AIM:

To write a c program to implement best fit algorithm for memory management.

ALGORITHM:

1. Start the process

2. Declare the size

3. Get the number of processes to be inserted

4. Allocate the first hole that is big enough searching

5. Start at the beginning of the set of holes

6. If not start at the hole that is sharing the pervious first fit search end

7. Compare the hole

8. if large enough then stop searching in the procedure

9. Display the values

10. Stop the process

PROGRAM:

```
#include<stdio.h>
#include<process.h>
void main()
{
        int a[20],p[20],i,j,n,m;
        printf("Enter no of Blocks.\n");
        scanf("%d",&n);
        for(i=0;i<n;i++)
        {
                printf("Enter the %dst Block size:",i);
                scanf("%d",&a[i]);
        }
        printf("Enter no of Process.\n");
        scanf("%d",&m);
        for(i=0;i<m;i++)
        {
                printf("Enter the size of %dst Process:",i);
                scanf("%d",&p[i]);
        }
      for(i=0;i<n;i++)
        {
        for(j=0;j<m;j++)
                {
                        if(p[j]<=a[i])
                        {
                                printf("The Process %d allocated to %d\n",j,a[i]);
                                p[j]=10000;
                                break;
                        }
                }
}
        for(j=0;j<m;j++)
        {
        if(p[j]!=10000)
```

```
                {
        printf("The Process %d is not allocated\n",j);
                }
        }
}
```

OUTPUT:

```
Enter no of Blocks.
5
Enter the 0st Block size:500
Enter the 1st Block size:400
Enter the 2st Block size:300
Enter the 3st Block size:200
Enter the 4st Block size:100
Enter no of Process.
5
Enter the size of 0st Process:100
Enter the size of 1st Process:350
Enter the size of 2st Process:400
Enter the size of 3st Process:150
Enter the size of 4st Process:200
The Process 0 allocated to 500
The Process 1 allocated to 400
The Process 3 allocated to 200
The Process 2 is not allocated
The Process 4 is not allocated
```