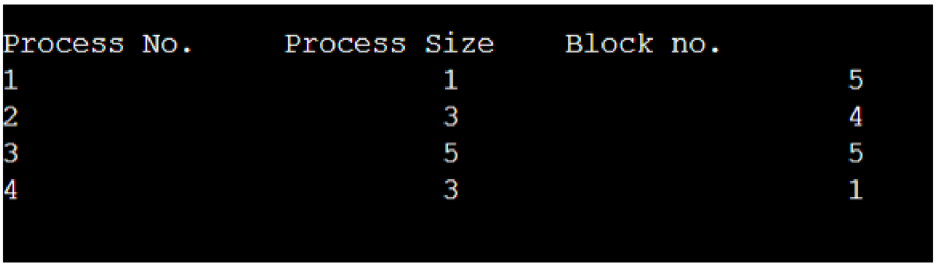
EXP 21

```
#include <stdio.h>


void implimentWorstFit(int blockSize[], int blocks, int processSize[], int processes)
{
    int allocation[processes];


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


    for (int i=0; i<processes; i++)
    {


        int indexPlaced = -1;
        for (int j=0; j<blocks; j++)
        {
            if (blockSize[j] >= processSize[i])
            {
                if (indexPlaced == -1)
                    indexPlaced = j;


                else if (blockSize[indexPlaced] < blockSize[j])
                    indexPlaced = j;
            }
        }


        if (indexPlaced != -1)
        {
            allocation[i] = indexPlaced;
            blockSize[indexPlaced] -= processSize[i];
        }
    }

    printf("\nProcess No.\tProcess Size\tBlock no.\n");
    for (int i = 0; i < processes; i++)
    {
        printf("%d \t\t\t %d \t\t\t", i+1, processSize[i]);
        if (allocation[i] != -1)
            printf("%d\n",allocation[i] + 1);
        else
            printf("Not Allocated\n");
    }
}

int main()
{
    int blockSize[] = {5, 4, 3, 6, 7};
    int processSize[] = {1, 3, 5, 3};
    int blocks = sizeof(blockSize)/sizeof(blockSize[0]);
    int processes = sizeof(processSize)/sizeof(processSize[0]);

    implimentWorstFit(blockSize, blocks, processSize, processes);

    return 0 ;
}
```



```
Process No.     Process Size    Block no.
1                        1                      5
2                        3                      4
3                        5                      5
4                        3                      1
```