

# THE BRITISH COMPUTER SOCIETY

## THE BCS PROFESSIONAL EXAMINATIONS BCS Level 4 Certificate in IT

### SOFTWARE DEVELOPMENT

24<sup>th</sup> April 2007, 2.30 p.m.-4.30 p.m.  
Time: TWO hours

Both Section A and Section B carry 50% of the marks. You are advised to spend about 1 hour on Section A (30 minutes per question) and 1 hour on Section B (12 minutes per question).

*The marks given in brackets are **indicative** of the weight given to each part of the question.*

Calculators are <b>NOT</b> allowed in this examination.
---

#### SECTION A

Answer TWO questions out of FOUR. Each question carries 30 marks.

1. The annual flood of the river Nile is important to the prosperity of countries such as Sudan and Egypt. Readings are taken of the river level at fixed stations along the Nile bank every hour and collected in a serial file. Each record has the following fields:

Station name (30 characters)

Station Reference (8-digit integer)

Date (ddmmyy) where dd = day digits, mm = month digits, yy = year digits

Followed by 24 pairs of values, each having hour of reading (24-hour clock) and the reading of the river level (real number)

- a) Write a suitable description of this record in a language of your choice. State which language you are using. **(5 marks)**
- b) Write ONE typical record which could be used as test data. **(3 marks)**
- c) The following initial algorithm is for a program intended to detect when the flood is beginning:
- ```
FOR each station from first to last DO
  READ one record from the file
  FOR each hourly reading in the record DO
    Obtain difference between readings at current hour and the previous hour
    IF the water has risen THEN
      PRINT the increase in river height
    IF three successive differences are positive THEN
      Calculate the rate of river rise as an hourly percentage
  ENDFOR
ENDFOR
```

Develop this algorithm to the point where coding would be straightforward. Use the record structure you have written for part a) but do not copy it out again. State your intended target language.

**(22 marks)**

**Turn over]**

2. a) Dry run the following pseudocode with input values  $x = 18, y = 36$ .  
All the variables hold integer values.

**(15 marks)**

| Line number | Code                                            |
|-------------|-------------------------------------------------|
| 1           | FUNCTION ged(u,v)                               |
| 2           | REPEAT                                          |
| 3           | IF u < v THEN                                   |
| 4           | t ← u                                           |
| 5           | u ← v                                           |
| 6           | v ← t                                           |
| 7           | ENDIF                                           |
| 8           | u ← u - v                                       |
| 9           | UNTIL u = 0                                     |
| 10          | ged ← v                                         |
| 11          | ENDFUNCT                                        |
| 12          | BEGIN {top level}                               |
| 13          | PRINT 'EXECUTION STARTS HERE'                   |
| 14          | PRINT 'Input top and bottom values of fraction' |
| 15          | INPUT x , y                                     |
| 16          | z ← ged(x , y)                                  |
| 17          | PRINT 'Greatest common factor = ',z             |
| 18          | x ← x DIV z                                     |
| 19          | y ← y DIV z                                     |
| 20          | PRINT 'reduced fraction ',x,' / ',y             |
| 21          | ENDPROG                                         |

Note - DIV is integer division; thus  $5 \text{ DIV } 3 = 1$ .

- b) Write brief notes on the use of identifiers and comments in the code. **(4 marks)**
- c) Re-write the code so that values of u are printed on entry to and exit from the function. Incorporate any improvements which would make the code more readily understood by programmers. **(11 marks)**

3. a) What are the operations usually applied to the data structure called a stack? **(5 marks)**
- b) Write an implementation of a stack (data structure and operations) in a programming language that you know. The values to be stored in the stack will be integers. You may assume that there will never need to be more than 20 values stored in the stack at any one time. **(15 marks)**
- c) Using your implementation, create a program which reads 10 integers from the input and outputs them in reverse order. **(10 marks)**

4. a) A function can be recursive or not recursive. If you were reading a function definition, how would you discover if it was recursive? **(4 marks)**
- b) Any function is allowed to have local variables, but what special arrangement has to be made for local variables of a recursive function? **(4 marks)**
- c) Write down the output you would expect from the function *rec* below following the call *rec(5)*
- Note: The function rec is defined in two languages (version 1 & version 2). You may use either definition to obtain your answer. (12 marks)*
- d) Suppose the programmer missed out the 'if' test and that the middle line was just written *rec(p-1)*. What would happen now when the function was called? **(5 marks)**
- e) Based on your answer to d), give ONE important rule for a programmer writing a recursive function. **(5 marks)**

| Version 1                                                                                                                  | Version 2                                                                                                                  |
|----------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|
| <pre>void function rec(int p){   printf("starting %d \n",p);   if(p&gt;0) rec(p-1);   printf("finishing %d \n",p); }</pre> | <pre>FUNCTION rec( p as INTEGER) PRINT "starting " p IF p GREATER THAN 0 THEN rec(p - 1) PRINT "finishing" p ENDFUNC</pre> |

### SECTION B

Answer FIVE questions out of EIGHT. Each question carries 12 marks.

