
Ingeniería en Seguridad Informática y Redes

Ingeniería en Sistemas Computacionales

Algoritmos de Solución Numérica

# Problemas con Comprobación

SI727576 - Edgar Guzmán Claustro

IS727272 - Marco Ricardo Cordero Hernández

Jal., 29 de agosto de 2023

## Contexto

En el marco del desarrollo de algoritmos, la gran parte del tiempo se espera que algún error salga de su ejecución (esto no necesariamente implica que este comportamiento sea deseado). A raíz de esto, la consecuencia esperada es la resolución de errores para lograr los resultados deseados. Una gran herramienta preliminar, la cual en realidad debería ser el paso 0 para cualquier desarrollo de la resolución de un problema, son las *corridas de escritorio* para verificar el comportamiento de los algoritmos.

Con el trasfondo provisto, el documento actual tiene el propósito de demostrar la utilización de corridas de escritorio en algoritmos previamente diseñados, los cuales atienden a la resolución de los siguientes problemas:

1. Entregar como resultado el número de días que faltan para el final del año desde una fecha dada por el usuario. Restricción: No considerar años bisiestos.
2. Resolver la multiplicación de un par de matrices cuadradas de una dimensión dada por el usuario.

## Resultados

Problema 1

*Pseudocódigo*

```
        Algoritmo dias_restantes
1.          Escribir Sin Saltar "Ingrese día "; Leer día

2.          Si 0 < día y día <= 31 Entonces
3.                  Escribir Sin Saltar "Ingrese mes "; Leer mes

4.                  Si 0 < mes y mes <= 12 Entonces
5.                          Segun mes Hacer
6.                                  2:
7.                                          resultado <- 28 - día
8.                                  4, 6, 9, 11:
9.                                          resultado <- 30 - día
10.                                 De Otro Modo:
11.                                         resultado <- 31 - día
                            Fin Segun

12.                         Si resultado >= 0 Entonces
13.                                 Mientras mes <> 12 Hacer
14.                                         mes <- mes + 1
15.                                         Segun mes Hacer
16.                                                 2:
17.                                                         resultado <- resultado + 28
18.                                                 4, 6, 9, 11:
19.                                                         resultado <- resultado + 30
20.                                                 De Otro Modo:
21.                                                         resultado <- resultado + 31
                                            Fin Segun
                                    FinMientras

22.                                 Escribir "Días restantes: ", resultado
23.                         SiNo
24.                                 Escribir "Combinación de día (", día, ") y mes (", mes, ")
          inválida. Revisa tus entradas."
                            FinSi
25.                 SiNo
26.                         Escribir "Mes inválido. Revisa tu entrada."
                    FinSi
27.         SiNo
28.                 Escribir "Día inválido. Revisa tu entrada."
            FinSi
        FinAlgoritmo
```

*Comprobación*
Caso de prueba: día = 14, mes = 2; Resultado esperado: 320 días


1. Escribir Sin Saltar "Ingrese día "; Leer día (día = 14)
2. Si 0 < día y día <= 31 Entonces (0 < 14 SÍ, y 14 <= 31 SÍ)
3. Escribir Sin Saltar "Ingrese mes "; Leer mes (mes = 2)
4. Si 0 < mes y mes <= 12 Entonces (0 < 2 SÍ, y 2 <= 12 SÍ)
5. Segun mes Hacer [variante de condicional para casos múltiples de variable]
6. 2 (mes(2) == 2 SÍ)
7. resultado <- 28 - día (resultado = 28 - 14 = 14)

