NANYANG TECHNOLOGICAL UNIVERSITY

SC2001/ CX2101: Algorithm Design and Analysis

Week 10

Huang Shell Ying

Dynamic Programming

- It is a problem solving paradigm
- Divide a problem into ***overlapping*** subproblems
- Do not compute the answer to the same subproblem more than once
- Dynamic programming is a powerful tool to solve optimization problems that satisfy the ***Principle of Optimality***
- Top-down approach: direct result of the recursive definition of the problem
- Bottom-up approach: compute the smallest problem first and build up the solutions in a table

Poll on the principle of optimality

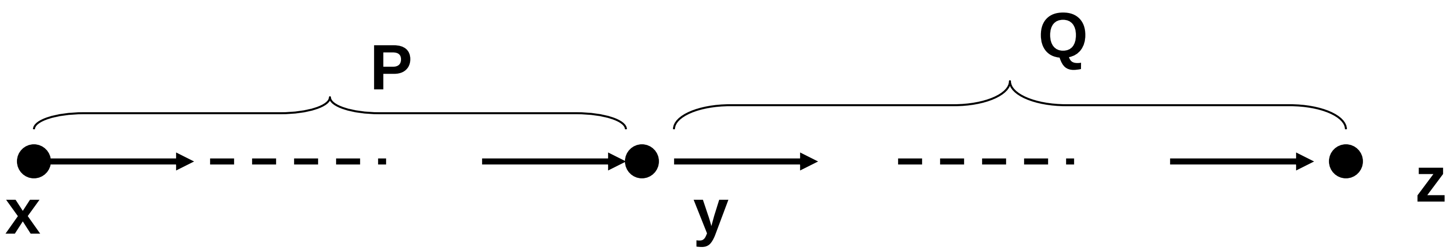
- A problem is said to satisfy the principle of optimality if the subsolutions of an optimal solution of the problem are themselves optimal solutions for their subproblems.

- Poll 1:

  Does the shortest path/distance problem satisfy the principle of optimality, yes or no.

- **Property of Shortest Path**

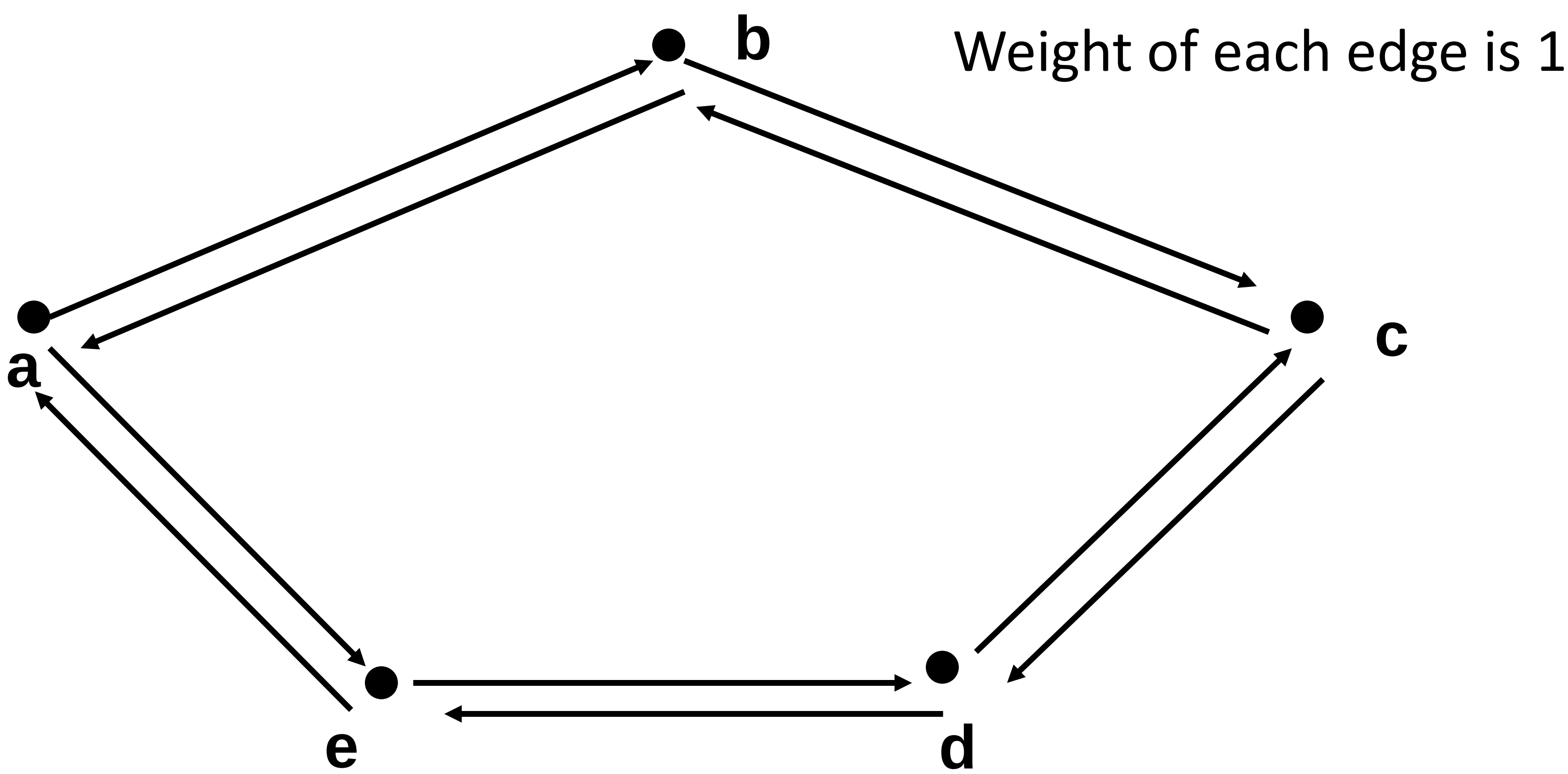
  In a weighted graph G, suppose that a shortest path from x to z consists of a path P from x to y followed by a path Q from y to z. Then P is a shortest path from x to y and Q is a shortest path from y to z.