5. a) Write a program which prints the mean number of weeks (**mean**) in every month for a non-leap year. Thus January has 31 days hence its mean = 31/7 or 4.4285 weeks.
- b) Calculate the standard deviation (S) from the formula given below:
- $$S = \text{SQRT} \left[ \sum_{i=\text{Jan}}^{\text{Dec}} (\text{mean}_i - \text{meanwk})^2 / 12 \right]$$
- where **meanwk** is calculated by adding the **means** for January to December together and dividing the total by 12.
- Information:  
 January, March, May, July, August, October, December have 31 days  
 April, June, September, November have 30 days  
 February has 28 days. **(12 marks)**

6. A factory makes ball bearings having 1.0 cm diameter. As part of the quality control process a sample of these is made regularly and the diameter of them measured accurately. Those whose diameters differ from 1.000 cm by up to 0.005 cm are classed as grade 1; those that differ by up to 0.05 are classed as grade 2. Others are rejected and returned for re-processing.

Write an interactive program which accepts ball bearing diameters until a negative value is input. It counts and prints how many of them are grade 1, grade 2 or rejects with corresponding percentages. **(12 marks)**

7. a) What does the term Random Access mean? **(4 marks)**
- b) Briefly describe a part of a computer system which can be used as a Random Access device. **(4 marks)**
- c) Give brief details of situation in which a Random Access file would ideally be used. **(4 marks)**

**Turn over]**

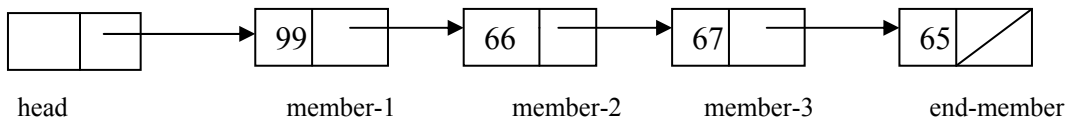
8. The function  $F_n(x) = \text{LN}(1.0 + \text{SIN}(x))$  may be evaluated using the Maclaurin series:

$$F_n(x) = x - 1/2 x^2 + 1/6 x^3 - 1/12 x^4 + 1/20 x^5 - \dots$$

A finite number of terms (N) are used in an evaluation of  $F_n(x)$ .

- Write down the next two terms in this series. **(2 marks)**
- Show how to calculate each term in the series. **(4 marks)**
- Write a program which evaluates  $F_n(x)$  from input values of  $x$  and  $N$  using the Maclaurin series and prints the calculated value. The program should compare this value with the value obtained using the system functions for  $\text{LN}(x)$  and  $\text{SIN}(x)$  in the function definition and print the difference between these two values  
Information:  $\text{LN}(x)$  is the logarithmic function using  $e$  as base. **(6 marks)**

9.



The diagram represents a linked list with a single pointer field.

- Write a declaration for one member of this list in a suitable language. State which language you are using. **(3 marks)**
  - Copy the diagram and show the necessary pointer movements to delete member-2 and insert it after member-3. **(3 marks)**
  - Write code for these pointer movements in the same language you used in part a). Include the necessary variable declarations in your answer. **(6 marks)**
10. Consider the design and implementation of the search facility in a text editor. During the process a specification will be drawn up, sometime later an algorithm will be produced and later still some documentation. With reference to this example write brief notes on what you understand by:
- a specification **(4 marks)**
  - an algorithm **(4 marks)**
  - documentation **(4 marks)**
11. a) Describe the difference between sequential and parallel programming. **(4 marks)**
- b) Describe a problem where parallel programming is useful. What characteristics does the problem possess which makes parallel programming particularly suitable? **(8 marks)**
12. It is possible to group software under three headings: Systems Software, Programming Software and Applications Software.
- In which of the three groups you would put each of the following items of software
    - Pascal Compiler
    - iTunes music software
    - Windows XP OS
    - Outlook express email client
    - UNIX OS
    - Javascript interpreter**(3 marks)**
  - Describe briefly the three software categories in your own words in a way that would help someone work out in which category to place a new piece of software. **(9 marks)**