



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorio de Computación Salas A y B

Profesor(a):

Manuel Castañeda Castañeda

Asignatura:

Fundamentos de programación

Grupo: 18

No de practica(s):

03

Integrante(s):

Velasco Molina Ricardo Alonso

No de lista o brigada: 53

Semestre:

2025-1

Fecha de entrega:

02 de septiembre de 2024

Observaciones:

Calificación:

Diseño de algoritmos.

Cuestionario previo, definir...

1. ¿Qué es un algoritmo?

Son instrucciones, estas deben de llevar un orden en forma de lista, ya que de eso dependen toma de decisiones y pasos a seguir de un proceso, cada característica se define para tomar en cuenta la función que cumple dentro del proceso.

2. Que características tiene un algoritmo.

Los algoritmos deben de ser precisos, llevar un orden especificado paso a paso, este proceso debe de tener un fin y ser exacto, es decir, debe de cumplir con un resultado.

3. ¿Cuáles son las etapas de diseño de un algoritmo?

- La entrada, en esta se escriben los datos que serán procesados.
- Proceso, aquí se realizan todas las operaciones que sean necesarias para darle solución al problema.
- Salida, esta es la parte final, la cual muestra los resultados.

Desarrollo de la práctica.

Elaborar 16 algoritmos.

1.- Obtener el área de un círculo.

Entrada: Radio (r)

Salidas: Área

1. Inicio
2. Definir área "a" = $\pi * r^2$
3. Mostrar "radio", (r)
4. Calcular "a"
5. Mostrar "El resultado del área es " "a"
6. Fin

2.- Obtener la resistencia de un circuito eléctrico.

Entradas: Intensidad (W) Voltaje (V)

Salidas: Resistencia

1. Inicio
2. Definir Resistencia = I/V
3. Mostrar "Ingrese el voltaje"
4. Leer "V"
5. Mostrar "Ingrese la intensidad"
6. Leer "I"
7. Calcular "R"
8. Mostrar "El voltaje del circuito es " "R"
9. Fin

3.- Algoritmo para obtener la velocidad de un automóvil a velocidad constante.

Entradas: Distancia (m), Tiempo (s)

Salidas: Velocidad

1. Inicio
2. Mostrar "Ingrese la distancia recorrida en metros"
3. Leer Distancia (m)

4. Escribir "Ingrese el tiempo que tardó en recorrer esa distancia en segundos"
5. Leer Tiempo (s)
6. Velocidad (m/s) = (m)/(s)
7. Mostrar "La velocidad del automóvil es:", Velocidad, "(m/s)".
8. Fin

4.- Obtener la Fuerza de gravedad en CU.

Entradas: Masa 1 "M1", Masa 2 "M2"

Salidas: Fuerza de gravedad (F)

1. Inicio
2. Definir valor de $CGU = 6.6Nm^2 kg^{-2}$
3. Mostrar "introduzca distancia 1 y 2"
4. Leer las masas de objetos, (m1 y m2)
5. Leer la distancia entre m1 y m2
6. Calcular $CGU = 6.6Nm^2 kg^{-2}$
7. Calcular Fuerza de gravedad (F) = $G * (m1 * m2) / r^2$
8. Mostrar "la fuerza de gravedad es" F
9. Fin

5. Obtener el equivalente a grados F a partir de grados C.

Entradas: Celsius, "C"

Salidas: Fahrenheit, "F"

1. Inicio
2. Definir = $(Celsius * 9/5) + 32$
3. Mostrar "Ingresa la temperatura en grados celsius "
4. Leer "C"
5. Calcular "F"
6. Mostrar "La temperatura en grados Farenheit es ", F, "Grados Farenheit".
7. Fin

6. Obtener el equivalente entre dólares y pesos.

Entradas: solares, tasaCambio

Salidas: pesos

1. Inicio
2. Escribir "Ingrese la cantidad en dólares:"
3. Leer dólares
4. Escribir "ingrese la tasa de cambio (pesos por dólar):"
5. Leer tasaCambio
6. $\text{pesos} = \text{dolares} * \text{tasaCambio}$
7. Escribir "El equivalente en pesos es:", pesos, "MXN".
8. Fin

7.- Obtener el mayor de entre tres números, indicando si son iguales.