…

12. Si resultado >= 0 Entonces (14 >= 0 SÍ)
13. Mientras mes <> 12 Hacer (2 != 12 SÍ) [Inician iteraciones]

| Iterador (mes != 12) | Siguiente mes (mes += 1) | Resultado |
|---|---|---|
| 2 | 3 | Resultado = 14 + 31 = 45 |
| 3 | 4 | Resultado = 45 + 30 = 75 |
| 4 | 5 | Resultado = 75 + 31 = 106 |
| 5 | 6 | Resultado = 106 + 30 = 136 |
| 6 | 7 | Resultado = 136 + 31 = 167 |
| 7 | 8 | Resultado = 167 + 31 = 198 |
| 8 | 9 | Resultado = 198 + 30 = 228 |
| 9 | 10 | Resultado = 228 + 31 = 259 |
| 10 | 11 | Resultado = 259 + 30 = 289 |
| 11 | 12 | Resultado = 289 + 31 = 320 |
| 12 | | |

22. Escribir "Días restantes: ", resultado (resultado = 320)

Caso de prueba: día = 32, mes = 11; Resultado esperado: Alerta de día inválido

1. Escribir Sin Saltar "Ingrese día "; Leer día (día = 32)
2. Si 0 < día y día <= 31 Entonces (0 < 32 SÍ, y 32 <= 31 NO)

…

27. SiNo
28. Escribir "Día inválido. Revisa tu entrada."

Caso de prueba: día = 13, mes = 13; Resultado esperado: Alerta de mes inválido

1. Escribir Sin Saltar "Ingrese día "; Leer día (día = 13)
2. Si 0 < día y día <= 31 Entonces (0 < 13 SÍ, y 13 <= 31 SÍ)
3. Escribir Sin Saltar "Ingrese mes "; Leer mes (mes = 13)
4. Si 0 < mes y mes <= 12 Entonces (0 < 13 SÍ, y 13 <= 12 NO)

…

25. SiNo
26. Escribir "Mes inválido. Revisa tu entrada."

Caso de prueba: día = 29, mes = 2; Resultado esperado: Alerta de combinación de parámetros inválida

1. Escribir Sin Saltar "Ingrese día "; Leer día (día = 29)
2. Si 0 < día y día <= 31 Entonces (0 < 29 SÍ, y 29 <= 31 SÍ)
3. Escribir Sin Saltar "Ingrese mes "; Leer mes (mes = 2)
4. Si 0 < mes y mes <= 12 Entonces (0 < 2 SÍ, y 2 <= 12 SÍ)
5. Segun mes Hacer [variante de condicional para casos múltiples de variable]
6. 2 (mes(2) == 2 SÍ)
7. resultado <- 28 - día (resultado = 28 - 29 = -1)

…

12. Si resultado >= 0 Entonces (-1 >= 0 NO)

…

23. Escribir "Combinación de día (29) y mes (2) inválida. Revisa tus entradas."

Problema 2

*Pseudocódigo*

***Se modificó el pseudocódigo ya que se detectaron errores en el antes entregado***

```
Algoritmo mul_mat
    definir size, counter, a,b,c Como Entero
    Mientras size <= 0 Hacer
        Escribir "Ingrese el tamaño de las matrices: "
        Leer size
    Fin Mientras
    Dimensionar a[size,size], b[size,size], c[size,size]
    Para i<-1 Hasta size Con Paso 1 Hacer
        Para j<-1 Hasta size Con Paso 1 Hacer
                Escribir 'Ingrese los datos para la matriz 1 [', i, ' : ', j, ']: '
                leer a[i,j]
                Escribir 'Ingrese los datos para la matriz 2 [', i, ' : ', j, ']: '
                leer b[i,j]
        Fin Para
    Fin Para
    Escribir "Resultado"
    counter<- 0
    Para i<-1 Hasta size Con Paso 1 Hacer
        Para j<-1 Hasta size Con Paso 1 Hacer
                Para k<-1 Hasta size Con Paso 1 Hacer
                        counter<- counter+1
                        c[i,j]<-c[i,j]+a[i,k]*b[k,j]
                        Si counter == size Entonces
                                Escribir "Resultado de la matriz [",i,"][",j,"]: ",c[i,j]
                                counter<-0
                        Fin Si
                FinPara
        FinPara
    FinPara
FinAlgoritmo
```

