

Name :- TUSHAR KUMAR  
 Uni. Roll No. :- 2023109  
 Std. ID :- 20051120  
 Date :- 27<sup>th</sup> Aug, 2021

Course :- BSC. IT  
 Sem / Sec :- 2A  
 Sub. Name :- operating system practical  
 sub. code :- PBI-202

Ans 1. :- Algorithm

1. Start
2. Read total no. of block and files.
3. ~~Start~~ from Get the size of each block and the file from the user.
4. Start from the first process and find the maximum block size that can be assigned to the current process, if found then assign it to the current process.
5. If not found then leave that process and move ahead to check the rest of the processes.
6. Display the result.
7. The fragmentation column keeps the track of wasted memory.
8. Stop.

★ Coding

```
#include <stdio.h>
int main()
{
    printf ("In It It Memory Management Scheme - Worst
            Fit");
    int i, j, nblocks, nfiles, temp, top=0;
    int frag[10], blocks[10], files[10];
}
```

~~Just a question~~

Name :- TUSHAR KUMAR

Course :- BSC. IT

Uni. Roll No. :- 2023109

Sem/Sec :- 2A

Std. ID :- 20051120

Sub. Name :- operating system practical

Date :- 27<sup>th</sup> Aug, 2021

Sub. code :- PBI-202

```

static int block_arr[10], file_arr[10];
printf("Enter the Total Number of Blocks : ");
scanf("%d", &nblocks);
printf("Enter the Total Numbers of Files : ");
scanf("%d", &nfiles);
printf("Enter the size of blocks : \n");
for (i=0; i<nblocks; i++)
{
    printf("Block No. %d : ", i+1);
    scanf("%d", &blocks[i]);
}
printf("Enter the size of the files : \n");
for (i=0; i<nfiles; i++)
{
    printf("File No %d : ", i+1);
    scanf("%d", &files[i]);
}
for (i=0; i<nfiles; i++)
{
    for (j=0; j<nblocks; j++)
    {
        if (block_arr[j] != 1)
        {
            temp = blocks[j] - files[i];
            if (temp >= 0)
            {
                if (top < temp)

```



Name :- TUSHAR KUMAR

Course :- BSC. IT

Uni. Roll No. :- 2023109

Sem/Sec :- 2A

Std. ID :- 20051120

Sub. Name :- operating system Practical

Date :- 27<sup>th</sup> Aug, 2021

Sub. Code :- PBI-202

{ file-arr [i] = j ;

top = temp ;

}

}

}

frag[i] = top ;

block-arr [file-arr [i]] = 1 ;

top = 0 ;

}

printf ("\\n File Number \\t File Size \\t Block Number \\t Block  
size \\t Fragment ");

for (i=0; i&lt;nfiles; i++)

{ printf ("\\n %d \\t %d \\t %d \\t %d \\t %d \\t %d ",  
i, files[i], file-arr[i],  
blocks[file-arr[i]], frag[i]);

}

printf ("\\n")

return 0;

}

Tushar Kumar

Enter the number of files:2

Enter the size of the blocks:-

Block 1:5

Block 2:2

Block 3:7

Enter the size of the files :-

File 1:1

File 2:4

File_no:	File_size :	Block_no:	Block_size:	Fragement
1	1	3	7	6
2	4	1	5	1

C:\Users\ASUS\AppData\Local\Temp>c[]

Name :- TUSHAR KUMAR

Uni. Roll No. :- 2023109

Std. ID :- 20051120

Date :- 27<sup>th</sup> Aug, 2021

Course :- BSC. IT

Sem / Sec :- 2A

Sub. Name :- operating system practical

Sub. Code :- PB1-202

Ans 2 :- Coding :-

```

#include <stdio.h>
int absolutevalue(int)

void main()
{
    int queue[25], n, headposition, i, j, k, seek = 0,
        maxrange, difference, temp, queue1[20], queue2[20]
        , temp1 = 0, temp2 = 0;
    float averageseekTime;

    printf("Enter the maximum range of Disk : ");
    scanf("%d", &maxrange);

    printf("Enter the number of queue requests : ");
    scanf("%d", &n);

    printf("Enter the initial head position : ");
    scanf("%d", &headposition);

    printf("Enter the disk positions to be read(queue) : ");
    for(i=1; i<=n; i++)
    {
        scanf("%d", &temp);
        if(temp > headposition)
        {
            queue1[temp1] = temp;
            temp1++;
        }
    }
}