Entradas: $a1 = n1$, $b2 = n2$

Salida: $c3 = n3$

* mayor: el número más grande entre a, b y c

* iguales: un booleano que indica si a, b y c son iguales

Salidas:

8.- Algoritmo para el valor absoluto de un número.

Entradas $N = \text{número}$

Salidas

Valor absoluto

Inicio

1. Mostrar "dame un número"
2. Leer n

3. Si $n > 0$
4. Mostrar "el valor absoluto es" n
5. Si $n < 0$
6. Multiplicar $n(-1)$
7. Mostrar "el valor absoluto es" n
8. Fin

9.- A partir de un número si es par obtener su cuadrado y si es impar obtener su raíz cuadrada.

Entradas: Número (n)

Salidas: Cuadrado, Raíz cuadrada

1. Inicio
2. Leer número
3. Si el número es par entonces
 - * Calcular cuadrado = n^2
 - * Imprimir cuadrado
4. Si es impar entonces
 - * Calcular raíz cuadrada = $n^{1/2}$
 - * Imprimir raíz cuadrada
5. Fin

10.-obtener la raíz de un polinomio de 2º grado con la fórmula general.

Entrada: coeficientes del polinomio $ax^2+bx+c=0$

Salida: Raíces del polinomio (x_1 y x_2)

- 1.leer los coeficientes: asigna los valores de a , b y c Que corresponde al polinomio
- 2.calcular el discriminante (D):
 - utilizar la fórmula: $D=b^2-4ac$
 - si D es negativo, las raíces serán complejas.
- 3.Evaluar el discriminante:
 - Si $D > 0$ el polinomio tiene dos raíces reales distintas.
 - Si $D = 0$ el polinomio tiene una raíz doble.
 - Si $D < 0$ el polinomio tiene dos raíces complejas.

4. Calcular las raíces utilizando la fórmula general:

- Utiliza las siguientes fórmulas dependiendo del valor de D:

- Si $D > 0$, $(-b \pm \sqrt{D}) / (2a)$

- Si $D < 0$ $(-b) / (2a) \pm (\sqrt{D}) / (2a)$

5. Mostrar las raíces x_1 y x_2

6. Fin

11.- Aquí te presento un algoritmo para una calculadora básica que puede sumar, restar, multiplicar y dividir dos números, con la validación de no dividir entre cero.

Entradas: num1: 10, num2: 2, Op: división

Salidas: Resultado: 5

Entradas: num1: 10, num2: 0, Op: división

Salidas: Error: No se puede dividir entre cero

Entradas: num1: 5, num2: 3, Op: suma

Salidas: Resultado: 8

Entradas: num1: 7, num2: 2, Op: resta

Salidas: Resultado: 5

Entradas: num1: 4, num2: 5, Operación: multiplicación

Salidas: Resultado: 20

1. Pedir al usuario que ingrese el primer número (num1)

2. Pedir al usuario que ingrese el segundo número (num2)

3. Pedir al usuario que seleccione la operación deseada (suma, resta, multiplicación, división)

4. Realizar la operación seleccionada:

- Suma: resultado = num1 + num2

- Resta: resultado = num1 - num2

- Multiplicación: resultado = num1 * num2

- División:

- Verificar si num2 es distinto de cero

- Si es cero, mostrar error y terminar

- Si no es cero, resultado = num1 / num2

5. Mostrar r

6. Fin

12.- Sumatoria de los primeros 16 números pares.

Salida: Suma de los primeros 16 números pares

1. Inicio
2. Definir una variable para almacenar una suma "as"
3. Definir la variable "suma" = "0"
4. Definir la variable "contador" = "0"
5. Definir una variable llamada "numero" = "0"
6. Definir una variable para los números pares
7. Repetir solo si "contador" = o < "16" sumar el valor de "numero" a "suma"
8. Aumentar valor de "numero" en "2"
9. "numero par" + "as"
10. Aumentar valor de "contador" en "1"
11. Detener "suma" si "contador" > "16"
12. Mostrar valor de "suma"
13. Fin

13.- No debo faltar a clases 100 veces.

