

	Práctica
Facultad de Ingeniería	Laboratorio de docencia

Laboratorios de computación salas A y B

<i>Profesor:</i>	Alejandro Pimentel
<i>Asignatura:</i>	Fundamentos de Programación
<i>Grupo:</i>	Grupo 3 Bloque 135
<i>No de Práctica(s):</i>	Practica # 3
<i>Integrante(s):</i>	Muñoz Reyes Laura Vanessa - 2823
<i>No. de Equipo de cómputo empleado:</i>	Rumania 38
<i>No. de Lista o Brigada:</i>	No. de Cuenta: 3177522823
<i>Semestre:</i>	2020-1
<i>Fecha de entrega:</i>	02 de Septiembre del 2019
<i>Observaciones:</i>	Excelente

CALIFICACIÓN: 10

Practica #3: Solución de problemas y algoritmos

Introducción:

Todos los procesos realizados en la vida del ser humano llevan una serie de pasos ordenados para lograr ya sea un objetivo, una meta o un resultado, sin embargo muy pocas personas son conscientes que dicha lista corresponde a un algoritmo, el cual puede darle solución al problema de manera más concreta y eficaz.

Objetivo:

Elaborar algoritmos correctos y eficientes en la solución de problemas siguiendo las etapas de Análisis y Diseño pertenecientes al ciclo de vida del Software.

Marco Referencial:

¿Que son los algoritmos?

Un algoritmo es un conjunto ordenado e inequívoco de pasos ejecutables que definen un proceso finito.

Características Principales

- *Tiempo Secuencial*: Un algoritmo funciona en tiempo discretizado – paso a paso- definiendo así una “secuencia de estados computacionales” por cada entrada valida.
- *Estado abstracto*: Cada estado computacional puede ser descrito formalmente utilizando una estructura de primer orden y cada algoritmo es independiente de su implementación
- *Exploración acotada*. La transición de un estado al siguiente queda completamente determinada por una descripción fija y finita

Medios de expresión de un algoritmo

Los algoritmos pueden ser expresados de muchas maneras, incluyendo al lenguaje natural, pseudocódigo, diagramas de flujo y lenguajes de programación entre otros. Las descripciones en lenguaje natural tienden a ser ambiguas y extensas.

1. Diagrama de flujo
2. Diagrama Rectangulares Estructurados
3. Pseudocódigo
4. Sistemas Formales
5. Implementación

Ciclo de vida del software

El término ciclo de vida del software describe el desarrollo de software, desde la fase inicial hasta la fase final. Estos programas se originan en el hecho de que es muy costoso rectificar los errores que se detectan tarde dentro de la fase de implementación. El ciclo de vida permite que los errores se detecten lo antes posible y, por lo tanto, permite a los desarrolladores concentrarse en la calidad del software, en los plazos de implementación y en los costos asociados.



Procedimiento/Resultados:

1. Explicar la precondiciones y el conjunto de salidas de los algoritmos para:

1.1 Pescar

Precondiciones: Elegir un lugar adecuado ya sea un río o un lago, tener una lancha o canoa, tener una caña para pescar, tener carnada puede ser harina o masilla de pan, contar con paciencia y mucha actitud.

Salida: {Pescar, No pescar}

1.2 Lavarse las manos

Precondiciones: Agua, Jabón, Lavabo, Toalla, Manos

Salida: {Manos Limpias, Manos mal lavadas, Manos sucias}

1.3 Cambiar una llanta

Precondiciones: Estar en una superficie plana; Tener un carro, llanta de repuesto, una cuña, un gato hidráulico, una llave cruz.

Salida: {Coche con nueva llanta}

1.4 Convertir un número binario a decimal

Precondiciones: Tener un número binario, saber la tabla de potencias de dos, saber multiplicar

Salida: {Número decimal con igual cantidad de cifras como de bits del binario}

2. Desarrollar los algoritmos para:

2.1 Determinar si un número es positivo o negativo

Precondiciones: Número real

Inicio

Escribir "Ingresa el número a determinar"

$N = \text{Real}$

Leer número

Si número es > 0

Escribir "El número es positivo"

Si número es < 0

Escribir "El número es negativo"

Si número es $= 0$

Escribir "El número es neutro"

Fin

Salidas {n es positivo, n es negativo, n es neutro, n no es real}

2.2 Obtener el mayor de dos números

Precondiciones: tener dos números reales

Inicio

$n1, n2 = \text{real}$

Lea $n1, n2$

Si $n1 > n2$

Mayor = $n1$

Sino $n2 > n1$

Mayor = $n2$

Sino $n2 = n1$

Imprima "el número mayor es"

Imprima mayor

Fin

Salidas: { $n1$ es mayor, $n2$ es mayor, $n1 = n2$, $n1$ y/o $n2$ no son reales}

2.3 Obtener el factorial de un número.

Precondiciones: n es un número entero positivo o $n=0$, f es un número menor o igual a -1 .

