CS232 Lab2
sp2015
streller

PROBLEM BACKGROUND

A table is an unordered group of "records" or key-value pairs each of which has a designated field called the key field. Records within the table are identified by the value of the key field. The table is an appropriate ADT for information retrieval systems in which individual records must be accessed frequently (and quickly) but the entire collection of records must never be processed sequentially. Synonymous terms are lookup table and map. The latter stresses that a table is really a mapping. A simple example of a table is the dictionary: the word is the key and the definition is the value. Read the enclosed TableHandout.pdf for more.

PROBLEM STATEMENT:

Design a table ADT
Specifically write a table class.
It must be generic (i.e. templated) and contain all the necessary attributes and actions (data members and member functions) for the table class to behave as expected.

DELIVERABLES:

hard: table.h, table.t, table.cpp, and driver.cpp (driver.cpp demonstrates that your table performs)
stapled of course

soft in a zipped file, called CS232_Lab2_yourLastName,
sent to streller@ecc.edu with the subject CS232_Lab2
all above mentioned source code

due ***before*** the start next lab period (3 february 2015)

grading will be based on three aspects

1. table class design consisting of all necessary components
2. table class compiling
3. driver demonstrating table class functionality
4. demonstration on 3 February in Lab period

NOTE:

All submitted code for this course must use the separation model for code layout (ie .h, .cpp and where applicable .t files) See interfaces.pdf under the cs232 notes section of the course webpage

Code not in this format will receive an automatic 20 point deduction

Also all code must have the appropriate file comment header and function comment header as spelled out in the documentation standards handbook.

Acceptable alternatives are

```
/*********************************************************************************************
*
*   File name:     xxx.extention
*
*   This file contains ....
*    Or The purpose is blah, blah, etc ...
*
*   Programmer:
*
*   Date Written:
*
*   Date Last Revised:
*
*********************************************************************************************/
```

And

```
/*****************************************************************************************
*
*   Function Name:
*
*   Purpose:
*
*
*   Input Parameters:
*
*   Output parameters:
*
*   Return Value:
*
*****************************************************************************************/
```

Note the order of presentation. This MUST be followed

Code not in this format will receive an automatic 20 point deduction