

## KNAPSACK PROBLEM (KSP)

### BM218 ALGORİTMALAR

#### ÖDEV 4

i.  $N \leq K \leq N^2 \rightarrow N = 5$  için  $5 \leq K \leq 25$  olmalıdır.

ii.  $1 \leq W \leq K$

iii.  $1 \leq V \leq N^2$

A,B,C,D,E olmak üzere 5 nesne için tasarlanacaktır. Çantanın maksimum kapasitesi (K) ile her bir nesnenin değeri ve ağırlığı kullanıcı tarafından dışarıdan girilecektir

Nesneler	A	B	C	D	E
Ağırlıklar (W)	?				
Değerler (V)	?				

Knapsack Problemini 0/1 KSP Problemi olarak ele alacağım.

Tabloları belirlediğim değerlere göre doldurdum. Uygulamamda da tablodaki değerlerin doğruluğunu kanıtlamak amacıyla aynı değerlere yer verdim

Nesneler(W=11)	A	B	C	D	E
Ağırlıklar(W)	<b>1</b>	<b>2</b>	<b>5</b>	<b>6</b>	<b>7</b>
Değerler(V)	<b>1</b>	<b>6</b>	<b>18</b>	<b>22</b>	<b>28</b>

```
Run: untitled28 x
C:\Users\Pro\CLionProjects\untitled28\cmake-build-debug\untitled28.exe
Number of objects:5
5
Capacity of objects:11
11
Enter weight and value of object1:1 1
1 1
Enter weight and value of object2:2 6
2 6
Enter weight and value of object3:5 18
5 18
Enter weight and value of object4:6 22
6 22
Enter weight and value of object5:7 28
7 28
```

	0	1	2	3	4	5	6	7	8	9	10	11
w1=1,v1=1	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>
w1=2,v1=6	<b>0</b>	<b>1</b>	<b>6</b>	<b>7</b>	<b>7</b>	<b>7</b>	<b>7</b>	<b>7</b>	<b>7</b>	<b>7</b>	<b>7</b>	<b>7</b>
w1=5,v1=18	<b>0</b>	<b>1</b>	<b>6</b>	<b>7</b>	<b>7</b>	<b>18</b>	<b>19</b>	<b>24</b>	<b>25</b>	<b>25</b>	<b>25</b>	<b>25</b>
w1=6,v1=22	<b>0</b>	<b>1</b>	<b>6</b>	<b>7</b>	<b>7</b>	<b>18</b>	<b>22</b>	<b>24</b>	<b>28</b>	<b>29</b>	<b>29</b>	<b><u>40</u></b>
w1=7,v1=28	<b>0</b>	<b>1</b>	<b>6</b>	<b>7</b>	<b>7</b>	<b>18</b>	<b>22</b>	<b>28</b>	<b>29</b>	<b>34</b>	<b>35</b>	<b><u>40</u></b>

```
0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 1 1 1 1 1
0 1 6 7 7 7 7 7 7 7 7 7
0 1 6 7 7 18 19 24 25 25 25 25
0 1 6 7 7 18 22 24 28 29 29 40
0 1 6 7 7 18 22 28 29 34 35 40

-----
Max possible profit=40
-----
```

