FIRST FIT MEMORY MANAGEMENT

AIM:

TO WRITE A PROFRAM FOR FISRT FIT MEMORY MANAGEMENT USING C LANGUGAE.

The First Fit memory allocation checks the empty memory blocks in a sequential manner.

It means that the memory Block which found empty in the first attempt is checked for size.

But if the size is not less than the required size then it is allocated.

OR

In the First Fit Memory Management Scheme, we check the block in the sequential manner i.e we take the first process and compare its size with the first block.

If the size is less than the size of the first block then only it is allocated.

Otherwise, we move to the second block and this process is continuously going on until all processes are allocated.

ALGORITHM:

1 Step: START.

2 Step: At first get the no of processes and blocks.

3 Step: Allocate the process by **if(size of block>=size of the process)** then allocate the process else move to the next block.

4 Step: Now Display the processes with blocks and allocate to respective process.

5 Step: STOP.

OR

ALGORITHM:

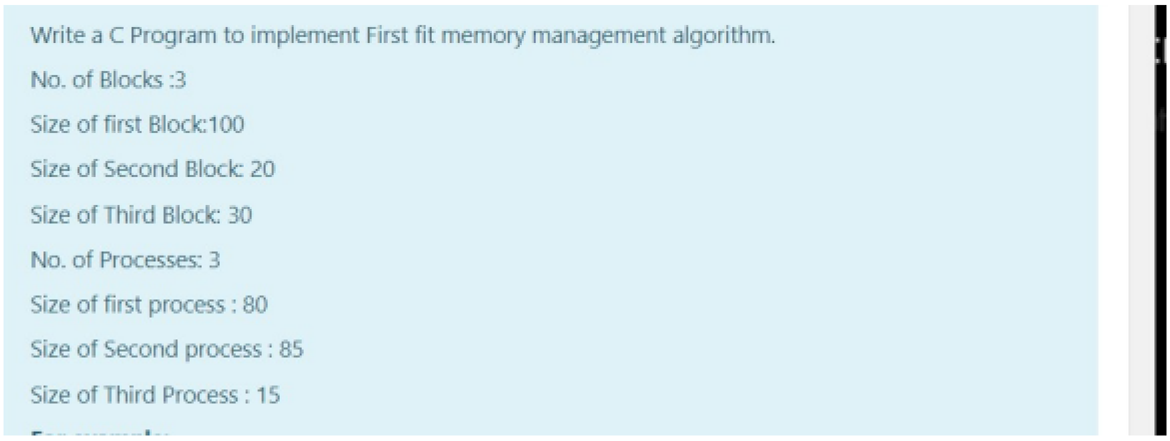
Step 1.Get the number of process and number of blocks.

Step 2.Get the size of each block.

Step 3. Allocate process If (size of the block > = size of the process)
//allocate the process
else
//move on to the next blog

Step 4.Display the process with the blocks allocated to a respective process

Step 5.Stop.



Write a C Program to implement First fit memory management algorithm.

