Knapsack using greedy technique

**# include<stdio.h>**

**Void knapsack(int n, float weight[], float profit[], float capacity) {**

**Float x[20], tp = 0;**

**Int i, j, u;**

**U = capacity;**

**For (i = 0; i < n; i++)**

**X[i] = 0.0;**

**For (i = 0; i < n; i++) {**

**If (weight[i] > u)**

**Break;**

**Else {**

**X[i] = 1.0;**

**Tp = tp + profit[i];**

**U = u – weight[i];**

**}**

**}**

**If (i < n)**

**X[i] = u / weight[i];**

**Tp = tp + (x[i] \* profit[i]);**

**Printf(“\nThe result vector is:- “);**

**For (i = 0; i < n; i++)**

**Printf(“%f\t”, x[i]);**

**Printf(“\nMaximum profit is:- %f”, tp);**

**}**

**Int main() {**

**Float weight[20], profit[20], capacity;**

**Int num, i, j;**

**Float ratio[20], temp;**

**Printf(“\nEnter the no. Of objects:- “);**

**Scanf(“%d”, &num);**

**Printf(“\nEnter the wts and profits of each object:- “);**

**For (i = 0; i < num; i++) {**

**Scanf(“%f %f”, &weight[i], &profit[i]);**

**}**

**Printf(“\nEnter the capacityacity of knapsack:- “);**

**Scanf(“%f”, &capacity);**

**For (i = 0; i < num; i++) {**

**Ratio[i] = profit[i] / weight[i];**

**}**

**For (i = 0; i < num; i++) {**

**For (j = i + 1; j < num; j++) {**

**If (ratio[i] < ratio[j]) {**

**Temp = ratio[j];**

**Ratio[j] = ratio[i];**

**Ratio[i] = temp;**

**Temp = weight[j];**

**Weight[j] = weight[i];**

**Weight[i] = temp;**

**Temp = profit[j];**

**Profit[j] = profit[i];**

**Profit[i] = temp;**

**}**

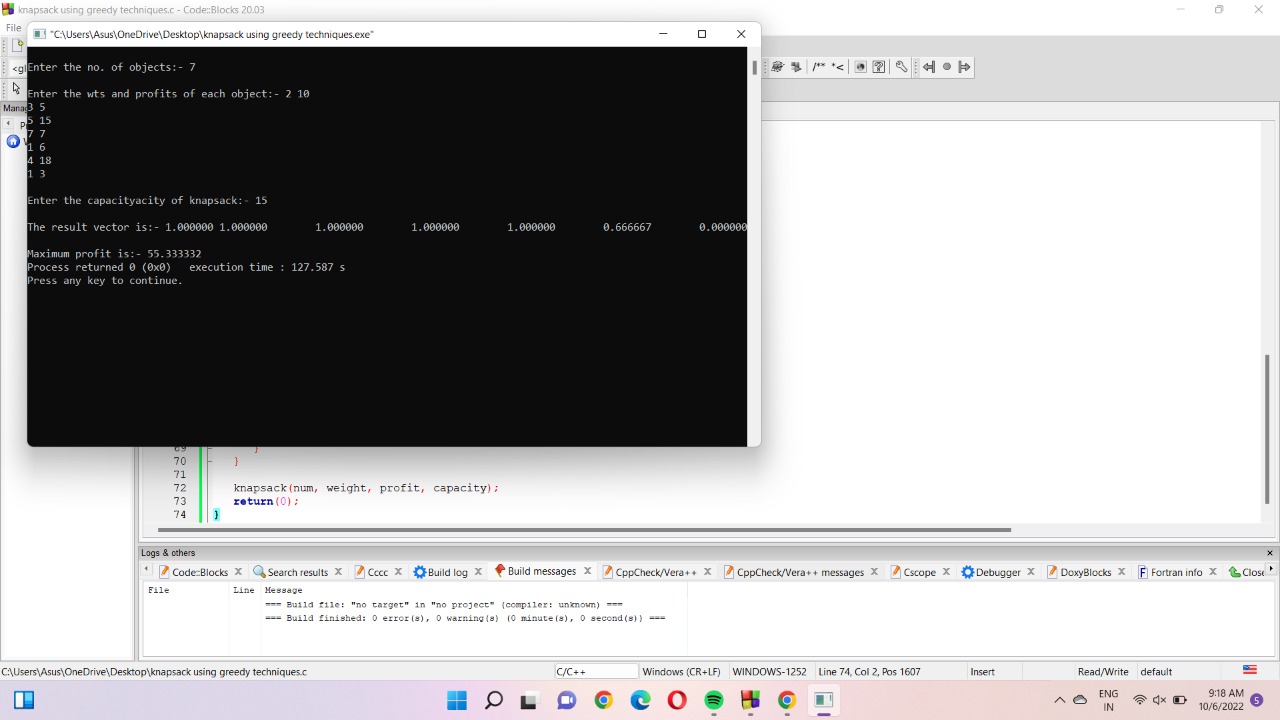
**}**

**}**

**Knapsack(num, weight, profit, capacity);**

**Return(0);**

**}**

****