Name : Riyadul Islam , ID : 22234103007 , Sec:1,
Intake : 50, Dept: CSE

Problem-1:

Problem1_007.cpp

```cpp
#include <bits/stdc++.h>
using namespace std;
int arr[1000000];
void my_sort(int *arr, int *arr_end){
    int n = arr_end - arr;
    for (int gap = n / 2; gap > 0; gap /= 2)
    {
        for (int i = gap; i < n; i += 1)
        {
            int temp = arr[i];
            int j;
            for (j = i; j >= gap && arr[j - gap] > temp; j -= gap)
                arr[j] = arr[j - gap];
            arr[j] = temp;
        }
    }
}

int main()
{
    int n;
    int i = 0;
    while (1)
    {
        cin >> n;
        if (n == 0){
            return 0;
```

*Problem1__007.cpp

```
27              return 0;
28
29          }
30          if (n == -1)
31          {
32              my_sort(arr, arr + i);
33              if (i % 2 == 0){
34                  cout << arr[i / 2 - 1] << " ";
35                  for (int j = i / 2-1; j < i - 1; j++)
36                  arr[j] = arr[j + 1];
37
38              }
39              else {
40                  cout << arr[i / 2] << " ";
41                  for (int j = i / 2; j < i - 1; j++)
42                  arr[j] = arr[j + 1];
43                  }
44
45              i--;
46          }
47          else
48          {
49              arr[i] = n;
50              i++;}
51      }
52      return 0;
53  }
54
```

*Problem1__007.cpp

C:\Users\Riyadh\OneDrive\Desktop\test\Problem1__007.exe

```
9
10
2
5
1
18
-1
5
```

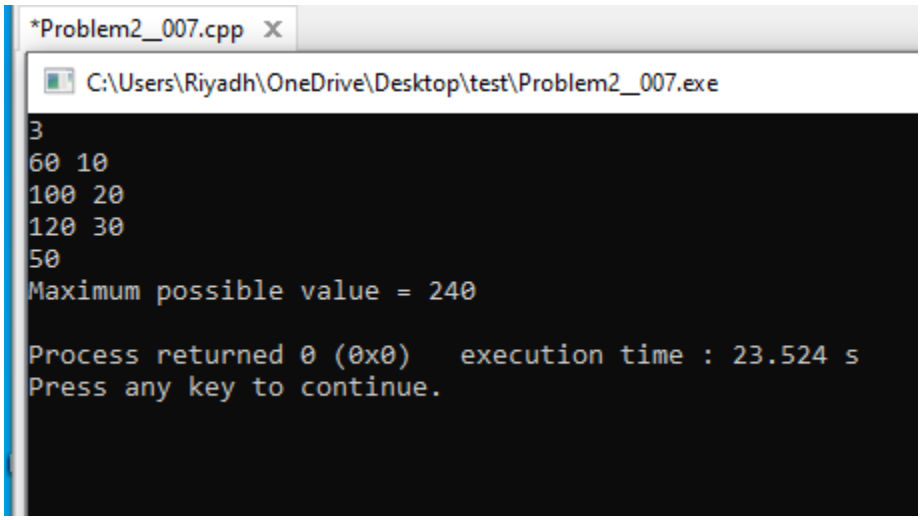
Problem: 2

Problem2_007.cpp

```cpp
#include <bits/stdc++.h>
using namespace std;
struct Item
{
    int value, weight;
};
void fractional_Knapsack(float W, Item arr[], int n)
{

    for (int gap = n / 2; gap > 0; gap /= 2)
    {
        for (int i = gap; i < n; i += 1)
        {
            Item temp = arr[i];
            int j;
            for (j = i; j >= gap && (arr[j - gap].value / arr[j - gap].weight) < (temp.value / temp.weight); j -= gap)
                arr[j] = arr[j - gap];
            arr[j] = temp;
        }
    }

    float res = 0;
    for (int i = 0; i < n; i++)
    {
        if (arr[i].weight <= W)
        {
```

*Problem2_007.cpp

```cpp
                res += arr[i].value;
                W -= arr[i].weight;
            }
            else
            {
                res += arr[i].value * (W / arr[i].weight);
                break;
            }
        }
        cout << "Maximum possible value = " << res << endl;

}
int main()
{
    Item arr[10000] ;
    int n;
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        cin >> arr[i].value >> arr[i].weight;
    }
    float W ;
    cin >> W;
    fractional_Knapsack(W, arr, n);
    return 0;
}
```



Problem:3

Problem3__007.cpp

```cpp
#include <bits/stdc++.h>
using namespace std;

struct Activity {
    int start, finish;
};
bool activityCompare(Activity s1, Activity s2) {
    return (s1.finish < s2.finish);
}

void activity_selection(Activity arr[], int n) {
    sort(arr, arr+n, activityCompare);
    int i = 0;
    cout << "(" << arr[i].start << ", " << arr[i].finish << "), ";

    for (int j = 1; j < n; j++) {
        if (arr[j].start >= arr[i].finish) {
            cout << "(" << arr[j].start << ", " << arr[j].finish << "), ";
            i = j;
        }
    }
}

```

Problem3_007.cpp

```cpp
    int i = 0;
    cout << "(" << arr[i].start << ", " << arr[i].finish << "), ";

    for (int j = 1; j < n; j++) {
        if (arr[j].start >= arr[i].finish) {
            cout << "(" << arr[j].start << ", " << arr[j].finish << "), ";
            i = j;
        }
    }
}

int main() {
    Activity arr[10000];
    int n;
    cin >> n;
    for (int i = 0; i < n; i++) {
        cin >> arr[i].start >> arr[i].finish;
    }
     cout << "Following activities are selected: \n";
    activity_selection(arr, n);
    return 0;
}
```

Problem3_007.cpp

```
C:\Users\Riyadh\OneDrive\Desktop\test\Problem3_007.exe
5
1 2
3 4
0 6
5 7
8 9
Following activities are selected:
(1, 2), (3, 4), (5, 7), (8, 9),
Process returned 0 (0x0)   execution time : 19.197 s
Press any key to continue.
```