Knowledge-based systems for bioinformatics – 1MB602
Professor Jan Komorowski
Exam
2004-12-16

Contact under exam: Jakub Orzechowski Westholm 070 22 44 026

The exam consists of three parts: Programming In Scheme, Programming In Prolog, Foundations of Knowledge Based Systems. Grading 3 -- 50%, 4 – 70%, 5 – 85% of points.

PART I – SCHEME (30%)

TASK 1. (10%) For each of the following expressions, what must `f` be in order for the SCHEME evaluation of the expression to not cause an error? For each expression, give a SCHEME definition of `f` such that evaluating the expression will not cause an error, and say what the expression's value will be, given your definition.

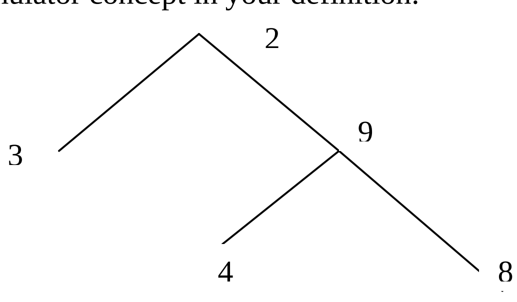
I. `f`
II. `(f)`
III. `(f 3)`
IV. `((f))`
V. `(((f)) 3)`

TASK 2. (10%) Given a list of length *k*, define a SCHEME procedure that returns a sub-list that starts at position *i* and ends at position *j*, where $0 \leq i \leq j < k$ For example,

`(sublist 2 6 '(b o o k k e e p i n g))` returns
`(o k k e e)`
and
`(sublist 2 6 '(b i o (i n f o r) m a t i c s))` returns
`(o (i n f o r) m a t)`

TASK 3. (10%) Sum all odd integers in a binary tree. Make your design using the dispatch concept in SCHEME.

I. Define an appropriate data structure
II. Use the accumulator concept in your definition.



For the above tree the answer should be 12.

PART II – PROLOG (20%)

TASK 4 (10%) Find the bugs in the following definition of `qsort/2` and in the auxiliary procedures where the intended meaning of the definitions is the well-known quick-sort algorithm.

```
qsort( [A|As], Sorted) :-
     partition( As, A, Smaller, Larger),
     qsort( Smaller, SS),
     qsort( Larger, LS),
     append( Smaller, Larger, Sorted).

partition( [A|As], Key, Smaller, Larger):-
     A < Key, partition( As, Key, Smaller).

partition( [A|As], Key, Smaller, Larger):-
     A > Key, partition(As, Key, Larger).

append([], [],[]).
append([A|As], Bs, Cs) :-
     append( As, Bs, Cs).
```

TASK 5 (10%) Given procedure `member/2` in PROLOG:

```
member( Elt, [Elt|_]).
member( Elt, [X|Rest]) :- member( Elt, Rest).
```

I. Give the SLD-tree for the following query
   `?- member( jacke, [jan, jacke, robin, jacke]).`

II. Modify your definition using cut so that there is only one answer to the query. Show how the SLD-tree is modified after adding the cut.

III. Modify your definition without using cut so that there is only one answer to the query. Show that with the SLD-tree.

PART III FOUNDATIONS OF KNOWLEDGE SYSTEMS (50%)

TASK 6. (5%) Using truth tables show which of the following formulae is a) satsifiable, b) true, c) contradictory.

I. $(p \vee q) \rightarrow (p \vee r)$
II. $(\neg p \vee \neg q) \rightarrow (q \wedge p)$
III. $(p \wedge (q \vee r)) \leftrightarrow ((p \wedge q) \vee (p \wedge r))$

TASK 7. (10%) Given that N denotes an arbitrary nucleotide base, define a recursive procedure (either in PROLOG or in SCHEME) that simulates a finite state automaton that recognizes the following DNA sequences represented as lists: `GNN(A|T)(G|C)A`. Structure and comment your design.

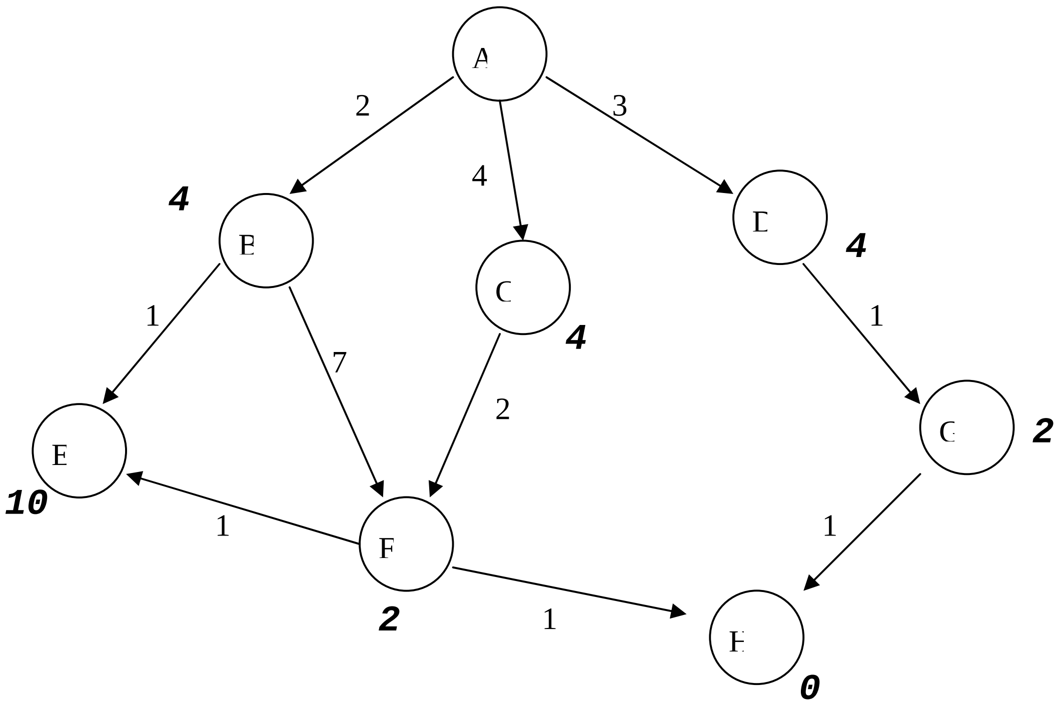
Examples:
```
GTTAGA
Yes
GGTTCA
Yes
GTTAAA
No
```

TASK 8. (15%) Define a DCG grammar that accepts the following inputs:

Condition-invariant regulators bind the promoters.
Condition-expanded regulators bind some promoters.
Condition-altered regulators bind two promoters.

TASK 9 (20%) Consider the following graph where the numbers by the edges define the cost of following that edge and the numbers by the node give the expected cost of going from that node to the goal. The initial state is A and the goal state is H.

I. Using Depth First Search (DFS), Breadth First Search (BFS) and the Greedy Search (GS) algorithms show which path every algorithm finds and in which order the nodes are visited.
II. Define a heuristic function for the A* algorithm so that it finds a cheaper path than the previous three algorithms.



END OF EXAM