```

Graphic Era Hill University, (Dehradun)

(Answer Sheet for Online Examination Aug, 2021)

Name :- TUSHAR KUMAR

Course :- BSC. IT

Uni. Roll No. :- 2023109

Sem/Sec :- 2A

Std. ID :- 20051120

Sub. Name :- operating system practical.

Date :- 27<sup>th</sup> Aug, 2021

Sub. Code :- PBI-202

else

{ queue2[temp2] = temp;  
temp2++;

}

}

for (i=0; i < temp - 1; i++)

{ for (j=i+1; j < temp1; j++)

{ if (queue1[i] > queue1[j])

{ temp = queue1[i];

queue1[j] = queue1[j];

queue1[j] = temp;

}

}

}

for (i=0; i < temp2 - 1; i++)

{ for (j=i+1; j < temp2; j++)

{ if (queue2[i] < queue2[j])

{ temp = queue2[i];

queue2[i] = queue2[j];

queue2[j] = temp;

}

}

Tushar Kumar

Graphic Era Hill University (Dehradun)  
(Answer Sheet for Online Examination Aug, 2021)

Name :- TUSHAR KUMAR  
Uni. Roll No. :- 2023109  
Stdu. ID :- 20051120  
Date :- 27<sup>th</sup> Aug, 2021

Course :- BSCoIT  
Sem/Sec :- 2A  
Sub. Name :- Operating system practical  
Sub. Code :- PBL-202

for ( $i=1, j=0; j < \text{temp}1; i++, j++$ )

{ queue[i] = queue1[j];

}

queue[i] = max range;

for ( $i = \text{temp}1 + 2, j=0; j < \text{temp}2; i++, j++$ )

{ queue[i] = queue2[j];

}

queue[i] = 0;

queue[0] = head position;

for ( $j=0; j \leq n; j++$ )

{ difference = absolute value (queue[j+1] - queue[j]);

seek = seek + difference

printf ("Disk head move from position %d to %d  
with seek %d\n", queue[j], queue[j+1],  
difference);

3

average seek time = seek / (float) n;

printf ("Total seek Time = %d\n", seek);

printf ("Average seek time = %f\n", average seek time);

Tushar Kumar

Graphic Era Hill University (Dehradun)  
(Answer Sheet for Online Examination Aug, 2021)

Name :- TUSHAR KUMAR

Univ. Roll No. :- 2023109

Std. ID :- 20051120

Date :-

Course :- BSC. IT

Sem/Sec :- 2A

Sub. Name :-

Sub. Code :-

int absoluteValue (int x)

{ if (x > 0)

{ return x;

}

else

{ return x \* -1;

}

}

Tushar  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

```
Enter the maximum range of Disk: 99  
Enter the number of queue requests: 7  
Enter the initial head position: 24  
Enter the disk positions to be read(queue): 12
```

```
26  
24  
4  
42  
8  
50
```

```
Disk head moves from position 24 to 26 with Seek 2  
Disk head moves from position 26 to 42 with Seek 16  
Disk head moves from position 42 to 50 with Seek 8  
Disk head moves from position 50 to 99 with Seek 49  
Disk head moves from position 99 to 24 with Seek 75  
Disk head moves from position 24 to 12 with Seek 12  
Disk head moves from position 12 to 8 with Seek 4  
Disk head moves from position 8 to 4 with Seek 4  
Total Seek Time= 170  
Average Seek Time= 24.285715
```

```
...Program finished with exit code 0  
Press ENTER to exit console.
```