No. of Blocks :3

Size of first Block:100

Size of Second Block: 20

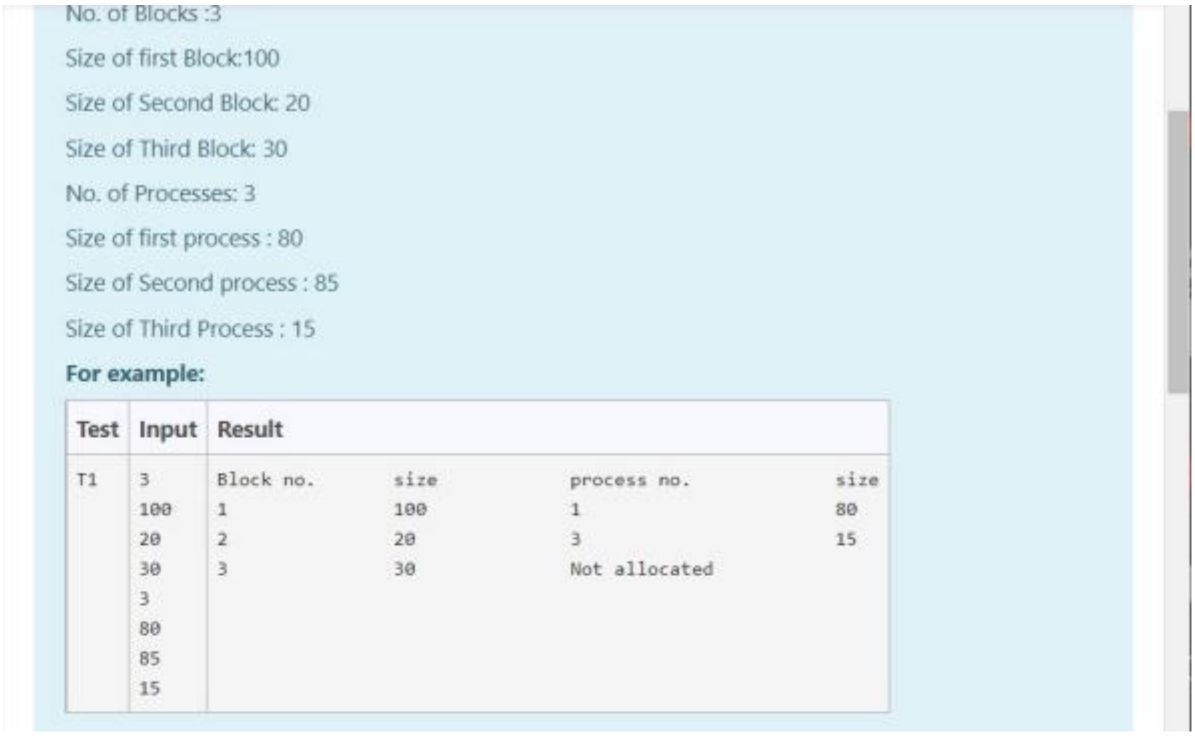
Size of Third Block: 30

No. of Processes: 3

Size of first process : 80

Size of Second process : 85

Size of Third Process : 15



No. of Blocks :3

Size of first Block:100

Size of Second Block: 20

Size of Third Block: 30

No. of Processes: 3

Size of first process : 80

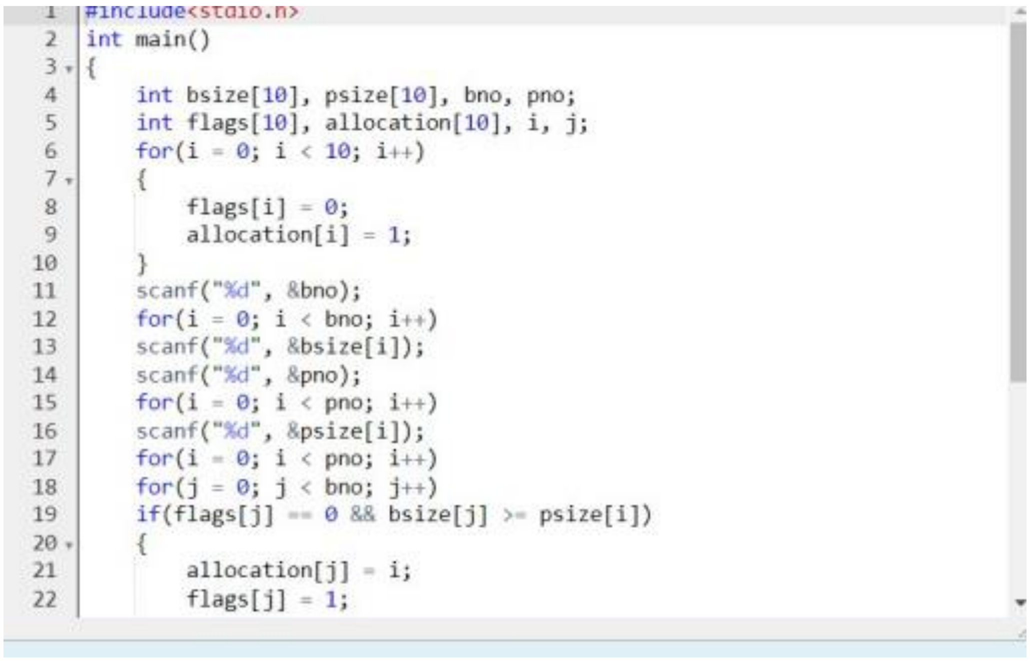
Size of Second process : 85

Size of Third Process : 15

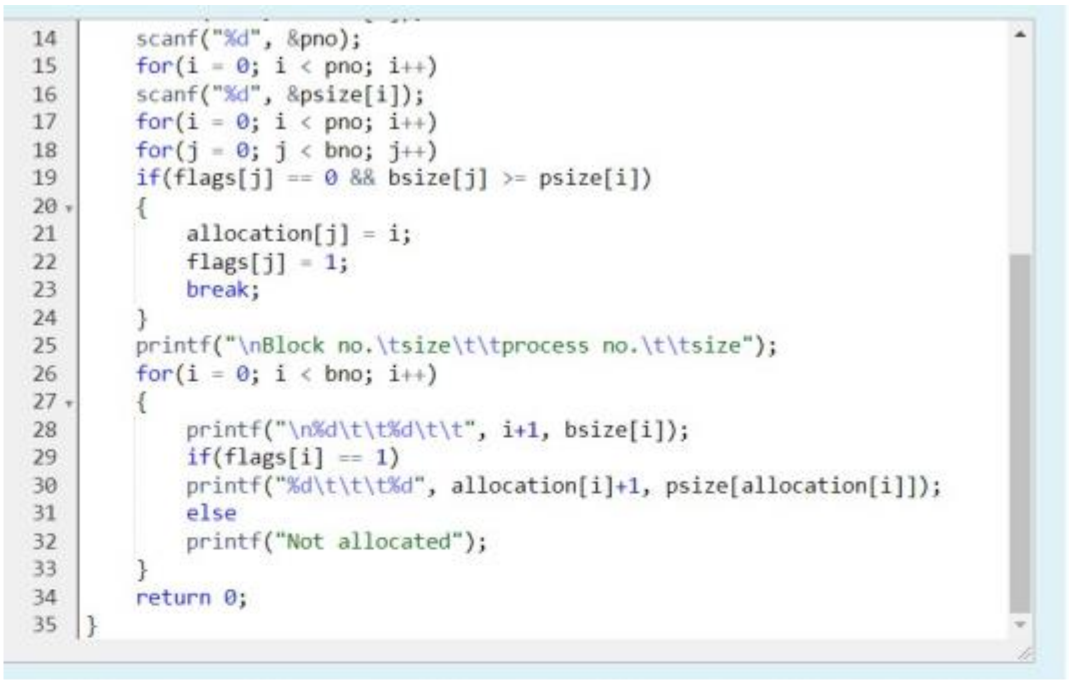
For example:

Test	Input	Result
T1	3 100 20 30 3 80 85 15	Block no. size process no. size 1 100 1 80 2 20 3 15 3 30 Not allocated

