**Program – 11**

**Aim – Write a program to implement knapsack problem**

**0/1 Knapsack**

**Soruce Code –**

#include<stdio.h>

#include<conio.h>

int max(int a, int b){

return (a > b) ? a : b;

}

int knapSack(int W, int wt[], int val[], int n){

if (n == 0 || W == 0)

return 0;

if (wt[n - 1] > W)

return knapSack(W, wt, val, n - 1);

else

return max(

val[n - 1] + knapSack(W - wt[n - 1], wt, val, n - 1),

knapSack(W, wt, val, n - 1));

}

main(){

int val[10],wt[10],i,W,n;

printf("Enter the number of elements (max 10): ");

scanf("%d",&n);

printf("Enter the value\n");

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

scanf("%d",&val[i]);

}

printf("Enter the weight\n");

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

scanf("%d",&wt[i]);

}

printf("Enter the weight of bag: ");

scanf("%d",&W);

printf("%d", knapSack(W, wt, val, n));

}

**Fractional Knapsack**

**Source Code –**

#include <iostream>

#include<conio.h>

#include<stdio.h>

using namespace std;

typedef struct{

int v;

int w;

float d;

}Item;

void input(Item items[],int sizeOfItems) {

cout << "Enter total "<< sizeOfItems <<" item's values and weight" << endl;

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

cout << "Enter "<< i+1 << " Value: ";

cin >> items[i].v;

cout << "Enter "<< i+1 << " value Weight: ";

cin >> items[i].w;

}

}

void display(Item items[], int sizeOfItems) {

int i;

cout << "values: ";

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

cout << items[i].v << "\t";

}

cout << endl << "weight: ";

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

cout << items[i].w << "\t";

}

cout << endl;

}

bool compare(Item i1, Item i2) {

return (i1.d > i2.d);

}

float knapsack(Item items[], int sizeOfItems, int W) {

int i, j, pos;

Item mx, temp;

float totalValue = 0, totalWeight = 0;

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

items[i].d = items[i].v / items[i].w;

}

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

if(totalWeight + items[i].w<= W) {

totalValue += items[i].v ;

totalWeight += items[i].w;

} else {

int wt = W-totalWeight;

totalValue += (wt \* items[i].d);

totalWeight += wt;

break;

}

}

cout << "total weight in bag " << totalWeight<<endl;

return totalValue;

}

int main() {

int W,n;

Item items[15];

cout << "Enter the number of elements (max 15): ";

cin >> n;

input(items, n);

cout << "Entered data \n";

display(items,n);

cout<< "Enter Knapsack weight \n";

cin >> W;

float mxVal = knapsack(items, n, W);

cout << "Max value for "<< W <<" weight is "<< mxVal;

}