1. Construct a C program to implement best fit algorithm of memory management.

#include <stdio.h>

#define MAX 25

int main() {

int blockSize[MAX], processSize[MAX];

int blockCount, processCount;

int allocation[MAX];

printf("Enter number of memory blocks: ");

scanf("%d", &blockCount);

printf("Enter sizes of %d memory blocks:\n", blockCount);

for (int i = 0; i < blockCount; i++)

scanf("%d", &blockSize[i]);

printf("Enter number of processes: ");

scanf("%d", &processCount);

printf("Enter sizes of %d processes:\n", processCount);

for (int i = 0; i < processCount; i++)

scanf("%d", &processSize[i]);

for (int i = 0; i < processCount; i++)

allocation[i] = -1;

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

int bestIndex = -1;

for (int j = 0; j < blockCount; j++) {

if (blockSize[j] >= processSize[i]) {

if (bestIndex == -1 || blockSize[j] < blockSize[bestIndex])

bestIndex = j;

}

}

if (bestIndex != -1) {

allocation[i] = bestIndex;

blockSize[bestIndex] -= processSize[i];

}

}

printf("\nProcess No.\tProcess Size\tBlock No.\n");

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

printf(" %d\t\t%d\t\t", i + 1, processSize[i]);

if (allocation[i] != -1)

printf("%d\n", allocation[i] + 1);

else

printf("Not Allocated\n");

}

return 0;

}