Making Change

250H
Problem:

How many ways can you make change of $1.00 with pennies, nickels, dimes and quarters?
Easier Question: How many ways can you make change of $0.16 with pennies, nickels, dimes, quarters?
How many ways can you make change of $0.16 with pennies, nickels, dimes, quarters? 6 ways

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1d</td>
<td>+</td>
<td>1n</td>
<td>+ 1p</td>
<td>1(10) + 1(5) + 1(1)</td>
</tr>
<tr>
<td>1d</td>
<td>+</td>
<td></td>
<td>6p</td>
<td>1(10) + 6(1)</td>
</tr>
<tr>
<td>3n</td>
<td>+</td>
<td>1p</td>
<td></td>
<td>3(5) + 1(1)</td>
</tr>
<tr>
<td>2n</td>
<td>+</td>
<td>6p</td>
<td></td>
<td>2(5) + 6(1)</td>
</tr>
<tr>
<td>1n</td>
<td>+</td>
<td>11p</td>
<td></td>
<td>1(5) + 11(1)</td>
</tr>
<tr>
<td></td>
<td>+</td>
<td>16p</td>
<td></td>
<td>16(1)</td>
</tr>
</tbody>
</table>
How many ways can you make change of $1.00 with pennies, nickels, dimes and quarters?

- Discuss in Breakout Rooms
  - 10 mins
- Stop being antisocial and talk to your classmates
  - You can blame Bill for this one
  - I’d never be so evil as to make you talk to people
How many ways can you make change of $1.00 with pennies, nickels, dimes and quarters?

1. $a_n$ is the number of ways to make change of $n$ cents using pennies. $(\forall n)[a_n = 1]$. 
How many ways can you make change of $1.00 with pennies, nickels, dimes and quarters?

1. $a_n$ is the number of ways to make change of $n$ cents using pennies. ($\forall n)[a_n = 1]$.

2. $b_n$ is the number of ways to make change of $n$ cents using the first two coins (Pennies and s-sent coins). ($\forall n)[b_n = a_n + b_{n-s}]$. We use that ($\forall n \leq -1)[a_n = 0]$. 
How many ways can you make change of $1.00 with pennies, nickels, dimes and quarters?

1. $a_n$ is the number of ways to make change of $n$ cents using pennies. \((\forall n)[a_n = 1]\).

2. $b_n$ is the number of ways to make change of $n$ cents using the first two coins (Pennies and s-cent coins). \((\forall n)[b_n = a_n + b_{n-s}]\). We use that \((\forall n \leq -1)[a_n = 0]\).

3. $c_n$ is the number of ways to make change of $n$ cents using the first three coins (Pennies, s-cent coins, and t-cent coins). \((\forall n)[c_n = b_n + c_{n-t}]\).
How many ways can you make change of $1.00 with pennies, nickels, dimes and quarters?

1. $a_n$ is the number of ways to make change of $n$ cents using pennies. $(\forall n)[a_n = 1]$.

2. $b_n$ is the number of ways to make change of $n$ cents using the first two coins (Pennies and s-sent coins). $(\forall n)[b_n = a_n + b_{n-s}]$. We use that $(\forall n \leq -1)[a_n = 0]$.

3. $c_n$ is the number of ways to make change of $n$ cents using the first three coins (Pennies, s-cent coins, and t-cent coins). $(\forall n)[c_n = b_n + c_{n-t}]$.

4. $d_n$ is the number of ways to make change of $n$ cents using all four coins (pennies, s-cent coins, t-cent coins, and u-cent coins). $(\forall n)[d_n = c_n + d_{n-u}]$. 
How many ways can you make change of $1.00 with pennies, nickels, dimes and quarters?

Another way to ask this question: Compute $d_{100}$.

\[ d_n = c_n + d_{n-25} \]

\[ d_{100} = c_{100} + d_{75} \]

\[ d_{100} = c_{100} + c_{75} + d_{50} \]

\[ d_{100} = c_{100} + c_{75} + c_{50} + d_{25} \]

\[ d_{100} = c_{100} + c_{75} + c_{50} + c_{25} + d_0 \]

\[ d_{100} = c_{100} + c_{75} + c_{50} + c_{25} + c_0 \]
How many ways can you make change of $1.00 with pennies, nickels, dimes and quarters?

Another way to ask this question: Compute $d_{100}$.

$c_0 = 1$

$c_{25} = b_{25} + b_{15} + b_5$

$b_{25} = a_{25} + a_{20} + b_{15} = 6$

$b_{15} = a_{15} + a_{10} + b_5 = 4$

$b_5 = a_5 + b_0 = 2$

$c_{25} = b_{25} + b_{15} + b_5 = 6 + 4 + 2$
How many ways can you make change of $1.00 with pennies, nickels, dimes and quarters?

