Programming languages – ProLog

Prolog homework (2024/25)

T. Goluch

1. Implement a `sort(list,sorted)` predicate, that returns a not ascending sorted list. Apply the chosen one sorting algorithm e.g. selection[1], insertion[2], buble[3], merge[4], quick[5] etc ... **(2 pkt.)**.

   Please do not use built-in predicates: `reverse`, `permutation` and `append`.

2. Implement `is_graphic(list,response)` predicate, stating whether the list creates a graphic sequence[6]. Use the sort predicate from the previous task. (**3 pkt.**).

   Helpful Resources:

   - http://szhorvat.net/pelican/hh-connected-graphs.html?fbclid=IwAR3E1PiPcxK5ShLKNJNqy1rLmbRQrkqD7G7F7mdps6JVxlv3-jlu9nnLEgA – A simple algorithm for realizing a degree sequence as a connected graph
   - https://supremus.sk/math/is_a_degree_sequence_graphic/index_en.php – Online Calculator - Can a degree sequence form a simple graph?

3. Implement `is_connected(list,response)` predicate, stating whether the list of vertex degrees creates a graphic sequence from which a connected graph can be created. Use the predicate from the previous task to check whether a graph can be created from vertex degrees list. (**2 pkt.**).

```
: is_connected( [1,0,1], RES )        % not connected graph
RES = N

: is_connected( [1,1,1], RES )        % not graphic series
RES = N

: is_connected( [1,1,1,1], RES )      % not connected graph
RES = N

: is_connected( [1,2,2,1,2], RES )         % connected graph
RES = Y
```

[1] https://en.wikipedia.org/wiki/Selection_sort
[2] https://en.wikipedia.org/wiki/Insertion_sort
[3] https://en.wikipedia.org/wiki/Bubble_sort
[4] https://en.wikipedia.org/wiki/Merge_sort
[5] https://en.wikipedia.org/wiki/Quicksort
[6] https://mrpandey.github.io/d3graphTheory/unit.html?graphic-sequence

```
: is_connected( [3,3,3,0,3], RES )          % not connected
graph
RES = N
```

In the penultimate case, we can create a path P_5 or a complete cycle/graph C_3/ and a path/complete graph P_2/K_2. Thus, a connected graph can be created.

The last case is an example of a disconnected graph in which the number of edges is greater than the number of vertices.

Helpful Resources:

- https://math.stackexchange.com/questions/732303/degree-sequence-of-connected-graphs – Degree sequence of connected graphs,
-
- http://szhorvat.net/pelican/hh-connected-graphs.html – A simple algorithm for realizing a degree sequence as a connected graph,

- https://arxiv.org/pdf/2009.03747.pdf – Connectedness matters: Construction and exact random sampling of connected networks, rozdział: Building a single connected realization of a degree sequence.