**Fractional Knapsack**

#include <bits/stdc++.h>

static bool cmp(pair<int,int> a, pair<int,int> b){

double x1 = (double)a.second / (double)a.first;

double x2 = (double)b.second / (double)b.first;

return x1>x2;

}

double maximumValue (vector<pair<int, int>>& items, int n, int w)

{

// Write your code here.

// ITEMS contains {weight, value} pairs.

sort(items.begin(),items.end(), cmp);

double ans = 0.0;

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

if (items[i].first <= w) {

w -= items[i].first;

ans += items[i].second;

}

else {

ans

+= items[i].second

\* ((double)w / (double)items[i].first);

break;

}

}

return ans;

}