University of Puerto Rico at Rio Piedras
Department of Mathematics
Qualifying Exam
MATE 6681: Higher Level Programming Languages

September-21, 2011

Your Name	Your Student ID

Instructions
Solve **exactly three** of the following five exercises. Please write clearly and show all your work to get credit. Good luck!

Exercise 1 (100 points)
The following is a C function that computes the number of (decimal) digits in an integer:

```
int numdigits(int x)
{
        int t = x;
        int n = 1;
        while (t >= 10 )
        {
                n++;
                t = t / 10;
        }
        return n;
}
```

(a) (70 points) Rewrite this function in functional style (i.e. as a recursive function).

(b) (30 points) Show how your function works on x = 7640

Exercise 2 (100 points)
The following C function computes the factorial of an integer:

```
int fact (int n)
{
        if( n <= 1)
                return 1;
        else
                return n * fact (n - 1);
}
```

Factorials grow extremely rapidly, and overflow is soon reached in function fact(n).

(a) (20 points) What happens during the execution of fact(n) when overflow occurs?.

(b) (20 points) Propose a solution to correct the overflow problem in part (a).

(c) (10 points) Show a C Implementation of your solution from part (b) into the given recursive function fact(n).

(d) (50 points) Rewrite the original fact(n) function into imperative style (i.e., using variables and eliminating recursion).

Exercise 3 (100 points)
Consider the following grammar:

$expr \rightarrow expr + term \mid term$
$term \rightarrow term * factor \mid factor$
$factor \rightarrow (\ expr\ ) \mid number$
$number \rightarrow number\ \ digit \mid digit$
$digit \rightarrow 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

Draw (a) (60 points) the parse tree and (b) (40 points) the *abstract syntax tree* for the following arithmetic expression:

$$3 * (4 + 5) * (6 + 7)$$

Exercise 4 (100 points)
Write a BNF description of (a) (50 points) postfix arithmetic expressions and (b) (50 points) prefix arithmetic expressions.

Exercise 5 (100 points)
Given the following C program, (a) (40 points) draw box-and-circle diagrams of the variables after each of the two assignments to **x (lines 11 and 15). (b) (30 points) Which variables are aliases of each other at each of those points? (c) (30 points) What does the program print?

```
(1) #include <stdio.h>
(2) main()
(3) {    int **x;
(4)      int *y;
(5)      int z;
(6)      x = &y;
(7)      y = &z;
(8)      z = 1;
(9)      *y = 2;
(10)     *x = y;
(11)     **x = z;
(12)     printf("%d\n",*y);
(13)     z = 3;
(14)     printf("%d\n",*y);
(15)     **x = 4;
(16)     printf("%d\n",z);
(17)     return 0;
(18) }
```