*Comprobación*

Caso de prueba, matriz 2x2

Matriz a

| | |
|---|---|
| 1 | 2 |
| 1 | 2 |

Matiz B

| | |
|---|---|
| 1 | 2 |
| 1 | 2 |

Matriz c (resultado)

| | |
|---|---|
| 3 | 6 |
| 3 | 6 |

Mientras size <= 0

1. Ingrese el tamaño de las matrices: Leer size (size = 2). Fin mientras
2. Ingresar datos de las matrices: i=j=1 hasta i=j=size

| i | j | a[i,j] | b[i,j] |
|---|---|---|---|
| 1 | 1 | 1 | 1 |
| 1 | 2 | 2 | 2 |
| 2 | 1 | 1 | 1 |
| 2 | 2 | 2 | 2 |

3. Multiplicación de matriz i=j=k=1 hasta i=j=k=size counter=0

| i | j | k | counter ++ | a[i,k] | b[k,j] | c[i,j]=c[i,j]+a[i,k]*b[k,j] | si counter == size |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | no |
| 1 | 1 | 2 | 2 | 2 | 1 | 3 | Sí, escribir c[i,j], counter = 0 |
| 1 | 2 | 1 | 1 | 1 | 2 | 2 | no |
| 1 | 2 | 2 | 2 | 2 | 2 | 6 | Sí, escribir c[i,j], counter = 0 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2 | 1 | 1 | 1 | 1 | 1 | 1 | no |
| 2 | 1 | 2 | 2 | 2 | 2 | 3 | Sí, escribir c[i,j], counter = 0 |
| 2 | 2 | 1 | 1 | 1 | 2 | 2 | no |
| 2 | 2 | 2 | 2 | 2 | 2 | 6 | Sí, escribir c[i,j], counter = 0 |

Caso de prueba, dimensión de matriz <= 0

Mientras size <= 0
Ingrese el tamaño de las matrices: Leer size (size = -2).
Vuelve a repetir instrucción de manera infinita hasta que se cumpla la condición: Ingrese el tamaño de las matrices.

## Conclusiones

Guzmán Claustro, Edgar

Al momento de realizar el ejercicio de la matrices, me percaté que se puede hacer aún más eficiente de lo que originalmente estaba por lo que tuve que actualizarlo. Decir que la programación no es como tal mi área de enfoque dentro de la carrera que actualmente curso, no es justificante para entregar un algoritmo ineficiente, añadiendo una gran falta de competencia al dejarlo de anterior manera.

Realizar "corridas de escritorio" para verificar el funcionamiento de un algoritmo, es una de las formas más simples y primitivas, en las que el desarrollador verifica su funcionalidad. Sin embargo, aprender a realizar estas ejecuciones manuales es fundamental para crear las bases en cualquier persona que programa un algoritmo. Y como cualquier cosa que se inicia a aprender, se tiene que tomar por lo más básico.

Cordero Hernández, Marco R.

En el ámbito de la ingeniería, muchos tendemos a ser egoístas y orgullosos con lo que hacemos, a tal grado en que no nos detenemos ni un momento a pensar en la posibilidad de que quizás lo que estemos haciendo no sea del todo correcto. Recalcando el factor del orgullo, el suponer que nuestros algoritmos que con tanto esfuerzo diseñamos, funcionará a la primera y con todos los casos que se le ingresen como entrada, es uno de los peores errores que se pueden cometer, y, en un ámbito productivo, esto puede llegar a ser catastrófico y muy costoso, no por nada existen individuos cuyas carreras se han desarrollado en torno al aseguramiento de la calidad.

Con este trasfondo, la presente actividad es solo una práctica que puede llegar a emplearse para corroborar el funcionamiento de algún algoritmo. Como bien se dice, 10 horas de depuración pueden ahorrar 15 minutos de comprensión y lectura del código, lo cual, para este caso, es idóneo. Quizás este ejercicio resulte tedioso en un contexto de desarrollo de alta cadencia, pero siempre es bueno realizarlo ocasionalmente.