

OLLSCOIL NA hÉIREANN, GAILLIMH  
NATIONAL UNIVERSITY OF IRELAND, GALWAY

---

SEMESTER 2 EXAMINATIONS 2003/2004

---

THIRD YEAR COMPUTER SCIENCE

---

ELEMENTS OF SOFTWARE ENGINEERING [CS427]

Dr Dave Johnson  
Prof Ted Hurley  
Dr Niall Madden

Time allowed: Two hours.  
Attempt **FOUR** questions.

---

1. (a) Draw a diagram representing the *Waterfall model* of the Software Development Life Cycle. Why is this model also known as the *Linear Model*?
  - (b) Write a short note on the *Requirements phase*. What is the main difference between *Requirements* and the *Specification* phases?
  - (c) The *Spiral* model is an alternative approach to software engineering. Give a short description of it, with an appropriate diagram. What is the main advantage of the *Spiral* model over the *Waterfall* model?
2. (a) Use first-order predicate logic to express the statement  
*Every even number greater than three is sum of two primes.*  
Show that this is the same as  
*There is no even number  $x$  greater than four such that  $x_1 + x_2 = x$  implies that  $x_1$  or  $x_2$  is composite.*
  - (b) The requirement of a piece of software is  
*The software will take two finite sequences of integers as input and determine if they are a permutation of each other*  
Give a specification for this using first-order predicate logic.  
Show how it will work on the input  
(i)  $\{0, 0, 1, 3, 2\}$  and  $\{0, 2, 3, 0, 1\}$ .      (ii)  $\{0, 0, 1, 3, 2\}$  and  $\{0, 1, 1, 3, 2\}$ .

3. (a) Give a short description of the *Design Phase* of the software life-cycle.  
What is the *Traceability Matrix* and why must it be invertible?
- (b) You've have been asked to design the Graphical User Interface for an interactive spell-checker. The system has a GUI. The system reads each word in the document. If a word is found that is not in the dictionary, a list of up to 10 suggestions is presented. The user may click on:  
(0-9) one of the suggestions,  
(a) ACCEPT, (b) ACCEPT ALL, (c) ENTER WORD, (d) QUIT.  
Give a Finite State Machine (FSM) description of such a GUI, stating clearly what the states, events and actions are.
4. (a) Choose three computer languages and suggest what type of software projects (or subsystems) they might be suitable for? What advantages or disadvantages would your choice have?
- (b) What is *beta testing*? In what case might it be used, and what are its advantages? Would it be part of *System* or *Unit* testing? Might it be classified as *functional* or *structural* testing?
5. (a) What is a pointer? Given an example of when it is preferable to use pointers, along with dynamic memory allocation, rather than statically defined arrays.
- (b) You are writing a program that requires an array of strings. Explain why you actually need to use a two-dimensional array.  
Write a piece of C code that dynamically assigns space for an array of strings. It should work as follows:
  - the user is prompted for a number of strings and the maximum length of the strings.
  - Using `calloc()` or `malloc()` space is dynamically assigned for the data.
 Include declarations for any variables you might require.
6. (a) A CSV (Comma Separated Values) file is a plain text data file used to store entries from a database. As the name suggests, entries/cells are separated by commas. Lines are terminated in the usual way.  
You are required to write a C function that counts the number of cells in a file. The prototype is:
 

```
int count_cells(char *file_name);    % Name of file is stored
                                     % in array pointed to by file_name
```

 The function should open the file for reading, count the number of cells, close the file, and return the number of cells.  
Give C code that would achieve this.
- (b) It transpires that the CSV file in part (a) contains information on lecture courses. The information include the course code (e.g., CS427), the course name (e.g., Elements of Software Engineering), the lecturer's name, and the number of students enrolled.
  - Give the code for a `struct` that you might use you store this information.
  - Show how to use `typedef` to create a user-defined type for this structure.
  - Show how to dynamically assign memory for an instance on the structure.