Algorithms Theory

14 – Dynamic Programming (4)

Edit distance
Approximate string matching
Sequence alignment

P.D. Dr. Alexander Souza

Dynamic programming

- Algorithm design technique, often applied to optimization problems
- Generally suitable for recursive approaches, when solutions to subproblems are required repeatedly.
- Approach: maintain a table of subproblem solutions
- Advantage: improved running time; often polynomial instead of exponential

Two different approaches

Bottom-up:

+ the table is maintained in an efficient way, time saving
+ subproblems are solved in a special, optimized order, space saving
- extensive rewriting of the original program code is necessary
- possibly, unnecessary subproblems are solved

Top-down: (memoization)

+ only slight modifications in the original program code are necessary
+ only those subproblems definitely required are solved
- separate table management is time consuming
- table size is often suboptimal

String matching problems

Transform string A into string B with replacements, inserts, deletes

Edit distance

For two given strings *A* and *B*, efficiently compute the edit distance *D(A,B)* as well as a minimum sequence of edit operations that transforms *A* into *B*.

m	a	-	t	h	e	m	-	-	a	t	i	c	i	a	n
m	u	l	t	i	p	l	i	c	a	t	i	o	-	-	n

String matching problems

Approximate string matching

For a given text *T*, a pattern *P* and a distance *d*, find all substrings *P´* of *T* with $D(P,P´) \leq d$.

Sequence alignment

Find optimal alignments of DNA sequences.

G A G C A - C T T G G A T T C T C G G
- - - C A C G T G G - - - - - - - - -