Poll on the principle of optimality

- Poll 2:

  Does the longest path/distance problem satisfy the principle of optimality, yes or no.



The solution to the problem of longest path from a to d is a $\rightarrow$ b $\rightarrow$ c $\rightarrow$ d.

A subproblem to this problem is the longest path from a to b.

The longest path from a to b is a $\rightarrow$ e $\rightarrow$ d $\rightarrow$ c $\rightarrow$ b, not a $\rightarrow$ b.

Example: Making Change

Problem: A country has coins with denominations

$$1 = d_1 < d_2 < \cdots < d_k.$$

You want to make change for n cents, using the smallest number of coins.

Example: U.S. coins

$$d_1 = 1 \quad d_2 = 5 \quad d_3 = 10 \quad d_4 = 25$$

Change for 37 cents – 1 quarter, 1 dime, 2 pennies.

What is the algorithm?

Change in another system

Suppose

$$d_1 = 1 \quad d_2 = 4 \quad d_3 = 5 \quad d_4 = 10$$

- Change for 7 cents – 5,1,1
- Change for 8 cents – 4,4

What can we do?

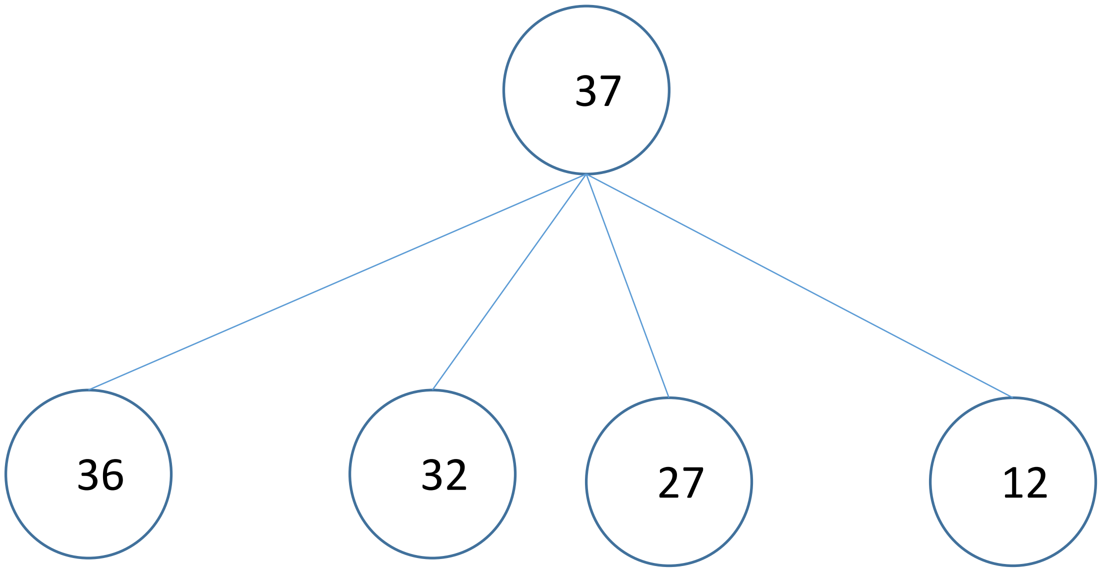
Solution

- Let $C[p]$ be the minimum number of coins needed to make change for p cents.
- Let x be the value of the first coin used in the optimal solution.
- Then $C[p] = 1 + C[p - x]$.

Problem: We don't know x.

