

### # 0-1 Knapsack problem using dynamic

```
def knapSack(W, wt, val, n):  
    dp = [0 for i in range(W + 1)]  
    for i in range(1, n + 1):  
        for w in range(W, 0, -1):  
            if wt[i - 1] <= w:  
                dp[w] = max(dp[w], dp[w - wt[i - 1]] + val[i - 1])  
    return dp[W]  
  
val = [60, 100, 120]  
wt = [10, 20, 30]  
W = 50  
n = len(val)  
print(knapSack(W, wt, val, n))
```