



**Universidad de Puerto Rico
Recinto Universitario de Mayagüez
Departamento de Ingeniería Eléctrica y Computadoras**

**ICOM 4036 – Programming Languages
Primavera 2003**

**Ejercicios de práctica
Examen Parcial II**

- 1) Write finite state diagrams to recognize the following languages:
 - a) Sequence of 0's and 1's containing the pattern "010111". Careful with overlapped patterns
 - b) Identifiers that begin with letter or underscore followed by letters, digits or underscore, and ending with a dollar (\$) sign.
- 2) Write a BNF for Fortran variable declarations
- 3) Write a BNF for Fortran statements. You may assume that a non-terminal <expression> has already been defined.
- 4) Write a BNF for Fortran functions using the BNF's from (2) and (3)
- 5) Write recursive descent parsing functions in the style of Sebesta (see lecture notes) for each of the BNF non-terminals defined in (2), (3) and (4).
- 6) Write an iterative procedure in FORTRAN to compute and return the greatest common divisor (GCD) of 2 integer arguments.
 - a) $GCD(a, b) = b$ if b divides a
 - b) $GCD(a, b) = GCD(b, r)$ otherwise, where $r = a \text{ MOD } b$
- 7) Write a FORTRAN iterative procedure called *precision()* with no arguments. The procedure must return the smallest double precision floating point number e that can be added to 1.0 such that the result of the sum $e+1.0$ is different from 1.0.
- 8) Write a procedure *roots(a,b,c)* that receives the three float coefficients of a polynomial and returns an integer representing the number of distinct real roots.
- 9) Write FORTRAN functions $\sin(x,p)$, $\tan(x,p)$ and $\exp(x,p)$ that take two REAL arguments. The procedure should return the approximated floating point value of the

function by computing the sum of an appropriate Taylor series. The result should be approximated to a precision p , that is, the value returned by the function should not differ from the exact value by more than p .

- 10) Write an imperative Fortran subroutine that receives an array of integers and its length and sorts the array using a well known sorting algorithm.