We try all coins and take the minimum.

To change 37 cents where d = {1, 5, 10, 25}



(i) Give a recursive definition of the function change(n)

change(n) = 0 if n == 0

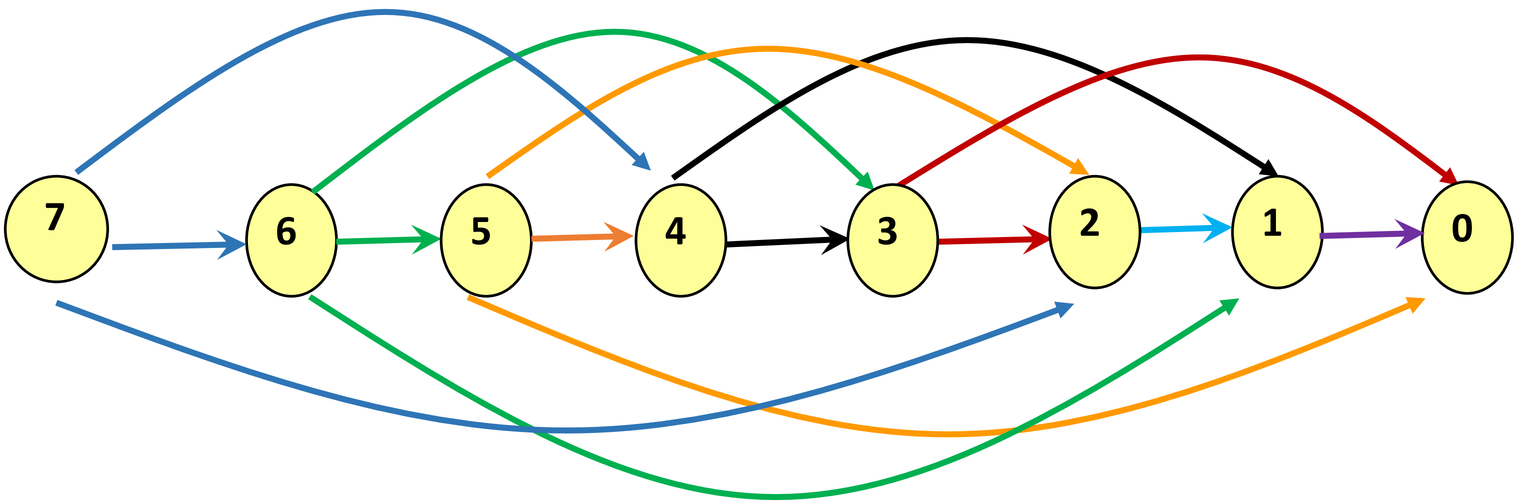
change(n) = ∞ if n < 0 // no need for this if smallest d_i is 1

change(n) = $\min_{i=1..m, d_i \le n} \left(1 + change(n - d_i)\right) \quad otherwise$

Does this problem satisfy the principle of optimality?

- The subproblems for change(n) are change(n – d_i), i = 1..m, $d_i \le n$.
- For the optimal solution of change(n), the solution to change(n – d_i) is the optimal solution. Otherwise, the solution to change(n) cannot be optimal.

(ii) Draw the subproblem graph for change(7) where d = {1, 3, 5}.



(iii) Design a dynamic programming algorithm of change(n) using the bottom-up approach

```
Change(n)   {
        C[0] = 0;
        For j = 1 to n   {
              min = 999999;
              For k = 1 to m
                    // k = 0 to m-1 if denominations are in d[0..m-1]
                       If (d[k] <= j and min > 1+ C[j-d[k]])
                              min = 1+ C[j-d[k]];
              C[j] = min;    }
     Return C[n];
}
```

Complexity: O(mn)

Example: Change(7)

d

1	2	5

m = 3

C

0	1	2	3	4	5	6	7
0	1	1	2	2	1	2	2

C[0] = 0
C[1] = 1 + C[0] = 1
C[2] = min(1+C[1], 1+C[0])= 1 // min between one 1¢ + C[1] and one 2¢ + C[0]
C[3] = min(1+C[2], 1+C[1])= 2 // min between one 1¢ + C[2] and one 2¢ + C[1]
C[4] = min(1+C[3], 1+C[2])= 2 // min between one 1¢ + C[3] and one 2¢ + C[2]
C[5] = min(1+C[4], 1+C[3], 1+C[0])= 1
// min among one 1¢ + C[4], one 2¢ + C[3] and one 5¢ + C[0]
C[6] = min(1+C[5], 1+C[4], 1+C[1])= 2
// min among one 1¢ + C[5], one 2¢ + C[4] and one 5¢ + C[1]
C[7] = min(1+C[6], 1+C[5], 1+C[2])= 2
// min among one 1¢ + C[6], one 2¢ + C[5] and one 5¢ + C[2]