

Ejercitación N.º 1: Implementación de algoritmos sencillos en Fortran y errores inherentes al cálculo computacional

Breve repaso de FORTRAN

Ejercicio 1

Para cada ítem, escriba un código Fortran 90 que:

- Escriba en pantalla un mensaje cualquiera, por ejemplo: "Hola".
- Escriba en pantalla un pedido de ingreso por teclado de un número entero, lea el número entero y luego escriba en pantalla el texto: "El número ingresado fue:", seguido del número.
- Escriba en pantalla pedidos de ingreso por teclado de dos números enteros, los lea y luego escriba en pantalla el texto: "Los números ingresados fueron:", seguido del número, y "la suma de ambos es:", seguido del resultado de la suma.
- Escriba en pantalla un pedido de ingreso por teclado de un número real no negativo, lea el número y luego escriba en pantalla el texto: "La raíz cuadrada del número ingresado es:", seguido del valor correspondiente.
- Escriba en pantalla un pedido de ingreso por teclado de un número natural, N , lo lea y si $N < 10$, que escriba en pantalla: " $N < 10$ " y en caso contrario, que escriba en pantalla: " $N \geq 10$ ".
- Escriba en pantalla un pedido de ingreso por teclado de un número natural, N , lo lea y luego escriba N veces el mensaje "Hola".
- Escriba en pantalla pedidos de ingreso por teclado de dos números naturales distintos, N_1 y N_2 , y los lea. Luego que escriba en pantalla un pedido de ingreso por teclado de un número real positivo x , y lo lea. Finalmente, si $x < 10$, que escriba en pantalla N_1 veces: " $x < 10$ " y en caso contrario, escriba en pantalla N_2 veces el mensaje: "La raíz cuadrada de x es:", seguido del valor correspondiente.
- Escriba un programa que pueda ser utilizado como una calculadora que haga operaciones elementales (+, -, *, %) y evalúe las funciones elementales que se usan más frecuentemente.

Ejercicio 2

Escriba un programa utilizando la sentencia `IMPLICIT NONE`, y no declare una variable que utilice en el programa. ¿Qué tipo de error le da el compilador? NOTA: De ahora en adelante utilice **SIEMPRE** la sentencia `IMPLICIT NONE` para todos sus programas, funciones y subrutinas.

Procesos básicos. Aritmética de enteros y reales. Funciones y Subrutinas. Procedimientos intrínsecos.

Ejercicio 3

- (a) Escriba un programa que genere una lista de números equiespaciados con un paso $h = 0.5$ en el intervalo $[1.0, 20.0]$. Imprimirlos por pantalla.
- (b) Escriba un código que genere una grilla uniforme de $n = 15$ valores `real(8)` en el intervalo $[\pi, 2\pi]$, y los almacene en un arreglo de rango 1, de dimensión n . Declare la variable n como un parámetro y utilícelo para definir la dimensión del arreglo donde almacenará los números. También defina π como un parámetro y asigne al mismo el valor $\pi = 4.0 * \text{atan}(1.0)$. Imprima por pantalla los valores del arreglo.

Ejercicio 4

Escriba un procedimiento tipo función que calcule el valor de la función $f(x) = e^{-x} \sin 2x$ y aplíquela a los datos generados en el ejercicio anterior.

Ejercicio 5

Escriba un programa que imprima en pantalla el resultado de las operaciones $i/5$ e $i/5.0$ para $i = 0, 1, \dots, 20$ (i declarado como variable entera). Analizar las diferencias que observa en el resultado.

Ejercicio 6

Escriba una función FORTRAN que:

- (a) determine si un número entero es par o impar.
- (b) calcule la función $f(n)$ tal que $f(n) = 4.0$ si n es impar; y $f(n) = 2.0$ si n es par.

