/*
ASSIGNMENT NO 1(b) :
Problem Description:0/1 knapsack using Dynamic programming.
*/

PROGRAM:

```
#include<iostream>
#include<math.h>
using namespace std;

int n=5;
int W=7;
int vajan[]={ 2,4,1,3,4};
int profit[]={ 15,14,10,45,30};
int dp[6][8];

void display()                          //function to display matrix
{
        for(int i=0;i<n+1;i++)
        {
                for(int j=0;j<W+1 ;j++)
                {
                        cout<<dp[i][j]<<"\t";
                }
                cout<<"\n";
        }
}
void knapsack(int vajan[], int profit[], int W)                 //knapsack function
{

        for(int i=0;i<n+1;i++)
        {
                for(int j=0;j<W+1 ;j++)
                {
                        dp[i][j]=0;                     // Initializing the matrix with zero
                }
        }


        for(int i=1;i<n+1;i++)
        {
                for(int j=1;j<W+1;j++)
                {
```

```
                int w=vajan[i-1];
                int v=profit[i-1];


                if(w<=j)
                {
                        //include
                        int incprofit= v+ dp[i-1][j-w];

                        //exclude
                         int excprofit= dp[i-1][j];

                        dp[i][j]=max(incprofit ,excprofit);

                }
                else
                {
                        //always exclude
                        int excprofit= dp[i-1][j];
                        dp[i][j]=excprofit;
                }

            }
        }

}

int main()
{
        cout<<"VAJAN ARRAY :\n";
        for(int i=0;i<5;i++)
        {
                cout<<vajan[i]<<"\t";           //array of weight
        }


        cout<<"\n\nPROFIT ARRAY :\n";
        for(int i=0;i<5;i++)
        {
                cout<<profit[i]<<"\t";          //array of profit
        }
        cout<<"\n\n";
        cout<<"\n\nBEFORE MATRIX :\n";
```

```
        display();
        knapsack(vajan, profit , W);
        cout<<"\n\nAFTER MATRIX :\n";
        display();


}


//output :
/*
VAJAN ARRAY :
2       4       1       3       4

PROFIT ARRAY :
15      14      10      45      30


BEFORE MATRIX :
0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0


AFTER MATRIX :
0       0       0       0       0       0       0       0
0       0       15      15      15      15      15      15
0       0       15      15      15      15      29      29
0       10      15      25      25      25      29      39
0       10      15      45      55      60      70      70
0       10      15      45      55      60      70      75
*/
```