The Open University of Sri Lanka
Department of Electrical and Computer Engineering

032

ECD1213 – Data Structures & Algorithms
Final Examination – 2005

(Closed Book Test)

Date: 22nd March 2006	**Time: 13.30 – 16.30 hrs**

PART A

Note: Answer all questions.

Program segments can be written in Pascal/ C/ Visual Basic or Java. Students must clearly specify the programming language used when answering questions.

1.

a) Describe advantages of postfix notation of arithmetic expressions over infix notation.
b) Explain the operation of *'Reverse Polish Calculator'*.
c) Transfer the following infix expression into postfix notation and show how the postfix expression is implemented within the 'stack data structure'.

75 + 15 / 3 – 3 * (18 + 2)

d) Declare a stack (15 elements) long, and implement the above transformed postfix expression in the declared stack. *(20 marks)*

2.

a) Describe what is meant by *'Recursion'*
b) Explain the *'Fibonacci Number Sequence'*.
c) Draw a flow chart/ structure chart/ pseudo code to get first 10 *'Fibonacci number sequence'* using the recursion principle.
d) Write a program segment to first 10 *'Fibonacci number sequence'*.

(20 marks)

PART B

Note: Answer any three questions. All questions carry equal marks.

1.

a) Describe the difference between the *'queue'* and the *'priority queue'* data structures.
b) In the word 'ORGANIZE', letters 'O', 'R', 'G', 'A', 'N', 'I', 'Z', 'E', is entered to a *'queue'* according to the order that letters are entered. Explain the principle

and show how the normal queue is implemented using these letters. *(you must show your declaration)*

c) If the above letters are entered in a *'priority queue'* according to the order of English Alphabet, show how the priority queue is implemented. *(You must show your declaration)*

d) What is the key value used to arrange letters in the priority queue in part (c).

(20 marks)

2.

a) Describe the *'Selection sort'* algorithm.

b) Illustrate the *'Selection sort'* using the following data set.
12, 34, 5, 45, 67, 8, 22, 96, 18

c) Implement the *'Selection sort'* algorithm using a suitable programming language.

(20 marks)

3.

a) Give the declaration for the *'binary tree'*.

b) Explain the *'pre-order'*, *'in-order'*, *'post-order'* traversing methods.

c) Draw the binary search tree results from inserting the following numbers in the order.
14, 15, 4, 9, 7, 18, 3, 5, 16, 4, 20, 17, 9, 14, 5

d) Write the results after the above tree is traversed using the traversing methods in part (b).

e) Construct an *'AVL balance tree'* inserting the following nodes in order. Draw all the necessary steps.
15, 22, 36, 26, 28, 9

(20 marks)

4.

a) Explain what is meant by a *'singly linked list'* using a diagram.

b) Give the declaration for the *'singly linked list'*.

c) Write a procedure to add a new node to the *'singly linked list'*.

(20 marks)

5. A complex number is a one that contains real and imaginary parts, both of which are real parts. If *c1* has real and imaginary parts *r1* and *i1*, respectively, and *c2* has real and imaginary parts *r2* and *i2*, respectively, then

- The sum of *c1* and *c2* has real part *(r1+r2)* and imaginary part *(i1+i2)*
- The difference of *c1* and *c2* has real part *(r1-r2)* and imaginary part *(i1-i2)*
- The product of *c1* and *c2* has real part *[(r1*r2)-(i1*i2)]* and imaginary part *[(r1*i2)+(r2*i1)]*

a) Implement complex numbers by declaring a record (in pascal) with real and imaginary parts. (Note: It is 'Structure' in C programming language).

b) Write procedures to add subtract and multiply such complex numbers.

(20 marks