Ejercicio 7

Escriba una subrutina que, sin hacer uso de funciones intrínsecas, determine si dos números enteros son divisibles entre sí. Escriba además un programa que reciba por teclado dos números, llame a la subrutina para determinar si los mismos son divisibles y escriba si lo son o no, en pantalla. Ahora investigue el funcionamiento de la función intrínseca `mod` y utilícela en la subrutina que escribió anteriormente.

Ejercicio 8

Escriba un procedimiento de tipo función que determine el mínimo de una lista de n valores reales almacenados previamente en un arreglo. Escriba el resultado en pantalla.

Utilice los siguientes datos: $A = [23.33, 8.08, 44.99, 21.95, 44.93, 29.82, 41.08, 28.57, 4.39, 43.79, 20.02, 26.59, 41.13, 39.32, 26.85]$

NOTA: El resultado debe ser escrito en pantalla por el programa principal, no por la función. Se considera una mala práctica de programación que las funciones y subrutinas escriban resultados por pantalla. Sí se puede hacer que escriban en pantalla en el caso de que ocurra un error.

Investigue el funcionamiento de la función intrínseca `minval` y utilícela en su programa para comparar con el resultado de la función que escribió anteriormente.

Ejercicio 9

Escriba una subrutina que intercambie el valor de dos variables reales a y b , y un

programa principal que emplee esta subrutina y al cual los valores de a y b sean i) ingresados por teclado, y ii) leídos desde un archivo.
En ambos casos el resultado debe ser escrito en pantalla y en un archivo de salida.

Ejercicio 10

Idem ejercicio anterior para dos arreglos de rango 1, A y B . En este caso se debe chequear ambos arreglos tengan la misma cantidad de elementos, para lo cual investigue el funcionamiento de la función intrínseca `size`.

Ejercicio 11

Escriba una subrutina FORTRAN que ordene de menor a mayor los elementos del arreglo del ejercicio 8.

EXTRA: luego que funcione bien la subrutina, agregue un argumento que le permita elegir ordenar el arreglo de menor a mayor o de mayor a menor.

Tipos de variables enteras y reales. Módulos. Alojamiento de memoria.

Ejercicio 12

Escriba un programa para determinar las diferentes valores que pueden tomar las funciones intrínsecas `SELECTED_INT_KIND(i)` y `SELECTED_REAL_KIND(i, j)`, y el rango de valores enteros y reales que pueden tomar las variables de cada tipo y clase. Recuerde que un valor de salida de las funciones `SELECTED_INT_KIND(i)` y `SELECTED_REAL_KIND(i, j)` igual a -1, significa que los argumentos (i o el par i, j) no son valores soportados.

Ejercicio 13

Escriba un módulo (llamado "kinds") en el que se definan los números correspondientes a cada clase (KIND) de variables enteras y reales obtenidos en el ejercicio 12.

Sugerencia 1: a modo de comentario (usando el símbolo "!" a la derecha de la definición de la variable que indica la clase), indicar el rango de valores que puede representarse con esa clase de variable.

Sugerencia 2: conserve este módulo para ser utilizado a lo largo de toda la materia.

Ejercicio 14

Escriba un programa FORTRAN que lea desde un archivo pre-existente la dimensión, n , de dos arreglos de rango 1 (vectores de componentes reales) A y B , seguido por las componentes de cada uno de los vectores y luego obtenga el vector $C = A + B$. Realizar esta tarea utilizando las sentencias que permiten una reserva dinámica de memoria para los 3 vectores (`ALLOCATABLE`, `ALLOCATE`, `DEALLOCATE`). Para calcular C , programe la suma componente por componente utilizando un loop y también utilizando la correspondiente "whole array operation" de FORTRAN.

Errores numéricos.

Ejercicio 15

Calcule el valor de las raíces de una ecuación cuadrática $ax^2 + bx + c = 0$, con $a =$

1.0 , $b = 3000.001$ y $c = 3.0$. Compare el valor calculado con las raíces verdaderas $x_1 = -0.001$ y $x_2 = -3000$.

