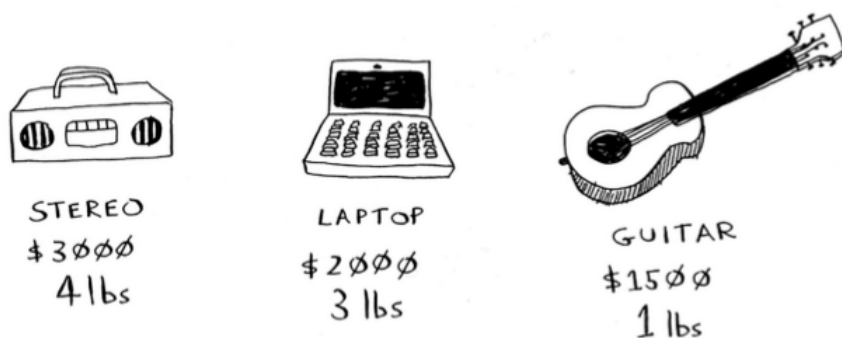
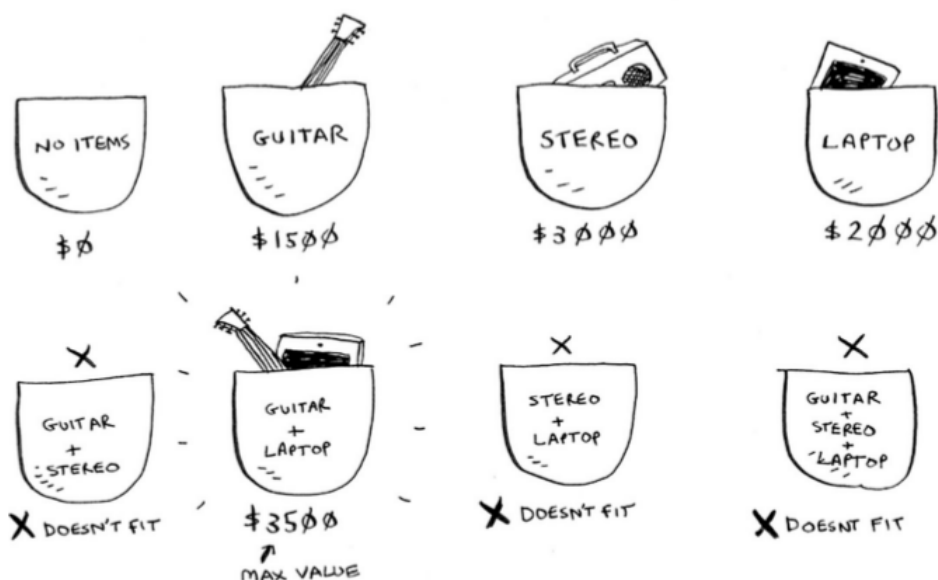


The Knapsack Problem

You have three items that you can put into the knapsack.



What items should you steal so that you steal the maximum money's worth of goods? The simplest algorithm is this: you try every possible set of goods and find the set that gives you the most value.



This works, but it's really slow. For 3 items, you have to calculate 8 possible sets. For 4 items, you have to calculate 16 sets. With every item you add, the number of sets you have to calculate doubles! This algorithm takes $O(2^n)$ time, which is very, very slow.

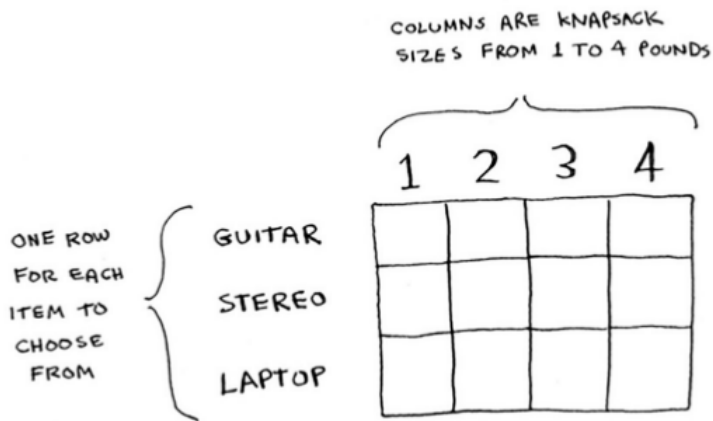
In chapter 8, you saw how to calculate an approximate solution. That solution will be close to the optimal solution, but it may not be the optimal solution. So how do you calculate the optimal solution?

Dynamic programming

Answer: With dynamic programming! Let's see how the dynamic-programming algorithm works here. Dynamic programming starts by solving subproblems and builds up to solving the big problem.

For the knapsack problem, you'll start by solving the problem for smaller knapsacks (or "sub-knapsacks") and then work up to solving the original problem.

