

```
int main()
{
    int fragments[10], blocks[10], files[10];
    int m, n, number_of_blocks, number_of_files, temp, top = 0;
    static int block_occupied[10], file_occupied[10];
    printf("\n Enter the total number of blocks: \t");
    scanf("%d", &number_of_blocks);
    printf("\n Enter the total number of files: \t");
    scanf("%d", &number_of_files);
    printf("\n Enter the size of blocks: \n");
    for (m = 0; m < number_of_blocks; m++)
    {
        printf("Block No. [%d]: \t", m+1);
        scanf("%d", &blocks[m]);
    }
    printf("\n Enter the size of the files: \n");
    for (m = 0; m < number_of_files; m++)
    {
        for (m = 0; m <
        printf("File No. [%d]: \t", m+1);
        scanf("%d", &files[m]);
    }
    for (m = 0; m < number_of_files; m++)
    {
        for (n = 0; n < number_of_blocks; n++)
        {
            if (block_occupied[n] != 1)
            {
                temp = blocks[n] - files[m];
                if (temp >= 0)
                {
                    if (top < temp)
```

```
{  
    file_ors[m] = n;  
    top = temp;  
}  
}  
}  
fragments[m] = top;  
block_ors[file_ors[m]] = 1;  
top = 0;  
}  
}  
printf("\n File number | file size | block number | block  
      size | fragment");  
{ for (m = 0; m < number_of_files; m++)  
    printf("\n %d | %d | %d | %d | %d",  
m, files[m], file_ors[m], blocks[file_ors[m]],  
fragments[m]);  
    printf("\n");  
    return 0;  
}
```