Leonard Mihalcea

Lab 6 - Parallelizing techniques

Due: week 11.

Goal

The goal of this lab is to implement a simple but non-trivial parallel algorithm.

Requirement

Solve the problem below:

Given a directed graph, find a Hamiltonean cycle, if one exists. Use multiple threads to parallelize the search.

The documentation will describe:

- the algorithms,
- the synchronization used in the parallelized variants,
- the performance measurements

Computer Specification

CPU: Intel Core i5-7300HQ
RAM: 16GB 2

Implementation

Algorithm - Backtracking

- Generate all possible paths starting from each node, and check each one if it is a Hamiltonian Cycle
- In the parallelized we have on thread executing the backtrack for one node

For parallelization I used the Thread class along with the ArrayBlockingQueue

Tests

Level 0

Regular linear algorithm: 2ms
Binary tree algorithm in parallel: 0ms

Level 50

Regular linear algorithm: 82ms
Binary tree algorithm in parallel: 40ms

Level 100

Regular linear algorithm: 609ms
Binary tree algorithm in parallel: 299ms