CS 252

W, 8 May 2024
Fibonacci #s

\[ f(0) = 1 \]
\[ f(1) = 1 \]

\[ f(n) = f(n-1) + f(n-2) \]
for \( n > 1 \)

\[ \begin{array}{c|cccccccc}
  n & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
  f(n) & 1 & 1 & 2 & 3 & 5 & 8 & 13 & 21 \\
\end{array} \]

\[ f(100) \]

\[ f(98) \]
\[ f(97) \]

Solution: compute small values first + gradually build up big ones
\[ M(5) = \text{max revenue for a piece of wire of length 5} \]

12 \( \left( 3' + 1'' + 1'' \right) \left( 5'' \rightarrow 12 \text{\$} \right) \)
What are we optimizing?

Allow integer cuts

\[
\text{Maximize: } \sum_{i=1}^{k} p \cdot \text{length of piece } i
\]

Total revenue from length \( n \) of wire
Search space? Set of combinations of cuts.

\[ \text{# combinations of cuts? } \quad 2^5 = 2^{n-1} \]

\[ n = 6 \]

\[ 0 1 2 3 4 5 6 \]

\[ 0 2 3 8 9 12 13 \]

\[ m = 6 \quad \text{We're never dealing lengths } \geq m \]

\[ 1 \quad 3 \quad 2 \]

\[ \text{1 element of the search space} \]

\[ n-1, \text{ 5 places I could make cuts} \]
Problem: maximize \( \sum_{i} \epsilon_{p_{\text{piece } i}} \)

Search space

sets of cuts, \( 2^{n-1} \) possibilities
Let $M(n)$ = the maximum money obtainable by cutting up a wire of length $n$. 

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2</td>
<td>3</td>
<td>8</td>
<td>9</td>
<td>12</td>
<td>13</td>
</tr>
</tbody>
</table>
\[ M(n) = \max (p[i] + M(n-i)) \quad \text{for } i = 1, \ldots, n \]

\[ M(0) = 0 \]

Best you can get from 1st cut \( i \), then cut up the rest.
\[
M(n) = \max_{i=1,\ldots,n} (p[i] + M(n-i)) \\
M(0) = 0 \\
M(1) = \max(p[1] + M(0)) = \max(2 + 0)
\]
\[
M(n) = \max_{i=1, \ldots, n} (p[i] + M(n-i)) \quad \text{and} \quad M(0) = 0
\]

For example:

\[
M(2) = \max(p[2] + M(0), p[3] + M(0)) = \max(2+2, 3+0) = 4
\]
\[ M(k) = \max_{i=5, \ldots, k} \left( p[i] + M(k-i) \right) \]
$M(n) = \max_{i=1,\ldots,n} (p[i] + M(n-i))$

$M(0) = 0$


$= \max (2 + 4, 3 + 2, 8 + 0)$
$$M(n) = \max_{i=1,\ldots,n} (P[i] + M(n-i))$$

$$M(0) = 0$$
\[
M(n) = \max_{i=1, \ldots, n} (p[i] + M(n-i))
\]

\[
M(0) = 0
\]
\[ M(n) = \max_{1 \leq i \leq n} (P[i] + M(n-i)) \]

\[ M(0) = 0 \]
\[ M(n) = O(n^2) \]

\[ M(1) \sim 1 \text{ const time} \]

\[ M(2) \sim 2 \]

\[ M(3) \sim 3 \]

\[ M(n) \leq 1 + 2 + 3 + \ldots + n \]

\[ \frac{n(n+1)}{2} \]

\[ = O(n^2) \]