PROGRAM:

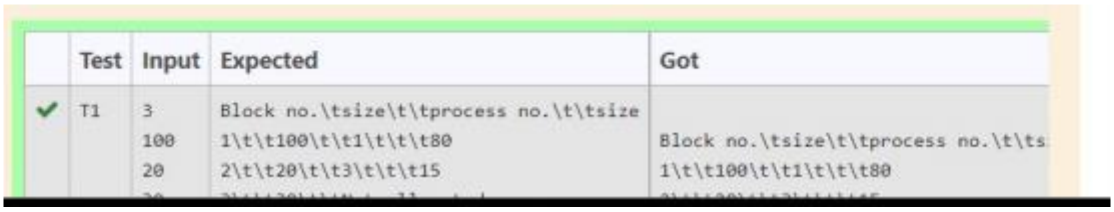


```c
#include<stdio.h>
int main()
{
    int bsize[10], psize[10], bno, pno;
    int flags[10], allocation[10], i, j;
    for(i = 0; i < 10; i++)
    {
        flags[i] = 0;
        allocation[i] = 1;
    }
    scanf("%d", &bno);
    for(i = 0; i < bno; i++)
    scanf("%d", &bsize[i]);
    scanf("%d", &pno);
    for(i = 0; i < pno; i++)
    scanf("%d", &psize[i]);
    for(i = 0; i < pno; i++)
    for(j = 0; j < bno; j++)
    if(flags[j] == 0 && bsize[j] >= psize[i])
    {
        allocation[j] = i;
        flags[j] = 1;
```

