Problem 9:

(a) Write a program to implement the First fit memory management algorithm. Program should take input total no. of memory block, their sizes, process name and process size. Output of program should give the details about memory allocated to process with fragmentation detail.

(b) Write a program to implement the Next fit memory management algorithm. Program should take input total no. of memory block, their sizes, process name and process size. Output of program should give the details about memory allocated to process with fragmentation detail.

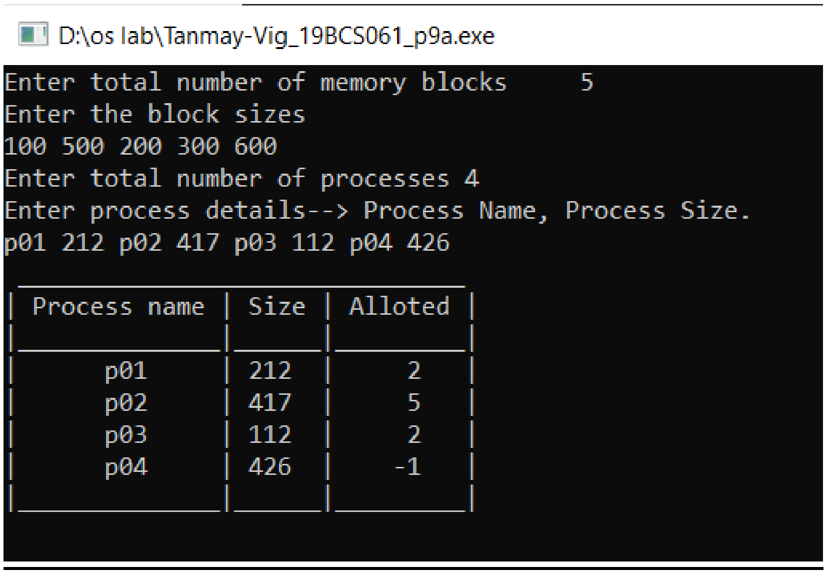
Answer:

(a) Source code:

```
#include<stdio.h>

#include<limits.h>

#include<stdbool.h>
```

```
typedef struct{
    char process_name[3];
    int size,allocated;
}process;

void print_table(process pr[],int m){
    puts(" ____________________________");
    puts("| Process name | Size | Alloted |");
    puts("|_____________|______|________|");
    for(int i=0;i<m;i++){
    printf("|      %s     | %3d  |    %2d   |\n",
                    pr[i].process_name,pr[i].size,pr[i].allocated);
    }
    puts("|_____________|______|________|");
}

void main()
{
    int n,m,i,j;
    printf("Enter total number of memory blocks\t");
    scanf("%d",&n);
    int mem_block[n];
    printf("Enter the block sizes\n");
    for(i=0;i<n;i++){
        scanf("%d",&mem_block[i]);
    }
    printf("Enter total number of processes\t");
    scanf("%d",&m);
    process pr[m];
    printf("Enter process details--> Process Name, Process Size.\n");
    for(i=0;i<m;i++){
        scanf("%s %d",pr[i].process_name,&pr[i].size);
        pr[i].allocated=-1;
    }
    for(i=0;i<m;i++){
        for(j=0;j<n;j++){
            if(mem_block[j]>=pr[i].size){
                mem_block[j]-=pr[i].size;
                pr[i].allocated=j+1;
                break;
            }
        }
    }
    print_table(pr,m);
}
```

Output:



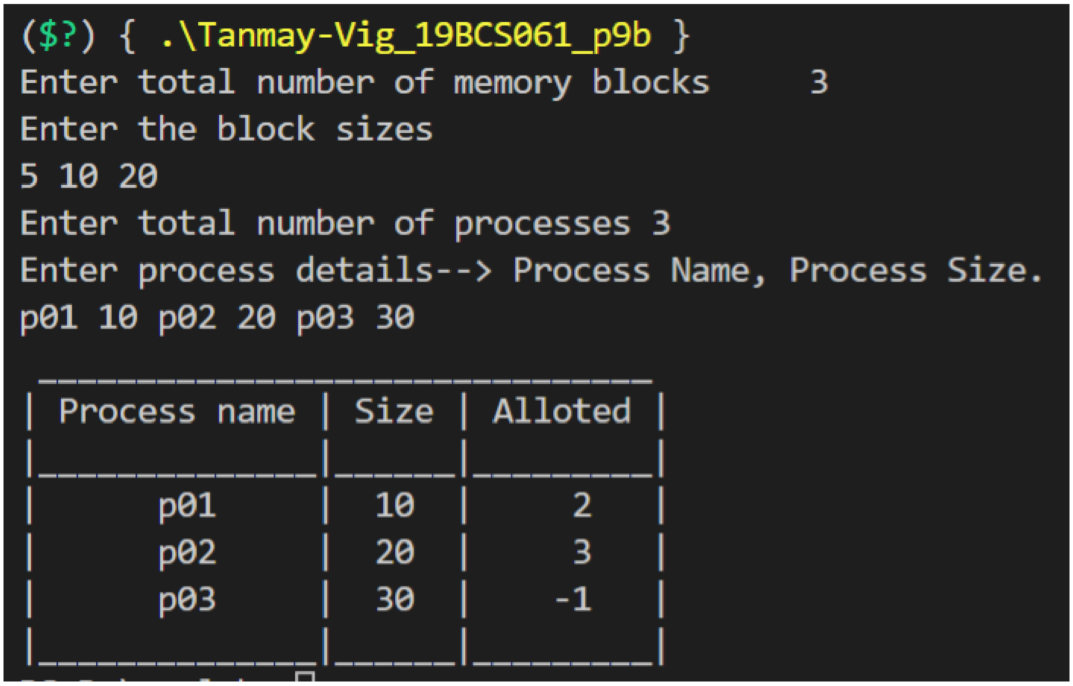
```
Enter total number of memory blocks     5
Enter the block sizes
100 500 200 300 600
Enter total number of processes 4
Enter process details--> Process Name, Process Size.
p01 212 p02 417 p03 112 p04 426
 _______________________________
| Process name | Size | Alloted |
|______________|______|_________|
|      p01     | 212  |     2   |
|      p02     | 417  |     5   |
|      p03     | 112  |     2   |
|      p04     | 426  |    -1   |
|______________|______|_________|
```

(b) Source Code:

```
#include<stdio.h>


typedef struct{
    char process_name[3];
    int size,allocated;
```

```
}process;

void algorithm(int mem_block[],process pr[],int m, int n){
    int i,j,k=0;
    for(i=0;i<m;i++){
        j=k;
        while(1){
            if(mem_block[j]>=pr[i].size){
                mem_block[j]-=pr[i].size;
                pr[i].allocated=j+1;
                k=j;
                break;
            }
            j=(j+1)%n;
            if(j==k) break;
        }
    }
}

void print_table(process pr[],int m){
    puts(" _________________________________");
    puts("| Process name | Size | Alloted |");
    puts("|______________|______|_________|");
    for(int i=0;i<m;i++){
    printf("|      %s     | %3d  |    %2d   |\n",
                    pr[i].process_name,pr[i].size,pr[i].allocated);
    }
    puts("|______________|______|_________|");
}

void main(){
    int n,m,i,j;
    printf("Enter total number of memory blocks\t");
    scanf("%d",&n);
    int mem_block[n];
    printf("Enter the block sizes\n");
    for(i=0;i<n;i++){
        scanf("%d",&mem_block[i]);
    }
    printf("Enter total number of processes\t");
    scanf("%d",&m);

    process pr[m];

    printf("Enter process details--> Process Name, Process
Size.\n");

    for(i=0;i<m;i++){

        scanf("%s %d",pr[i].process_name,&pr[i].size);

        pr[i].allocated=-1;

    }

    algorithm(mem_block,pr,m,n);

    print_table(pr,m);

}
```

Output:



```
($?) { .\Tanmay-Vig_19BCS061_p9b }
Enter total number of memory blocks     3
Enter the block sizes
5 10 20
Enter total number of processes 3
Enter process details--> Process Name, Process Size.
p01 10 p02 20 p03 30

 _______________________________
| Process name | Size | Alloted |
|______________|______|_________|
|      p01     |  10  |    2    |
|      p02     |  20  |    3    |
|      p03     |  30  |   -1    |
|______________|______|_________|
```