Another way to ask this question: Compute $d_{100}$.

$$d_{100} = c_{100} + c_{75} + c_{50} + c_{25} + c_0$$

$c_0 = 1$

$c_{25} = b_{25} + b_{15} + b_5 = 6 + 4 + 2 = 12$

$c_{50} = b_{50} + b_{40} + b_{30} + b_{20} + b_{10} + b_0 = 11 + 9 + 7 + 5 + 3 + 1 = 36$

$c_{75} = b_{75} + b_{65} + b_{55} + b_{45} + b_{35} + c_{25} = 16 + 14 + 12 + 10 + 8 + 12 = 72$

$c_{100} = b_{100} + b_{90} + b_{80} + b_{70} + b_{60} + c_{50} = 21 + 19 + 17 + 15 + 13 + 36 = 121$
How many ways can you make change of $1.00 with pennies, nickels, dimes and quarters? **242** ways

Another way to ask this question: Compute $d_{100}$.

$$d_{100} = c_{100} + c_{75} + c_{50} + c_{25} + c_0$$

$c_0 = 1$

$c_{25} = b_{25} + b_{15} + b_5 = 6 + 4 + 2 = 12$

$c_{50} = b_{50} + b_{40} + b_{30} + b_{20} + b_{10} + b_0 = 11 + 9 + 7 + 5 + 3 + 1 = 36$

$c_{75} = b_{75} + b_{65} + b_{55} + b_{45} + b_{35} + c_{25} = 16 + 14 + 12 + 10 + 8 + 12 = 72$

$c_{100} = b_{100} + b_{90} + b_{80} + b_{70} + b_{60} + c_{50} = 21 + 19 + 17 + 15 + 13 + 36 = 121$

$d_{100} = 1 + 12 + 36 + 72 + 121 = 242$
How can we code this?

How many ways can you make change of $0.16 with pennies, nickels, dimes, quarters? 6 ways

<table>
<thead>
<tr>
<th>1d</th>
<th>+</th>
<th>1n</th>
<th>+</th>
<th>1p</th>
<th>1(10) + 1(5) + 1(1)</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>1d</td>
<td>+</td>
<td>6p</td>
<td></td>
<td></td>
<td>1(10) + 6(1)</td>
<td>16</td>
</tr>
<tr>
<td>3n</td>
<td>+</td>
<td>1p</td>
<td></td>
<td></td>
<td>3(5) + 1(1)</td>
<td>16</td>
</tr>
<tr>
<td>2n</td>
<td>+</td>
<td>6p</td>
<td></td>
<td></td>
<td>2(5) + 6(1)</td>
<td>16</td>
</tr>
<tr>
<td>1n</td>
<td>+</td>
<td>11p</td>
<td></td>
<td></td>
<td>1(5) + 11(1)</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>+</td>
<td>16p</td>
<td></td>
<td></td>
<td>16(1)</td>
<td>16</td>
</tr>
</tbody>
</table>
How can we code this?

How many ways can you make change of $0.16 with pennies, nickels, dimes, quarters? 6 ways

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1d</td>
<td>+</td>
<td>1n</td>
<td>+</td>
<td>1p</td>
<td>1(10) + 1(5) + 1(1)</td>
</tr>
<tr>
<td>1d</td>
<td>+</td>
<td></td>
<td></td>
<td>6p</td>
<td>1(10) + 6(1)</td>
</tr>
<tr>
<td>3n</td>
<td>+</td>
<td>1p</td>
<td></td>
<td></td>
<td>3(5) + 1(1)</td>
</tr>
<tr>
<td>2n</td>
<td>+</td>
<td>6p</td>
<td></td>
<td></td>
<td>2(5) + 6(1)</td>
</tr>
<tr>
<td>1n</td>
<td>+</td>
<td>11p</td>
<td></td>
<td></td>
<td>1(5) + 11(1)</td>
</tr>
<tr>
<td></td>
<td>+</td>
<td>16p</td>
<td></td>
<td></td>
<td>16(1)</td>
</tr>
</tbody>
</table>

- Brute force
- Bad Recursion
- Good Recursion
How can we code this efficiently?

- Using Dynamic Programing!
- Dynamic Programming is used to optimize something that uses recursion
- We store the results of subproblems so we do not have to recompute the same thing later
- Ex: If we already computed $b_5$, we would waste time recomputing it
  - $b_5 = a_5 + b_0 = 2$
  - $b_{15} = a_{15} + a_{10} + [a_5 + b_0] = a_{15} + a_{10} + b_5 = a_{15} + a_{10} + 2$
makeChange(n):

S = [1, 5, 10, 25]

matrix = [n+1][4]

for i = 0 to 4
    table[0][i] = 1

for i = 1 to n+1
    for j = 0 to 4
        if i - S[j] >= 0 { x = matrix[i - S[j]][j] }
        else { x = 0 }
        if j >= 1 { y = table[i][j-1] }
        else { y = 0 }
        table[i][j] = x + y

return table[n][m-1]