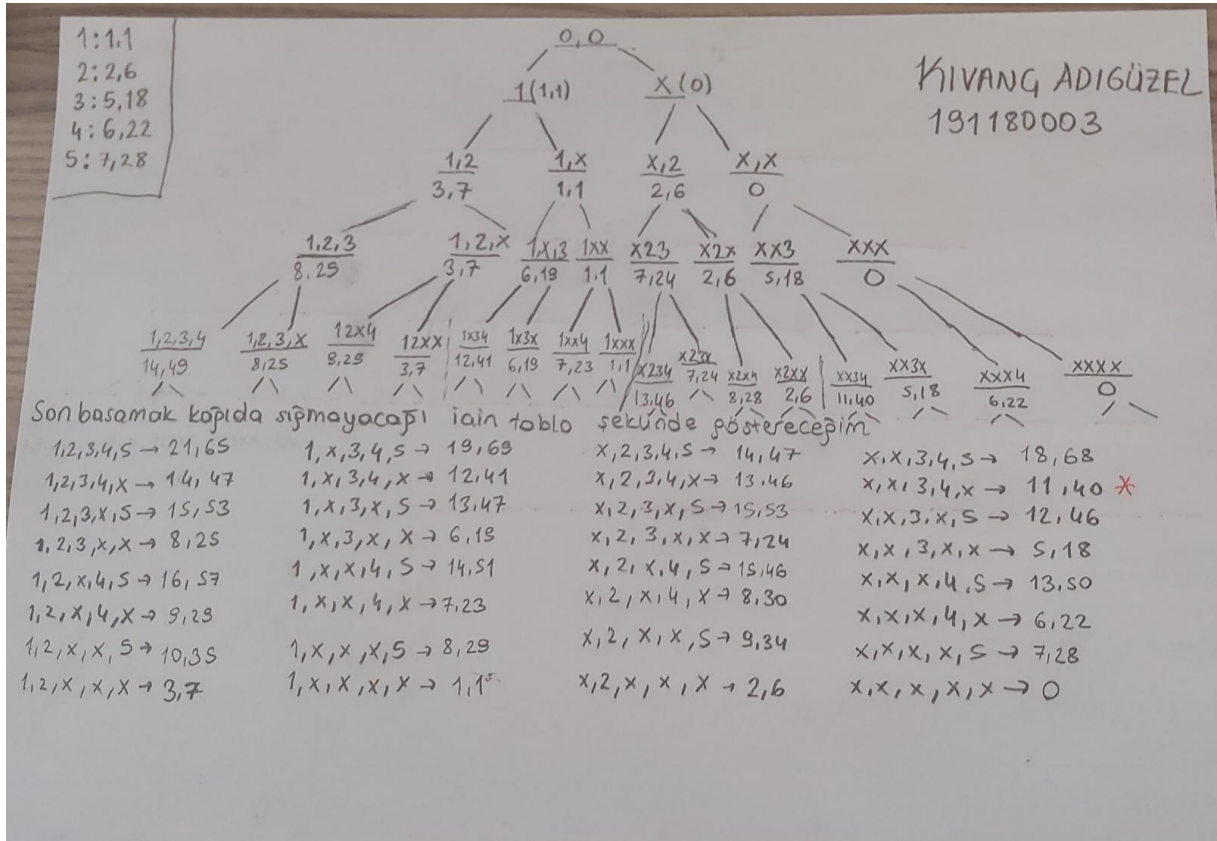
Son olarak, yaprak düğümler içindeki en uygun çözümü sunan düğümü ve çözüm değerini ekranda gösteriniz

#### KSP KODUM:

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>

void main()
{
    int v[20],w[20],i,j,n,W;
    void knapsack(int[],int[],int,int);
    system("cls");
    printf("Number of objects: ");
    scanf("%d",&n);
    printf("Capacity of objects: ");
    scanf("%d",&W);
    for(i=1;i<=n;i++)
    {
        printf("Enter weight and value of object%d: ",i);
        scanf("%d",&w[i]);
        scanf("%d",&v[i]);
    }
    knapsack(v,w,n,W);
    getch();
}
```

```
void knapsack(int v[],int w[20],int n,int W)
{
    int k[20][20],i,j;
    for(i=0;i<=n;i++)
    {
        for(j=0;j<=W;j++)
        {
            if(i==0 || j==0)
            {
                k[i][j]=0;
            }
            else if(j<w[i])
            {
                k[i][j]=k[i-1][j];
            }
            else
            {
                if(k[i-1][j]>k[i-1][j-w[i]]+v[i])
                {
                    k[i][j]=k[i-1][j];
                }
                else
                {
                    k[i][j]=k[i-1][j-w[i]]+v[i];
                }
            }
        }
    }
    for(i=0;i<=n;i++)
    {
        for(j=0;j<=W;j++)
        {
            printf("    %d",k[i][j]);
        }
        printf("\n");
    }
    printf("_____\n");
    printf("\nMax possible profit=%d",k[n][W]);
    printf("\n_____\n");
    getch();
}
```



Tabloda ve koda sağlamış olduğum 40 değerini bir de diyagramla kanıtlamış oldum.