CmpE321 - Project 1

Gökçe Uludoğan (gokce.uludogan@boun.edu.tr)
Rıza Özçelik (riza.ozcelik@boun.edu.tr)

Deadline: 11th March 2019, Monday, 09:00

1 Project Description

In this project, you will design a storage manager system. To do so, you will design a system catalogue to store the metadata as well as data storage units; pages and records. Your design must support following operations:

DDL Operations

- Create a type

- Delete a type

- List all types

DML Operations:

- Create a record

- Delete a record

- Search for a record (by primary key)

- List all records of a type

During the design process, you will have to make certain decisions and assumptions. The decisions include, but are not limited to:

- Page size (Something between 1KB and 2KBs is optimal)

- File size

- What information to store in your page headers and record headers

- Max number of fields a type can have

- Max length of a type name

- Max length of a field name

For the assumptions, you can assume that user always enters valid input and fields are always integers, though field and type names can be alphanumeric. Moreover, a disk manager already exists that is able to fetch the necessary pages when addressed. You can make further assumptions, as long as they do not conflict with the base constraints and you explained them in your report clearly.

Constraints:

- Your data *must* be organized in pages and pages must contain records. So, you *must* clearly explain your page and record structure in your report.

- You *are not allowed to* store all pages in the same file and a file *must* contain multiple pages. This means that your system *must* be able to create new files as storage manager grows. Moreover, when a file becomes free due to deletions, that file *must* be deleted.

- Though a file contains multiple pages, your system *must not* load the whole file to RAM when it is needed. Instead, it *must* read a file page by page.

All in all, you will propose a storage manager that obeys the abovementioned rules under certain assumptions and write a report explaining it in detail.

2 Report & Grading

You are expected to submit a report *written in LaTeX* that contains the sections below. Corresponding points are specified in parentheses.[1]

1. **Title Page:** Write course name, semester, assignment title, your name and student number. *(5 pts)*

2. **Introduction:** Briefly describe the project in your own words. *(10 pts)*

3. **Assumptions & Constraints:** Clearly specify your assumptions and constraints of the system in an itemized or tabular format. *(25 pts)*

4. **Storage Structures:** Explain your system catalog, page design, page header, record header etc. with tables/diagrams/figures. *(25 pts)*

5. **Operations:** Write your DDL and DML operations in pseudocode. Beware of referring to corresponding storage structures when needed. *(25 pts)*

6. **Conclusions & Assessment:** Evaluate your design, considering its ups and downs. State what is missing and what can be added and how. *(10 pts)*

---

[1]Grading scheme is tentative and subject to change when it necessitates.

3 Submission

The submissions will be through moodle. Submit a single pdf file named with your student number (e.g. 2015400XXX.pdf). If your pdf size is over moodle upload limit, submit a link (drive, dropbox etc.) that can be used to download your report.
Note that your reports will be inspected for plagiarism with previous years' reports as well as this year's. Any sign of plagiarism will be penalized.