

EEN318  
Gentle Fluffy Safe Test  
17th October 2016

Behave nicely. Puppies are still forbidden.

Who are you ?

---

What is your student number?

---

“I have neither given nor accepted any aid in this examination”: Sign if true.

---

---

do not write anything below that line.

Question	1	2	3	4
out of	33%	33%	33%	1%
you got				

1.

a.

Explain the difference between  
closed hashing = open addressing  
and open hashing = closed addressing.

There is just one point for being able to say which is which, so don't be too worried about mixing them up.

What are their relative advantages and disadvantages?

b.

Give a C++ implementation of a hash table of dictionary entries.  
A dictionary entry consists of a word and its definition, both strings.  
You can use either open or closed hashing.

Include a constructor,  
a method for adding a new entry,  
a method for looking up a word and returning its definition,  
and a destructor.



## 2.

An non-text “binary” file called `StolenNumbers.dat` contains a lot of personal details.

The file consists of a large number records for people, each record consists of 64 bytes, exactly as follows.

- 1 thirty-two bit int for birth date

- 20 characters for First Name (padded with spaces if necessary)

- 20 characters for Last Name (padded with spaces if necessary)

- 1 thirty-two bit int for social security number

- 16 characters for Primary Credit-card number

So, if the file has the records for 1,000 people, it will be exactly 64,000 bytes long.

Write a C++ program that uses the standard `<iostream>`, `<fstream>` and `<string>` functionality to sort this file so that the names appear in alphabetical order, primarily by last name, but when last names are the same, use first names.

Use any sorting algorithm you like.

This sort should be performed on disc. DO NOT read the file into an array and sort it there.

Imagine you are programming a very small microprocessor that has very limited memory capacity, but is connected to a normal capacity disc drive. The processor can only store a very few records in memory at once.

The sorting is not expected to be very fast.



3.

a.

What is a priority queue?

What are the operations that every priority queue must be able to perform?

Briefly describe (just a couple of sentences) a possible implementation of a priority queue.

b.

One simple representation for a graph is to use an *adjacency matrix*: each of the  $N$  nodes is given a unique number in the range 0 to  $N-1$  to identify it. A large two dimensional array  $A$  with  $N$  rows and  $N$  columns is created so that

$A[x][y]$  stores the cost of travelling directly from node  $x$  to node  $y$ .

if  $A[x][y]$  is zero, then there is no direct connection from  $x$  to  $y$ .

$A[x][y]$  does not need to equal  $A[y][x]$  - there can be one-way links.

Assume that a priority queue exactly as you described for part A has been implemented, and you can use it. Also assume that an adjacency matrix as just described has been created for an  $N$  node graph, and you can use it too.

Write a function that finds the length of the shortest path (or the cost of the cheapest path) between nodes  $S$  and  $D$ , which are provided as parameters.



4.

Multiple Choice:

- a. Are the Andes a special kind of hat?
- b. What about the Himalayas?
- c. What came first, the spider or the spider egg?
- d. Why did the chicken? Answer with an illustration.