

Ollscoil na hÉireann, Gaillimh
National University of Ireland, Galway

GX0087

Semester 1 Examinations, 2005

Exam Code(s)	1SD1
	1MF1
	1IT1
Module Code(s)	CT854
	CT519
	CT509
Module(s)	Application Programming
	Programming
Paper No.	1
Repeat Paper	Special Paper
External Examiner(s)	Prof. S. McClean
Internal Examiner(s)	Dr. M. Madden
	Dr. S. Redfern
	Dr. F. Smith

Instructions:

Time allowed: 3 hours
Answer any 2 questions from section A
and
Answer any 2 questions from section B

All questions carry equal marks.
Please use separate answer books for each section.

Duration	3 hrs
No. of Answer books	2
<u>Requirements:</u>	
Handout	
MCQ	
Statistical Tables	
Graph Paper	
Log Graph Paper	
Other Material	
No. of Pages	4
Department(s)	Information Technology

SECTION A

Q.1.

(i) 6 marks

Prototype and implement a **function** called `dicero11()`. This function should simulate the rolling of two dice. Each dice generates a random number between 1 and 6. The function should return the sum of the two dice rolled, i.e. a number between 2 and 12. **Hint:** The `rand()` function from `<stdlib.h>` returns a random integer between 0 and the constant `RAND_MAX`.

(ii) 14 marks

Write a program which makes use of your `dicero11()` function in order to simulate 100 roles of two dice. The program should keep track of how many times each of the values between 2 and 12 is produced by these dice rolls. It should produce output in the form shown below: the number of times each value has been produced is displayed as a line of star symbols.

```
2: **
3: ****
4: *****
5: *****
6: *****
7: *****
8: *****
9: *****
10: *****
11: *****
12: *****
```

Q.2.

(i) 5 marks

Write a program to read in numbers until the number -999 is encountered. The sum of all numbers read until this point should then be printed out.

(ii) 8 marks

Write a **function** called **is_leap** that receives a year number as an argument and returns a number indicating whether that year is a leap year or not. (NB: a leap year is divisible by 4 but not by 100, *or* is divisible by 400. So, 1900 is not a leap year but 2000 is). Write a program that uses the **is_leap** function to list all of the leap years between the year 1900 and the year 2100.

(iii) 7 marks

Write a **recursive function** that returns the factorial of a number.

Note: the factorial of a number n is $(n)*(n-1)*(n-2)*\dots*(1)$, i.e. 6 factorial is $6*5*4*3*2*1$

Q.3.

(i) 5 marks

Write a program that accepts two floats (real numbers) from the keyboard, and then repeatedly halves the larger of the two numbers, until it drops below the smaller of the two.

(ii) 15 marks

Write a program that accepts 9 integers from the user and then calculates and displays the **mean**, **median**, and **mode** of that set of numbers, where:

Mean = the average (i.e. sum of all values divided by number of values).

Median = the number in the set which has exactly half of the other numbers less than it and half greater than it.

Mode = the most common number in the set.

Sample output:

```
Enter 9 integers: 8 3 6 3 7 3 8 2 9
Mean: 5.44444444
Median: 6
Mode: 3
```

SECTION B

Q.4.

(i) *10 marks*

The `strncpy(s1,s2,n)` function copies exactly `n` characters from `s2` to `s1`, truncating `s2` or padding it with extra null characters as necessary. The target string may not be null-terminated if the length of `s2` is `n` or more. The function returns `s1`. Write your own version of this function.

(ii) *10 marks*

Write a function that replaces the contents of a string with the string reversed. Your function should work for a string of any length.

Q.5.

(i) *15 marks*

Write a program that takes two command-line arguments. The first is a string; the second is the name of a file. The program should then search the file, printing all lines that start with the given string.

(ii) *5 marks*

Describe some potential problems with the way that C handles strings. Detail how you would avoid these problems.

Q.6.

20 marks

Create a structure that can describe a restaurant. It should have members that include the name, address, average cost and type of food. Suppose an array of structures representing restaurants has been created. Write a function that prints out all the restaurants of a given food type in order of cost, least expensive first.