Experiment 13

```c
#include <stdio.h>

void firstFit(int blockSize[], int m, int processSize[], int n) {
    int allocation[n];
    for (int i = 0; i < n; i++) allocation[i] = -1;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            if (blockSize[j] >= processSize[i]) {
                allocation[i] = j;
                blockSize[j] -= processSize[i];
                break;
            }
        }
    }
    printf("\nProcess No\tProcess Size\tBlock No\n");
    for (int i = 0; i < n; i++)
        printf("%d\t\t%d\t\t%d\n", i+1, processSize[i], allocation[i] + 1);
}

int main() {
    int blockSize[10], processSize[10];
    int m, n;
    printf("Enter number of memory blocks: ");
    scanf("%d", &m);
    printf("Enter block sizes: ");
    for (int i = 0; i < m; i++) scanf("%d", &blockSize[i]);

    printf("Enter number of processes: ");
    scanf("%d", &n);
    printf("Enter process sizes: ");
    for (int i = 0; i < n; i++) scanf("%d", &processSize[i]);

    firstFit(blockSize, m, processSize, n);

    return 0;
}
```

```
Enter number of memory blocks: 5
Enter block sizes: 100 500 200 300 600
Enter number of processes: 4
Enter process sizes: 212 417 112 426

Process No          Process Size     Block No
1                   212              2
2                   417              5
3                   112              2
4                   426              0


----------------------------------
Process exited after 40.46 seconds with return value 0
Press any key to continue . . .
```