Ejercicio 16

Escriba una función FORTRAN que evalúe la sumatoria:

$$S_{M,N} = \sum_{i=M}^N i$$

donde M y N son números enteros. Escriba además un programa principal que utilice dicha función para calcular $S_{1,10}$, $S_{5,15}$, $S_{-5,10}$ y $S_{-20,50}$.

Ejercicio 17

La suma

$$f(n) = \sum_{i=1}^n \frac{1}{i^4}$$

converge a un valor de $\frac{\pi^4}{90}$ conforme n tiende a infinito.

Escriba un programa de precisión sencilla, es decir que f debe ser de tipo `real(4)`, para calcular $f(n)$ para $n = 10000$ calculando la suma desde $i=1$ hasta 10000. Después repita el cálculo pero en sentido inverso, es decir, desde $i=10000$ a 1, con incrementos de -1 . En cada caso, calcule el error relativo porcentual verdadero (respecto al valor exacto de la serie). Explique los resultados.

AYUDA 1: declare la variable i de tipo `integer(kind=8)`, de otro modo el resultado de i^4 excede el valor máximo que la variable puede almacenar.

AYUDA 2: tenga en cuenta que el resultado de la división de dos enteros en FORTRAN da como resultado un entero. Tome los recaudos necesarios para que $\frac{1}{i^4}$ de un número real.

Ejercicio 18

Escriba un programa FORTRAN que evalúe el factorial de un número natural. Determine el rango de valores para el cual su rutina da el valor correcto del factorial dependiendo de la precisión utilizada para las variables del programa. ¿Qué estrategia adoptaría para generar una rutina correcta para el mayor argumento natural posible?

Ejercicio 19

Emplee la expansión de la serie de Taylor de cero hasta tercer orden para predecir $f(2)$ si $f(x) = 25x^3 + 6x^2 + 7x + 8$, usando como punto base $x = 1$. Calcule el error relativo porcentual verdadero ϵ_t para cada aproximación.

Problemas desafío

Ejercicio 20

La función $\log(1+x)$ se puede expresar como una serie de potencias de la siguiente manera:

$$\log(1+x) = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n} x^n \quad \text{si } |x| < 1$$

- escriba una función FORTRAN que evalúe los primeros N términos de la serie, utilizando variables doble precisión y verificando que x se encuentre dentro del intervalo de convergencia de la serie y en caso contrario, que escriba un mensaje de error en pantalla. Los argumentos de la función deben ser x y N .
- compare el resultado de la función que escribió con el de la función intrínseca `log`, y determine el valor de N tal que la diferencia entre ambas sea menor a 10^{-9} . Para dicho test utilice, $x = 0, 0.2, 0.5$.

Ejercicio 21

Escriba un programa FORTRAN que evalúe el factorial de un número natural. Determine el rango de valores para el cual su rutina da el valor correcto del factorial dependiendo de la precisión utilizada para las variables del programa. ¿Qué estrategia adoptaría para generar una rutina correcta para el mayor argumento natural posible?

Ejercicio 22

Escriba un código FORTRAN que lea el orden n , la matriz de coeficientes y los términos independientes de un sistema de n ecuaciones lineales algebraicas y verifique si es posible transformar la matriz de coeficientes en una que sea diagonalmente dominante (intercambiando el orden de sus filas). En caso que eso sea posible, el código también debe realizar tal transformación.

Una matriz A de $n \times n$ es diagonalmente dominante si se verifica:

$$|a_{ii}| > \sum_{j=1, j \neq i}^n a_{ij} \quad \forall i = 1, \dots, n$$

SUGERENCIA: Para intercambiar dos filas de la matriz A puede utilizar la subrutina que desarrolló en el ejercicio 10. Suponiendo que dicha subrutina se llame `swap_arr`, para intercambiar las filas j y k debe llamarla de la siguiente forma:

```
call swap_arr(n, A(j,:), A(k,:))
```