Title: Memory Management-I

Experiment No: 08a

Experiment Name: Memory Management using First Fit.

Algorithm:

1. **Get no. of Processes and no. of blocks.**
2. **After that get the size of each block and process requests.**
3. **Now allocate processes**
   if(block size >= process size)
   //allocate the process
   else
   //move on to next block
4. **Display the processes with the blocks that are allocated to a respective process.**
5. **Stop.**

Source Code:

```
#include<stdio.h>

void main()
{
	int bsize[10], psize[10], bno, pno, flags[10], allocation[10], i, j;

	for(i = 0; i < 10; i++)
	{
		flags[i] = 0;
		allocation[i] = -1;
	}

	printf("Enter no. of blocks: ");
	scanf("%d", &bno);

	printf("\nEnter size of each block: ");
	for(i = 0; i < bno; i++)
		scanf("%d", &bsize[i]);

	printf("\nEnter no. of processes: ");
	scanf("%d", &pno);

	printf("\nEnter size of each process: ");
	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");
        }
}
```

Output:

```
File  Edit  View  Search  Terminal  Help
sachin@sachin-VirtualBox:~/Desktop/programs$ cc firstfit.c
sachin@sachin-VirtualBox:~/Desktop/programs$ ./a.out
Enter no. of blocks: 5

Enter size of each block: 100
500
200
300
600

Enter no. of processes: 4

Enter size of each process: 212
417
112
426

Block no.       size            process no.             size
1               100             Not allocated
2               500             1                       212
3               200             3                       112
4               300             Not allocated
5               600             2                       417sachin@sachin-VirtualBox:~/Desktop/programs$
```

Title: Memory Management-I

Experiment No: 8b

Experiment Name: Memory Management using Best Fit.

Algorithm:

1. **Get no. of Processes and no. of blocks.**
2. **After that get the size of each block and process requests.**
3. **Then select the best memory block that can be allocated using the above definition.**
4. **Display the processes with the blocks that are allocated to a respective process.**
5. **Value of Fragmentation is optional to display to keep track of wasted memory.**
6. **Stop.**

Source Code:

```
#include<stdio.h>

void main()
{
        int fragment[20],b[20],p[20],i,j,nb,np,temp,lowest=9999;
        static int barray[20],parray[20];

        printf("\n\t\t\tMemory Management Scheme - Best Fit");
        printf("\nEnter the number of blocks:");
        scanf("%d",&nb);
        printf("Enter the number of processes:");
        scanf("%d",&np);

        printf("\nEnter the size of the blocks:-\n");
        for(i=1;i<=nb;i++)
  {
                printf("Block no.%d:",i);
     scanf("%d",&b[i]);
  }
```

```
        printf("\nEnter the size of the processes :-\n");
        for(i=1;i<=np;i++)
    {
       printf("Process no.%d:",i);
       scanf("%d",&p[i]);
    }

        for(i=1;i<=np;i++)
        {
                for(j=1;j<=nb;j++)
                {
                        if(barray[j]!=1)
                        {
                                temp=b[j]-p[i];
                                if(temp>=0)
                                        if(lowest>temp)
                                        {
                                                parray[i]=j;
                                                lowest=temp;
                                        }
                        }
                }

                fragment[i]=lowest;
                barray[parray[i]]=1;
                lowest=10000;
        }

        printf("\nProcess_no\tProcess_size\tBlock_no\tBlock_size\tFragment");
        for(i=1;i<=np && parray[i]!=0;i++)

printf("\n%d\t\t%d\t\t%d\t\t%d\t\t%d",i,p[i],parray[i],b[parray[i]],fragment[i]);
}
```

Output:

```
File  Edit  View  Search  Terminal  Help
sachin@sachin-VirtualBox:~/Desktop/programs$ ./a.out

                        Memory Management Scheme - Best Fit
Enter the number of blocks:5
Enter the number of processes:4

Enter the size of the blocks:-
Block no.1:100
Block no.2:500
Block no.3:200
Block no.4:300
Block no.5:600

Enter the size of the processes :-
Process no.1:212
Process no.2:417
Process no.3:112
Process no.4:426

Process_no      Process_size    Block_no        Block_size      Fragment
1               212             4               300             88

2               417             2               500             83

3               112             3               200             88

4               426             5               600             174
sachin@sachin-VirtualBox:~/Desktop/programs$
```