Entrada: ninguna

Salidas: "no debo faltar a clases"

1. Inicio
2. Definir $n=0$
3. Mostrar " No debo faltar a clases"
4. $n=n+1$
5. Si $n < 100$ entonces repetir paso 3
6. si no Fin

14.- Gestor de Contraseñas.

Entradas: contraseñaCorrecta, contraseñaIngresada, intentos

1. Inicio
2. contraseñaCorrecta=EstudianteFI
3. intentos=0
4. Mientras intentos < 3 Hacer
5. Escribir "Ingresa la contraseña:"

6. *Leer contraseña ingresada.*

7. Si $\text{contraseñaIngresada} = \text{contraseñaCorrecta}$ Entonces

8. Escribir "Acceso concedido"

9. Fin

10. Sino

11. $\text{Intentos} = \text{intentos} + 1$

12. Escribir "Contraseña incorrecta "

13. FinSi

14. FinMientras

15. Fin

15.-Algoritmo número mágico.

Entradas: Intentos =10, N=80

Salidas: NM

1.Inicio

2.Mostrar "dame un número"

3.Leer a

4.Si $a=80$ ir a paso 17

5.Si $a>80$

6.Mostrar "el número es muy alto"

7.Restar -1 a intentos

8.Si " $\text{intentos} > 0$ "

9.Regresar a paso 2

10.Si " $\text{intentos} < 0$ " ir a paso 17

11.Si $a < 80$

12.Mostrar "el número es muy bajo"

13.Restar -1 intentos

14.Si " $\text{intentos} > 0$ "

15.Regresar a paso 2

16.Si " $\text{intentos} < 0$ "

17.Fin

16.- Calculadora de dos números donde se puede reiniciar la calculadora.

Entradas; Sumar (sum), Número 1 (n1), Número 2 (n2), Reiniciar (r), Restar (res), Multiplicar (m), Dividir (d)

Salidas: Resultado, Calculadora reiniciada

1. Inicio

2. Leer variables $n1 = 0$, $n2 = 0$, resultado = 0, reiniciar = False

3. Menú de opciones

Mostrar "1. Sumar"

Mostrar "2. Restar"

Mostrar "3. Multiplicar"

Mostrar "4. Dividir"

Mostrar "5. Reiniciar"

Mostrar "6. Salir"

4. Leer números

$n1 = \text{ingresar}(\text{"Ingrese el primer número: "})$

$n2 = \text{ingresar}(\text{"Ingrese el segundo número: "})$

5. Realizar operación

Si op "1":

$\text{resultado} = n1 + n2$

Si op "2":

$\text{resultado} = n1 - n2$

Si op "3":

$\text{resultado} = n1 * n2$

Si op "4": $n2 \neq 0$:

$\text{resultado} = n1 / n2$

Si no

Mostrar "Error: Dividiste por cero"

6. Mostrar Resultado

Si i = "5" y i = "6":

Mostrar "El resultado es: ", resultado

7. Para reiniciar calculadora

reiniciar: $n1 = 0$, $n2 = 0$

resultado = 0

Sino: reiniciar = Falso

8. Fin

Conclusiones.

El trabajo nos sirvió para identificar las partes necesarias para que funcione un algoritmo, ya que partiendo de la base de su escritura comprendemos porque es esencial el listado de todos los pasos del proceso, así como de la toma de decisiones y como cada una corresponde a un fin y resultado logable con los algoritmos.

Bibliografía.

Raghu Singh (1995). International Standard ISO/IEC 12207 Software Life Cycle Processes. Agosto 23 de 1996, de ISO/IEC. Consulta: Junio de 2015. Disponible en: <http://www.abelia.com/docs/12207cpt.pdf>

Carlos Guadalupe (2013). Aseguramiento de la calidad del software (SQA). [Figura 1]. Consulta: Junio de 2015. Disponible en: <https://www.mindmeister.com/es/273953719/aseguramiento-de-la-calidad-delsoftware-sqa>

Andrea S. (2014). Ingeniería de Software. [Figura 2]. Consulta: Junio de 2015. Disponible en: <http://ing-software-verano2014.blogspot.mx>

Michael Littman. (2012). Intro to Algorithms: Social Network Analysis. Consulta Junio de 2015, de Udacity. Disponible en: <https://www.udacity.com/course/viewer#!/c-cs215/l-48747095/m-48691609>