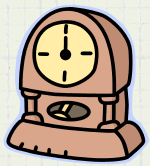
0/1 Knapsack subproblems



$175, 10kg

$90, 9kg

$20, 4kg

$50, 2kg

$10, 1kg

$200, 20kg

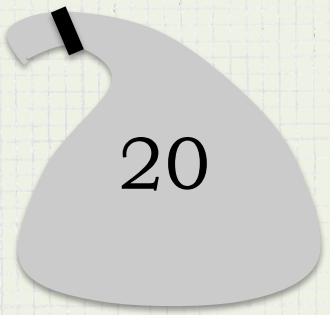
```
value,taken = chooseBest(items,maxWeight)
```

Alternative #1: don't take first item

Subproblem:

$90, 9kg

$20, 4kg



$50, 2kg

$10, 1kg

$200, 20kg

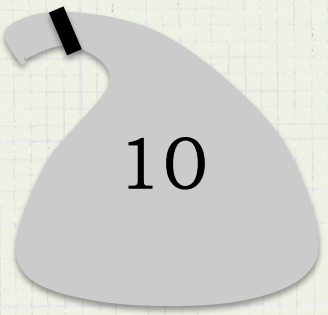
```
v1,t1 = chooseBest(items[1:],maxWeight)
```

Alternative #2: take first item

Subproblem:

$90, 9kg

$20, 4kg



$50, 2kg

$10, 1kg

$200, 20kg

```
v2,t2 = chooseBest(items[1:],maxWeight - item[0].getWeight())
v2 += item[0].getValue()  # add first item to knapsack
t2 += [item[0]]
```

Combine solutions

Choose most valuable alternative:

Where's the overlap?

- Overlap not guaranteed

- But if there are many items to consider: