# CSI2131, Winter 2001
# Assignment 1
# Due on Monday, January 29, 11:00am

Worth: 7.5%     Marks out of 100

## 1 Program: Internal Sorting 65 marks

In this program, you are going to deal with a file that stores student passwords. Your program is going to sort the file by student number. The objective of this exercise is to get you familiar with the C++ language and compiler.

You will write a C++ program to sort a student password file. Your program will perform "internal sorting", i.e. sorting the file by bringing the whole file into main memory.

The user will provide input and output file names. The input file (that you read) and the output file (that you create) are going to be text files containing fixed length records with fixed length fields described below:

| **Field name** | **Field Type and Length** | **Note** |
|---|---|---|
| StudentNumber | 7 characters (numeric) | unique key |
| Password | 6 characters | |
| StudentName | 20 characters | |

After each record, a "newline" character is present, so that the file is easily readable by a human being (like your TA :-)) ).
Your program should do the following:

- Read the input file records, record by record, storing them in main memory.
- Sort the data in main memory (by increasing student number) using an efficient sorting algorithm of your choice (any $O(n\ log\ n)$ algorithm or Quicksort).
- Write the records in sorted order into an output file.

Create appropriate C++ classes in order to perform all these tasks. (keep in mind that part of this program will be re-used in Assignment 2). **Note:** In this assignment, you will deal with this file in a sequential way. In Assignment

2, you will re-use your code in order to allow the initial password to be changed, which will then involve direct access and binary search.

You will be evaluated for correctness, efficiency and programming style. For instance, an efficiency issue to consider is the following. Suppose you have stored the data in an array of `Student` objects. Explicitly sorting the whole array would involve a lot of bytes to be moved around when you swap two objects. Can you think of a more efficient and elegant way of doing this?

**IMPORTANT:** You should follow: (1) the general assignment standards; (2) the standards specific to this assignment. Both will be posted in the web page at a later date, but you can start your assignment before that!

## 2 Problem A: Disks 15 marks

The IBM 3350 disk drive uses block addressing. The two following subblock organizations are available:

Count-data, where the extra space used by count subblock and interblock gaps is equivalent to 185 bytes; and

Count-key-data, where the extra space used by the count and key subblocks and accompanying gaps is equivalent to 267 bytes, plus the key size.

An IBM 3350 has 19,069 usable bytes per track, 30 tracks per cylinder, and 555 cylinders per drive. Suppose you have a file with 350,000 80-byte records that you want to store on an IBM 3350 drive. Answer the following questions. Unless otherwise directed, assume that the blocking factor is 10 and that the count-data subblock organization is used. Explain all your answers carefully.

**a.** How many blocks can be stored on one track? How many records?

**b.** How many blocks can be stored on one track if the count-key-data subblock organization is used and the key size is 13 bytes?

**c.** How many cylinders are required to hold the file (blocking factor 10 and count-data format)? How much space will go unused due to internal track fragmentation?

**d.** If the file were stored on contiguous cylinders and if there were no interference from other processes using the disk drive, the average seek time for a random access of the file would be about 12 msec. Use this rate to compute the average time needed to access one record randomly. Assume that the disk rotates at 3600 r.p.m. **Hint:** On average, the rotational delay is half a revolution.

# 3 Problem B: Magnetic Tapes 20 marks

Consider a 1,000,000- record file where each record contains 100 bytes. The file is to be backed up on 2,400-foot reels of 6250-bpi tape with 0.3-inch interblock gaps. Tape speed is 200 inches per second. Answer the following questions. Explain all your answers carefully.

**a.** Show that only one tape would be required to back up the file if a blocking factor of 50 is used.

**b.** If a blocking factor of 50 is used, how many extra records could be accommodated on a 2,400-foot tape?

**c.** What is the effective recording density when a blocking factor of 50 is used?

**d.** How large does the blocking factor have to be to achieve the maximum effective recording density? What negative results can result from increasing the blocking factor? (Note: an I/O buffer large enough to hold a block must be allocated)

**e.** What would be the minimum blocking factor required to fit the file onto the tape?

**f.** If a blocking factor of 50 is used, how long would it take to read one block, including the gap? What would the effective transmission rate be? How long would it take to read the entire file?