Every dynamic-programming algorithm starts with a grid. Here's a grid for the knapsack problem:



The grid starts out empty. You're going to fill in each cell of the grid. Once the grid is filled in, you'll have your answer to this problem!

The guitar row

	1	2	3	4
GUITAR				
STEREO				
LAPTOP				

This is the guitar row, which means you're trying to fit the guitar into the knapsack. At each cell, there's a simple decision: **do you steal the guitar or not?**

The first cell has a knapsack of capacity 1 lb. The guitar is also 1 lb, which means it fits into the knapsack! So the value of this cell is \$1,500, and it contains a guitar.

Like this, each cell in the grid will contain a list of all the items that fit into the knapsack at that point. Let's look at the next cell. Here you have a knapsack of capacity 2 lb. Well, the guitar will definitely fit in there! The same for the rest of the cells in this row.

	1	2	3	4
GUITAR	\$1500 G	\$1500 G	\$1500 G	\$1500 G
STEREO				
LAPTOP				

The stereo row

Now that you're on the second row, you can steal the stereo or the guitar. At every row, you can steal the item at that row or the items in the rows above it. So you can't choose to steal the laptop right now, but you can steal the stereo and/or the guitar. Let's start with the first cell, a knapsack of capacity 1 lb.

You have a knapsack of capacity 1 lb. Will the stereo fit in there? Nope, it's too heavy! Because you can't fit the stereo, \$1,500 remains the max guess for a 1 lb knapsack.

What if you have a knapsack of capacity 4 lb? Aha: the stereo finally fits! The old max value was \$1,500, but if you put the stereo in there instead, the value is \$3,000! Let's take the stereo.

	1	2	3	4
GUITAR	\$1500 G ↓	\$1500 G ↓	\$1500 G ↓	\$1500 G ↓
STEREO	\$1500 G	\$1500 G	\$1500 G	\$3000 S
LAPTOP				

You just updated your estimate! If you have a 4 lb knapsack, you can fit at least \$3,000 worth of goods in it.

The laptop row

Let's do the same thing with the laptop! The laptop weighs 3 lb, so it won't fit into a 1 lb or a 2 lb knapsack. The estimate for the first two cells stays at \$1,500. At 3 lb, the old estimate was \$1,500. But you can choose the laptop instead, and that's worth \$2,000. So the new max estimate is \$2,000! At 4 lb, things get really interesting. This is an important part. The current estimate is \$3,000. You can put the laptop in the knapsack, but it's only worth \$2,000.

	1	2	3	4
GUITAR	\$1500 ↓ G	\$1500 ↓ G	\$1500 ↓ G	\$1500 ↓ G
STEREO	\$1500 ↓ G	\$1500 ↓ G	\$1500 ↓ G	\$3000 ↓ S
LAPTOP	\$1500 ↓ G	\$1500 ↓ G	\$2000 ↓ L	

Hmm, that's not as good as the old estimate. But wait! The laptop weighs only 3 lb, so you have 1 lb free! You could put something in this 1 lb.

$$\text{\$3000 STEREO} \quad \text{vs} \quad \left(\text{\$2000 LAPTOP} + \frac{???}{1 \text{ LB OF FREE SPACE}} \right)$$

What's the maximum value you can fit into 1 lb of space? Well, you've been calculating it all along.

	1	2	3	4
GUITAR	\$1500 ↓ G	\$1500 ↓ G	\$1500 ↓ G	\$1500 ↓ G
STEREO	\$1500 ↓ G	\$1500 ↓ G	\$1500 ↓ G	\$3000 ↓ S
LAPTOP	\$1500 ↓ G	\$1500 ↓ G	\$2000 ↓ L	

MAX VALUE FOR 1 lb →

According to the last best estimate, you can fit the guitar into that 1 lb space, and that's worth \$1,500. So the real comparison is as follows:

3000\$ Stereo VERSUS (2000\$ Laptop + 1500\$ Guitar)?

You might have been wondering why you were calculating max values for smaller knapsacks. I hope now it makes sense! When you have space left over, you can use the answers to those subproblems to figure out what will fit in that space.

Here is the formula for this problem:

$$\begin{matrix} \text{ROW} & \text{COLUMN} \\ \downarrow & \downarrow \\ \text{CELL}[i][j] \end{matrix} = \max \left\{ \begin{array}{l} 1. \text{ THE PREVIOUS MAX (VALUE AT CELL } [i-1][j] \text{)} \\ \text{vs} \\ 2. \text{ VALUE OF CURRENT ITEM + VALUE OF THE REMAINING SPACE} \\ \quad \quad \quad \uparrow \\ \quad \quad \text{CELL}[i-1][j - \text{ITEM'S WEIGHT}] \end{array} \right.$$