

NATIONAL UNIVERSITY OF SCIENCE AND TECHNOLOGY
FACULTY OF APPLIED SCIENCE
COMPUTER SCIENCE DEPARTMENT
JULY SUPPLEMENTARY EXAMINATIONS 2005

SUBJECT: DATA STRUCTURES AND ALGORITHMS
CODE: SCS2103

INSTRUCTION TO CANDIDATES

Answer any five questions. Paper contains Seven questions.

Time: 3 hours

QUESTION ONE

Explain what the following functions do, use function calls to explain yourself

- a) void mac(char* s, int m, int n)
{char c;
if (m<n)
{c= s[m]; s[m] = s[n]; s[n]=c;
mac(s,m+1,n-1);}
}
- [10]
- b) void r(int n)
{printf("%1d", (n%10));
if ((n/10)!=0) r(n/10);}
- [10]

QUESTION TWO

- a) why is it important to free dynamically allocated memory? [2]
b) Which function serves a similar purpose as the free() function but on a different data structure? [2]
c) What determines the size of a pointer variable? [2]
d) What is a header file? [2]
e) What is a string? [2]
f) What performance attributes of a program are used to determine its efficiency? [2]
g) Why is it necessary to declare a variable before it is used? [2]
h) What is the advantage of double linked lists over single linked lists? [2]
i) What is a stack overflow? [2]

QUESTION THREE

Assuming the following definition:

```
typedef struct item
{ int num;
  struct item * l;
  struct item * r;
}node;
```

Write a function that will accept a binary search tree and a node and **recursively** insert the node in the appropriate location.

[20]

QUESTION FOUR

- a) i. What is the base case in a recursive program? [2]
ii. What is the natural way to consider decomposing a linked list into substructures helpful for devising a recursive solution to a problem. [2]
iii. What is a list? [2]
iv. When would it be advantageous to use a sequential array list representation instead of a linked list representation? [2]
v. What is a preprocessor directive? [2]
- b) Write a recursive function `mult(m,n)`, to multiply two positive integers, `m` and `n`, using only repeated addition [10]

QUESTION FIVE

- a) Explain what the following function does without mention of any variable names or constructs in the code. [10]
b) Diagrammatically trace through the function call `b(5)` [10]

```
node* b(int n)
{   if (n<1) return NULL;
    else
    {   node* ptr = (node*)malloc(sizeof(node));
        node* temp = ptr;
        for (int i = 1; i<n; i++)
            {   for(int j = 1; j<i/2; j++)
                    temp=temp->next;
                node* nu =
                (node*)malloc(sizeof(node));
                nu->next = temp->next;
```

```
temp->next = nu;  
temp = ptr;  
return ptr;  
}
```

QUESTION SIX

Write a function to illustrate the bubble sort algorithm and explain it briefly [20]

END OF QUESTION PAPER