A PRIORITY QUEUE FOR A SIMPLIFIED AGENDA MANAGER IN A RULE-BASED EXPERT SYSTEM SHELL

CS-5381- Analysis Of Algorithm
Fall – 2017

AMITSINGH PARDESHI

Contents

OBJECTIVE:

To create an agenda manager application for building a priority queue system implemented by using a heap. This agenda manager application reads the given input file and generates rule based inference engine. The engine spits out the rule with the highest priority cycle by cycle while reading the subsequent line in the given input file.

INTERFACES:

- buildQueue(listOfRules) – Builds the initial heap using the list of rules.
- insertHeap(rule) - inserts the new rule at the right position in the heap.
- extractmax() – Returns the rule with the highest priority.
- removeRule(index) – remove the rule at given index.
- maxHeapify(index) – performs heapify at given index.

PROGRAM LOGIC:

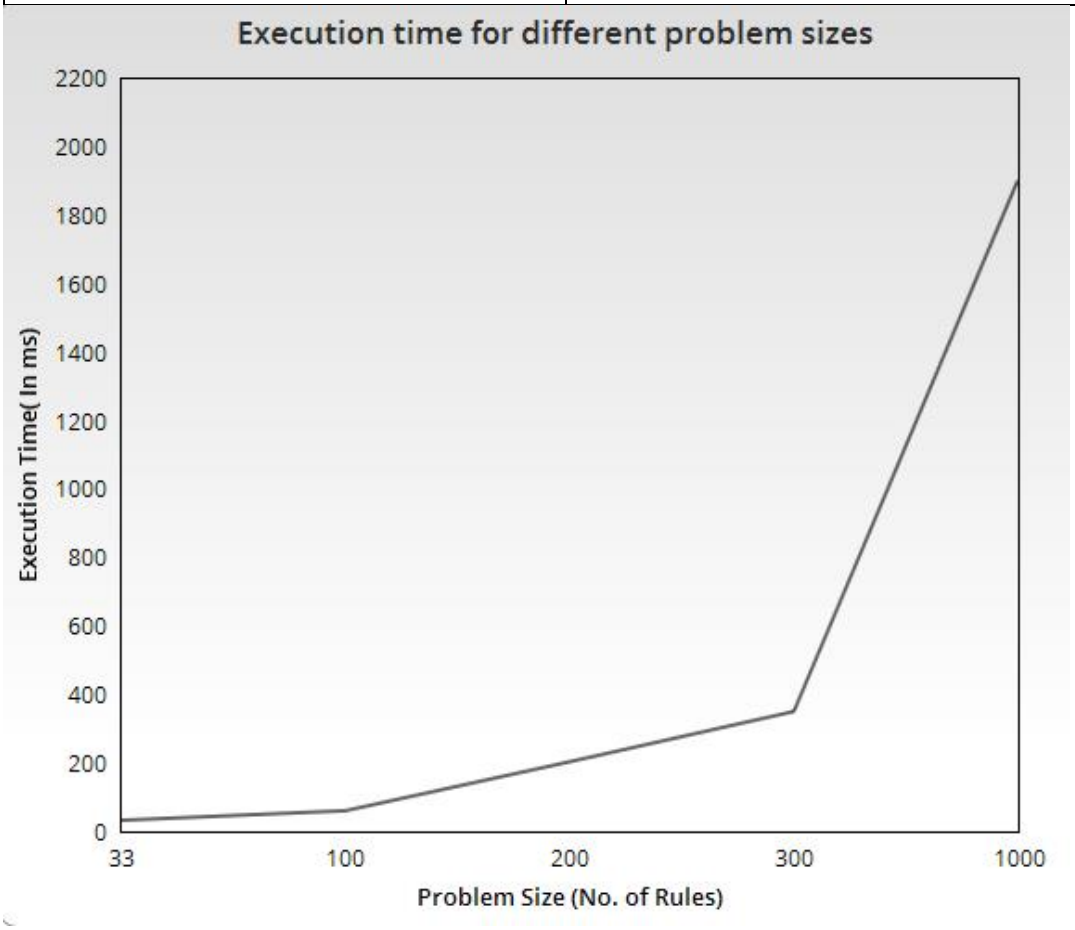
- Read the input file line by line.
- Split the line to get each rule.
- Create Rule object which has two fields (name and priority).
- Insert the rule object into the list.
- If read line is the first line then call **buildQueue**() method by passing the list above which will create the initial heap.
- Else if the read line is not the first one then create rule object and call **insertHeap**() method by passing the rule object created.
- The rule object will be inserted at the right position.
- Call **extractmax**() method which will return the max element in the heap.
- Print the returned max element (Executed rule) and delete it.
- Also print the Activated Rules in the same cycle.
- Continue the process till all the lines are read or the number of cycles reached up to 30.

SOURCE CODE:
https://github.com/pardeshiamitsingh/AgendaManger/

TEST CASE RESULTS:
https://github.com/pardeshiamitsingh/AgendaManger/tree/master/results

GRAPH:

Time Elapsed (In ms)	Problem Size (No. of Rules)
32	33
60	100
203	200
350	300
1904	1000



The worst-case time complexity for priority queue is O(nlgn) which almost looks like above graph.