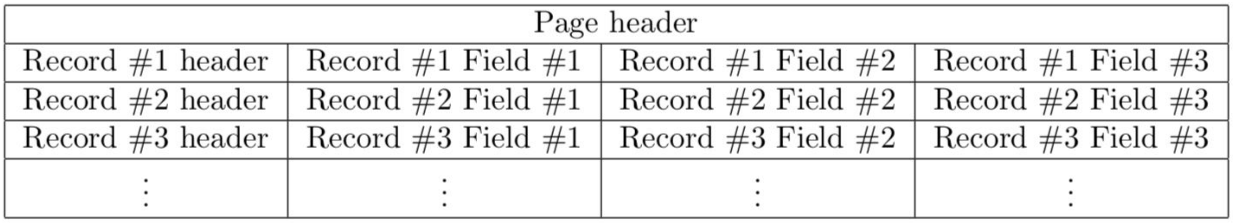
CMPE 321 - Assignment I
Designing Storage Manager System

Due date 12.03.2018 23:59

In this assignment, a storage manager system will be designed. The design shall have a system catalog which stores metadata, and multiple data files that store the actual data. The system needs to support common data handling operations of DDL (Data Definition Language) and DML (Data Manipulation Language).

1. Introduction

Data on disk is stored in data structures called pages. Each page begins with a page header which is followed by records. Each record, in turn, begins with a record header followed by the fields of that record, see Table 1.



Page header			
Record #1 header	Record #1 Field #1	Record #1 Field #2	Record #1 Field #3
Record #2 header	Record #2 Field #1	Record #2 Field #2	Record #2 Field #3
Record #3 header	Record #3 Field #1	Record #3 Field #2	Record #3 Field #3
⋮	⋮	⋮	⋮

Table 1: Page structure

A storage manager system keeps validity, integrity of the data; with the end service of definition, manipulation, access, and so on. A system catalog, as a subsystem, interprets metadata and helps the storage manager to map/extract meaning from raw disk data/pages to the records, fields levels.

2. Description

As this system is given as a general concept storage manager, most specifications of this design will be decided by you. You also will need to determine some constraints and assumptions. These include, but are not limited to:

1. Page size (Something between 1KB and 2KBs is optimal)
2. File size
3. What information to store in your page headers and record headers
4. Max number of fields a type can have
5. Max length of a type name
6. Max length of a field name

All fields shall be of type integer. Type and field names, however, shall be alphanumeric.

You can choose to keep all types in the same page (mixed style) or to allocate a different page for each type (separate style). The number of types in the system may vary over time (See DDL Operations below).
The user can run following 7 DDL & DML operations:

DDL Operations

1. Create a type
2. Delete a type
3. List all types

DML Operations

1. Create a record
2. Delete a record
3. Search for a record (by primary key)
4. List all records of a type

Make reference to various parts of storage structures you defined (such as page header, record header etc.) in specifying your algorithms.
Do not create a new file for a new page. A file should be able to hold multiple pages, but your algorithms are supposed to read/process a file page by page.
For this assignment, you can assume that the disk manager fetches a page when the page address is supplied.
Do not employ any error checking. Try to make your design reasonably efficient but do not include extra storage structures for performance improvements. Assume that the user always inputs valid data.

3. Report

You are supposed to submit a report that have the following sections:
1. Title Page (Course name, semester, assignment title, your name and student number)
2. Introduction (Briefly describe your project, use your own words)
3. Assumptions & Constraints (e.g. max number of fields a record can have, max length for a field name. Use tabular representations or subsections for your assumptions and constraints)
4. Data Structures (Explain your system catalog, page design, page header, record header, etc. with diagrams)
5. Operations (Algorithms written in pseudocode)
6. Conclusions & Assessment (Evaluate your design, considering its ups and downs)

4. Submission

You will use moodle to submit the assignment. Please submit a pdf or zipped file with the name format: CMPE321_#AssignmentNo_#Name_#Surname_#StudentNo.pdf
(e.g.: CMPE321_1_Bob_Alan_2018100001.pdf)
**Please do not use any special or non-english characters.*
If the submission package does not fit to moodle, submit a dropbox/google drive etc. link.