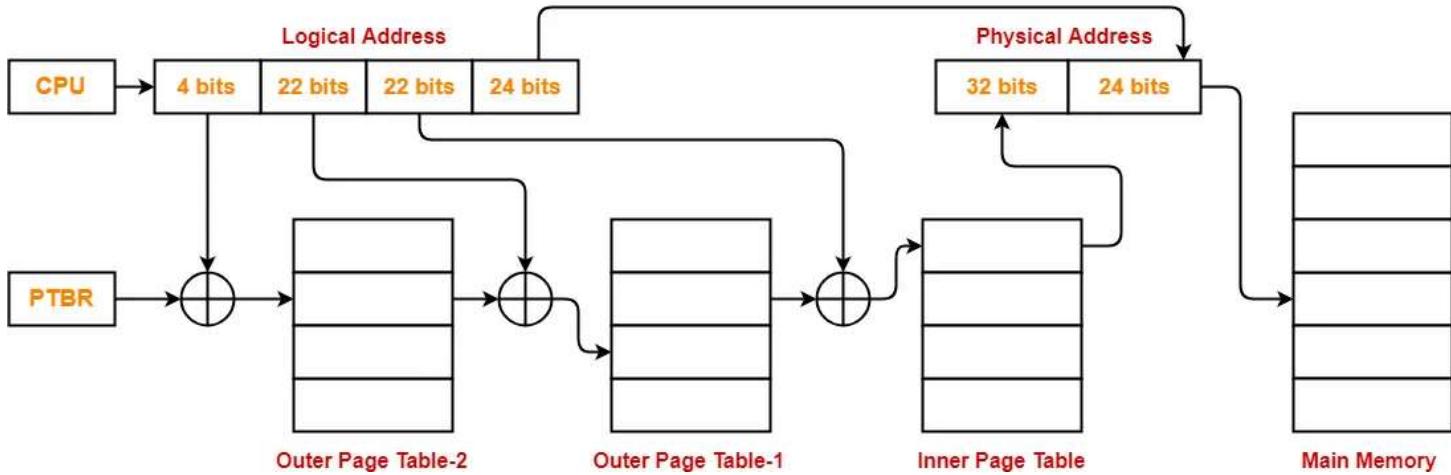
Number of Bits Required to Search an Entry in Outer Page Table-2

Outer page table-2 contains $16 = 2^4$ entries.

Thus,

Number of bits required to search a particular entry in outer page table-2 = 4 bits

The paging system will look like as shown below-



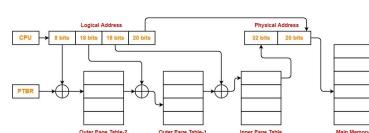
Next Article- [Practice Problems On Multilevel Paging | Set-02](#)

Get more notes and other study material of [Operating System](#).



Watch video lectures by visiting our YouTube channel [LearnVidFun.](#)

Summary



Article Name Multilevel Paging | Paging | Practice Problems

Description Practice Problems based on Multilevel Paging in OS.

Multilevel paging is a paging scheme where there exists a hierarchy of page tables. Multilevel Paging is needed when the size of page table is greater than the frame size.

Author Akshay Singhal

Publisher Name Gate Vidyalay

Publisher Logo