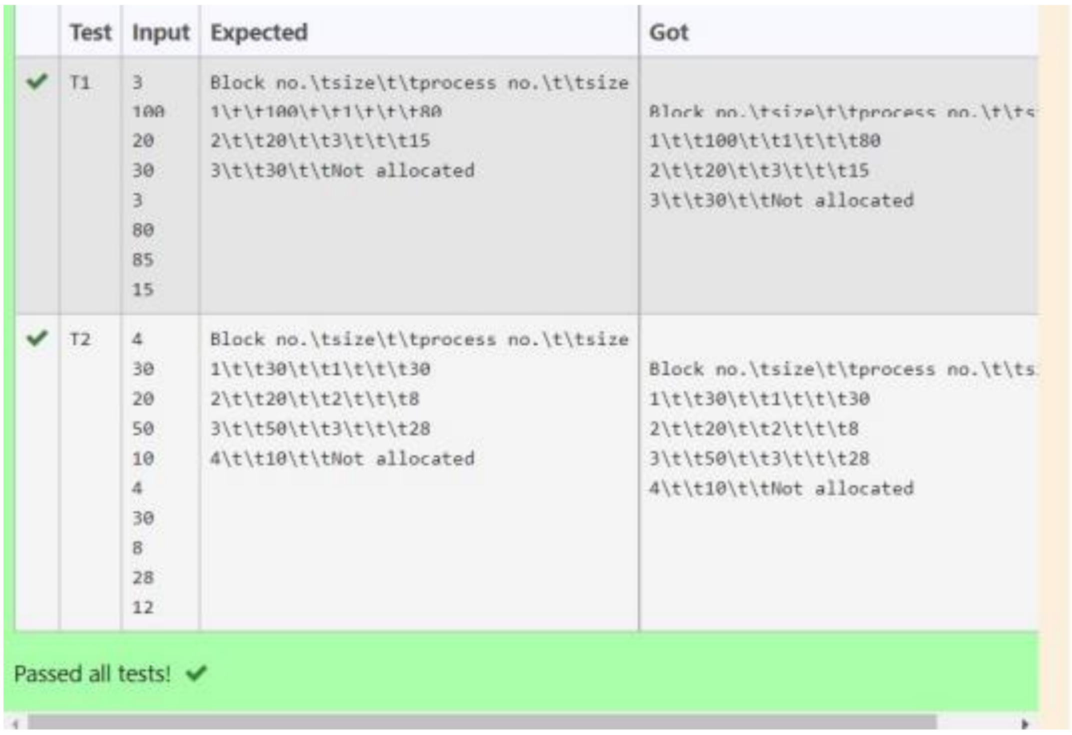


```c
    scanf("%d", &pno);
    for(i = 0; i < pno; i++)
    scanf("%d", &psize[i]);
    for(i = 0; i < pno; i++)
    for(j = 0; j < bno; j++)
    if(flags[j] == 0 && bsize[j] >= psize[i])
    {
        allocation[j] = i;
        flags[j] = 1;
        break;
    }
    printf("\nBlock no.\tsize\t\tprocess no.\t\tsize");
    for(i = 0; i < bno; i++)
    {
        printf("\n%d\t\t%d\t\t", i+1, bsize[i]);
        if(flags[i] == 1)
        printf("%d\t\t\t%d", allocation[i]+1, psize[allocation[i]]);
        else
        printf("Not allocated");
    }
    return 0;
}
```

OUTPUT:



	Test	Input	Expected	Got
✓	T1	3 100 20	Block no.\tsize\t\tprocess no.\t\tsize 1\t\t100\t\t1\t\t\t80 2\t\t20\t\t3\t\t\t15	 Block no.\tsize\t\tprocess no.\t\ts 1\t\t100\t\t1\t\t\t80



	Test	Input	Expected	Got
✓	T1	3 100 20 30 3 80 85 15	Block no.\tsize\t\tprocess no.\t\tsize 1\t\t100\t\t1\t\t\t80 2\t\t20\t\t3\t\t\t15 3\t\t30\t\tNot allocated	Block no.\tsize\t\tprocess no.\t\ts 1\t\t100\t\t1\t\t\t80 2\t\t20\t\t3\t\t\t15 3\t\t30\t\tNot allocated
✓	T2	4 30 20 50 10 4 30 8 28 12	Block no.\tsize\t\tprocess no.\t\tsize 1\t\t30\t\t1\t\t\t30 2\t\t20\t\t2\t\t\t8 3\t\t50\t\t3\t\t\t28 4\t\t10\t\tNot allocated	Block no.\tsize\t\tprocess no.\t\ts 1\t\t30\t\t1\t\t\t30 2\t\t20\t\t2\t\t\t8 3\t\t50\t\t3\t\t\t28 4\t\t10\t\tNot allocated

Passed all tests! ✓

RESULT:

FIRST FIT MEMORY MANAGEMENT WAS SUCCESSFULLY IMPLEMENT USING C LANGUAGE.