Inicio

Fijar $n = \text{entero positivo}$

Fijar $f = \text{real}; f < -1$

Fijar respuesta = real

Escribir "ingrese un número"

Leer n

Si $n = 0$, escribir el factorial de n es igual a 1

Si $n = 1$, escribir el factorial de n es igual a 1

Sino multiplicar $n \times (n+f) = n(n-1)$

Luego $\times (n-2)$

Y así sucesivamente hasta que $(n+f)=1$

"Imprimir la factorial de n es"

Respuesta

Fin

Salidas: {Resultado de factorial de n, n no es ni positivo entero ni 0}

3. Verifica los algoritmos anteriores al ejecutarlos paso a paso con los siguientes números

3.1 54, -9, -14, 8, 0

- Inicio
Escribir "Ingresa el número a determinar"
N = 54
Leer número
54 es > 0
SI
"El número es positivo"
Fin
- Inicio
Escribir "Ingresa el número a determinar"
N = -9
Leer número
-9 > 0
NO
-9 < 0
SI
"El número es negativo"
Fin
- Inicio
Escribir "Ingresa el número a determinar"
N = -14
Leer número
-14 > 0
NO
-14 es < 0
SI
"El número es negativo"
Fin
- Inicio
Escribir "Ingresa el número a determinar"
N = 8
Leer número
8 es > 0
SI
"El número es positivo"
Fin

➤ Inicio
 Escribir "Ingresa el número a determinar"
 $N = 0$
 Leer número
 $0 > 0$
 NO
 $0 < 0$
 NO
 $0 = 0$
 SI
 "El número es neutro"
 Fin

3.2 (4,5), (-9,16), (127,8+4i), (7,m)

➤ Inicio
4, 5 = real
 Lea n_1, n_2
 Si $4 > 5$
 NO
 Sino $5 > 4$
 SI
 Mayor =5
 "el número mayor es"
 5
 Fin

➤ Inicio
-9, 16 = real
 Lea n_1, n_2
 Si $-9 > 16$
 NO
 Sino $16 > 9$
 SI
 Mayor =16
 "el número mayor es"
 16
 Fin

➤ Inicio
127, 8+4i = real
 Lea n_1, n_2
 n_2 no es un número real
 Fin

➤ Inicio
7, m = real
 Lea n_1, n_2
 M no es un número real
 Fin

3.3 5, 9, 0, -3

➤ Inicio

Fijar 5 = entero positivo

Fijar $f = \text{real}; f < -1$

Escribir "ingrese un número"

Leer n

Si $5 = 0$,

NO

Si $5=1$,

NO

Multiplicar $5 \times (n+(-1)) = 5(5-1)$

Luego $5 (5-2)$

Y así sucesivamente hasta que $(n+f)=1$

$5(4)(3)(2)(1)$

"Factorial de n es"

12

Fin

➤ Inicio

Fijar 9 = entero positivo

Fijar $f = \text{real}; f < -1$

Escribir "ingrese un número"

Leer n

Si $9= 0$

NO

Si $9=1$

NO

Sino multiplicar $n \times (n+f) = 9(9-1)$

Luego $x (9-2)$

Y así sucesivamente hasta que $(9+f)=1$

$9(8)(7)(6)(5)(4)(3)(2)(1)$

"Factorial de n es"

362880

Fin

➤ Inicio

Fijar 0 = entero positivo

Fijar $f = \text{real}; f < -1$

Escribir "ingrese un número"

Leer n

Si $0= 0$,

"el factorial de n es igual a 1"

Fin

➤ Inicio

Fijar -3 = entero positivo

Fijar $f = \text{real}; f < -1$

Escribir "ingrese un número"

Leer n

$n=-3$ no es un entero positivo

Fin

4. Desarrollar algoritmos propios de un procesador (asignando registros genéricos) para:

4.1 Cambiar el signo de un número binario

Precondiciones: Tener un número binario, conocer que 0=+ y 1= -

Tomar número binario de registro 1
Observar bits de derecha a izquierda
Escribir bit opuesto debajo de cada uno a manera de columna
Determinar si es positivo o negativo
Guarda resultado en registro 2

Salidas: {Número binario positivo, número binario negativo}

4.2 Hacer una suma larga binaria

Precondiciones: tener dos números binarios largos, conocer las condiciones de $1+0=1$; $0+0=0$; $0+1=1$; $1+1=10$

Tomar un número binario del registro f
Tomar otro número binario del registro k
Escribir un número binario debajo del otro de manera que queden alineados los últimos bits
Empezar la operación de derecha a izquierda por columnas
Escribir el resultado en una tercera fila debajo de cada columna correspondiente
Si tuviéramos un resultado de 10 escribimos 0 en la fila de resultado y llevamos 1 a la columna de la izquierda para sumarlo con los bits de dicha columna
Así sucesivamente hasta llegar a la última columna
Guardar resultado en registro z

Salidas {Número binario nuevo, 0}

Conclusiones:

Con esta práctica entendí como realizar un algoritmo, comprobarlo mediante ejemplos pero sobre todo lo que más me llamo la atención es como todo lo que hacemos en nuestra vida puede ser representado mediante un algoritmo.

Me costó un poco de trabajo el algoritmo de factorial de un número pues es una operación más compleja que no es igual para todos los números, pero trate de resolverla con mi propio lenguaje.

En conclusión me gusta como estamos aprendiendo paso a paso y con ejemplos tomados de la cotidianidad que hacen las cosas más fáciles de entender.

Bibliografía:

https://drive.google.com/drive/folders/1LLKjvk9PL5AgUSLU9FRkOOMYdQM2-9_Q