Coursework 1

Released: November 13, 2015
Deadline: January 13, 2016
Weight: 25 %

1. Introduction

The coursework is an independent study of **one** of the data structures listed in Section 2. The following points should be covered:

1. Informal introduction to the data structure, and a description of its main applications.
2. A description of the main ideas behind the analysis of the data structure.
3. An implementation of the main operations on the data structure. Any programming language can be used, but the choice has to be justified.
4. An empirical evaluation of the performance of the data structure on a typical application with a "real-world" data set of significant size.

The coursework report should be submitted via Moodle.

- ▷ `report.pdf` a 5-8 page report covering the points above
- ▷ `src.zip` the source code for the implementation. The zipped directory should contain a `Makefile` which allows the source code to be built on the School's Linux servers without any additional software installed
- ▷ `data.zip` the data files used for the experiments, including a shell script `run.sh` that starts the experiments

The following marking scheme will be used

- ▷ Introduction 10 %
- ▷ Analysis 20 %
- ▷ Implementation 20 %
- ▷ Empirical evaluation 20 %
- ▷ Quality of writing 10 %
- ▷ Group B data structure: 20 %

2. Data structures

	Group A	Group B
0	B-trees	Bloom filters
1	AVL trees	Cuckoo hashing
2	Skip lists	Compact suffix trees
3	Treaps	Finger trees
4	Ropes	Tango trees

Students can choose Group A or Group B. The data structures in Group B are more complex and may require reading more advanced texts. The allocation of data structure within the chosen group is made by the Haskell program below.

```
-- G54AAD: Coursework allocation 2015/2016

datastructures =
    [("B-trees", "Bloom filters"),("AVL trees", "Cuckoo hashing"),
     ("Skip lists", "Compact suffix trees"), ("Treaps", "Finger trees"),
     ("Ropes", "Tango trees")]

students =
    ["Toghrul Abilli", "Edwin Aldridge", "Kejia Chen", "Tianxin Deng",
     "Xiaoyu Deng", "Benjamin Michael Easton", "Stephen Ford",
     "Barnabas Forgo", "Daniel Philip Freeman", "Samuel Thomas Frier",
     "Steven John Frost", "Luke Geeson", "Tom Grover", "Elliot Halliwell",
     "Rebeen Ali Hamad", "Fang Hao", "Lanxiao Hu", "Osman Ali Sadek Ibrahim",
     "Rui Jiao", "Yuxuan Jin", "Tayyibe Kansoy", "Eliaas Khoury",
     "Ian Ashley Knight", "Qingyuandi Lin", "Casper Oakley", "Miles Plaskett",
     "Zimu Qin", "Dong Sheng", "Jack Sparham", "Xu Sun",
     "Yik Him Tang", "Tung Vu Manh", "Matthew Shane Wade", "Junbin Wang",
     "Xin Wei", "Luying Yan", "Huimin Yao", "Jie Yu",
     "Zongzhe Yuan", "Haozhe Zhang", "Duyang Zhou", "Wenjue Zhu"]

pair ds [] = []
pair [] ss = pair datastructures ss
pair (d:ds) (s:ss) = (d,s):(pair ds ss)

allocation = pair datastructures students
```