6)Dynamic Knapsack Problem

```
#include<stdio.h>

#include<conio.h>

void Dknapsack();

int max(int,int);

int i,j,n,m,p[10],w[10],v[10][10];

void main()

{

 printf("\n Enter the no.of item :\n");

 scanf("%d",&n);

 printf("\n Enter the weight of the each item:\n");

 for(i=1;i<=n;i++)

 {

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

 }

 printf("\n Enter the profit of each item:\n");

 for(i=1;i<=n;i++)

 {

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

 }

 printf("\n Enter the knapsack's capacity :\t ");

 scanf("%d",&m);

 Dknapsack();

}

void Dknapsack()

{

 int x[10];
```

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

{

for(j=0;j<=m;j++)

{

if(i==0 || j ==0)

{

v[i][j]=0;

}

else if(j-w[i]<0)

{

v[i][j]=v[i-1][j];

}

else{

v[i][j]= max(v[i-1][j],v[i-1][j-w[i]]+p[i]);

}

}

}

printf("\n the output is : \n");

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

{

for(j=0;j<=m;j++)

{

printf("%d\t",v[i][j]);

}

printf("\n\n");

}

printf("\n the optimal solution is %d ",v[n][m]);

printf("\n the solution vector is : \n");
```

```
 for(i=n;i>=1;i--)
 {
 if(v[i][m]!=v[i-1][m])
 {
 x[i]=1;
 m=m-w[i];
 }
 else
 {
 x[i]=0;
 }
 }
 for(i=1;i<=n;i++)
 {
 printf("%d\t",x[i]);
 }
}
int max(int x,int y)
{
 if(x>y)
 {
 return x;
 }
 else{
   return y;
 }
}
```