	Carátula para entrega de prácticas
Facultad de Ingeniería	Laboratorio de docencia

Laboratorios de computación salas A y B

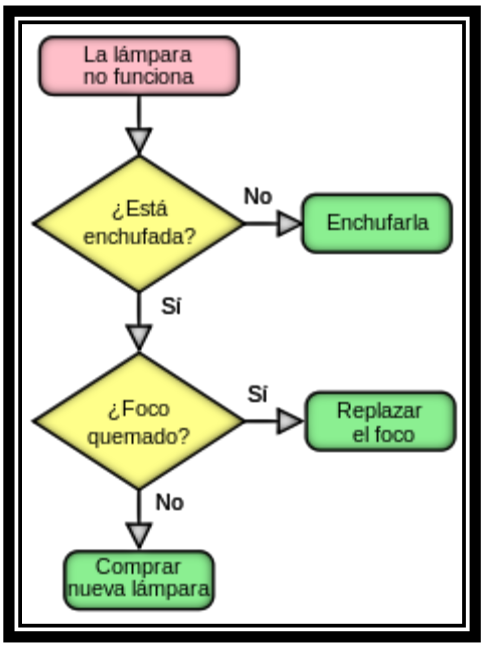
Profesor:	ALEJANDRO PIMENTEL
Asignatura:	FUNDAMENTOS DE PROGRAMACIÓN
Grupo:	BLOQUE 135
No de Práctica(s):	PRACTICA 3 “Solución de problemas y algoritmos”
Integrante(s):	ALITZEL ANAID GUTIÉRREZ RAMOS
No. de Equipo de cómputo empleado:	
No. de Lista o Brigada:	9370
Semestre:	1er SEMESTRE
Fecha de entrega:	02-09-2019
Observaciones:	Excelente. Te recuerdo que todo buen reporte escrito debe llevar introduccion y conclusiones o secciones equvalentes (como tu "DEFINICION")

CALIFICACIÓN: **10**

OBJETIVOS

Conocer, elaborar, explorar y practicar diferentes tipos de algoritmos, así como su significado ya que son más que importantes e indispensables desde nuestra vida cotidiana y el ciclo de vida del software.

DEFINICIÓN



Algoritmo. En Matemática, ciencias de la Computación y disciplinas relacionadas, un algoritmo (del latín, dixit algorithmus y éste a su vez del matemático persa Al Juarismi) es un conjunto reescrito de instrucciones o reglas bien definidas, ordenadas y finitas que permite realizar una actividad mediante pasos sucesivos que no generen dudas a quien lo ejecute. Dados un estado inicial y una entrada, siguiendo los pasos sucesivos se llega a un estado final y se obtiene una solución. Los algoritmos son objeto de estudio de la algoritmia.

CONCEPTO

Conjunto reescrito de instrucciones o reglas bien definidas, ordenadas y finitas que permite realizar una actividad mediante pasos sucesivos que no generen dudas a quien lo ejecute.

DIAGRAMA DE FLUJO

Los diagramas de flujo son descripciones gráficas de algoritmos; usan símbolos conectados con flechas para indicar la secuencia de instrucciones y están regidos por ISO. Son usados para representar algoritmos pequeños, ya que abarcan mucho espacio y su construcción es laboriosa. Por su facilidad de lectura son usados como introducción a los algoritmos, descripción de un lenguaje y descripción de procesos a personas ajenas a la computación.

DIAGRAMAS RECTANGULARES ESTRUCTURADOS

Una de las dificultades de los Diagramas de Flujo radica en que así como brinda la posibilidad de representar gráficamente el flujo de la solución a un problema también abre el espacio para que un programador desordenado ponga flechas de flujo inadecuadamente y finalmente obtenga una representación más compleja que la idea misma. Precisamente la técnica de Diagramas Rectangulares Estructurados también permite usar herramientas gráficas para representar la solución a un problema con la ventaja de que no brinda la posibilidad de que seamos desordenados en nuestra concepción. Se basa en representar todo el algoritmo dentro del marco de un rectángulo y a diferencia de la técnica anterior, se mueve básicamente con la utilización de tres símbolos que corresponden a cada una de las estructuras básicas de la lógica de programación.

https://www.ecured.cu/Algoritmo

ACTIVIDADES DESARROLLADAS

1. Empecé explicando las precondiciones y el conjunto de salidas de los algoritmos para:

a) Pescar.

Precondiciones:

- Zona apta para llevar esta actividad (***Ejemplo:*** lago, río, laguna, mar, etcétera).
- Vehículo en que transportarse y estar a salvo (***Ejemplo:*** barco, lancha, canoa, etcétera).
- Anzuelos, dependiendo lo que se va a pescar.
- Caña de pescar.
- Boyas para pescar.
- Algún costal o recipiente para colocar el pez o los peces capturados.

Salidas:

- Pescar.
- No pescar.

b) Lavarse las manos.

Precondiciones:

- Manos.
- Lavabo
- Agua.
- Jabón.
- Toalla.

Salidas:

- Manos limpias.
- Manos sucias.
- Manos no muy bien lavadas.

c) Cambiar una llanta.

Precondiciones:

- Estar en un lugar seguro
- Una llanta de repuesto.
- Un carro.
  Herramientas;
- Un gato hidráulico.
- Llave de cruz.
- Una cuña

Salidas:

- Carro funcional con llanta nueva.

d) Convertir número binario a decimal.

Precondiciones:

- Tener conocimientos básicos de la multiplicación.
- Saber que es un numero binario.
- Tener un numero binario.
- Saber que es decimal.
- Conocer la tabla de potencia 2.

Salidas:

- Numero binario ahora decimal.

2. Desarrolle estos algoritmos:

a) Determinar si un número es positivo o negativo.

Precondiciones:

- Tener un número real.
- “Ingresa el número real”.
  N = Real.
- “El número es positivo”.
  >0
- “El número es negativo”.
  <0
- “El número es neutro”.
  = 0

Salidas:

- N no real.
- N neutro.
- N positivo.
- N negativo.

b) Obtener el mayor de dos números.

Precondiciones:

- Tener dos números reales.
- “Ingresa el número real”.
  N1, N2 = Real.
- “N1 es mayor y N2 es menor”.
  N1>N2
- “N2 es mayor y N1 es menor”.
  N1<N2
- “El número mayor es”.
  N2=N1
- Da el numero mayor.

Salidas:

- N1 y N2 no reales.
- N1=N2 neutros.
- N1 mayor.
- N2 menor.

c) Obtener el factorial de un número.

Precondiciones:

- F= número real menor o igual a -1.
  F< -1
- N= número entero positivo o igual a 0.
- Fijar respuesta “=real”.
- “Ingrese un número”.
- El factorial de n es igual a 1.
  N=0
  N=1
- Multiplicar:
  N . (N+F) = 1
  N . (-2)
- Hasta llegar a:
  (N+F) = 1
- “Imprimir factorial de n”.

Salidas:

- Resultado factorial de N
- N no será positivo entero ni cero.

3. Verifique los algoritmos anteriores, ejecutándolos con los siguientes valores:

a) 54, -9, -14, 8, 0.

1. "Ingresa el número a determinar". ***2. N = 54*** 3. 54 > 0 4. SI 5. "El número es positivo".	1. "Ingresa el número a determinar". ***2. N = -9*** 3. -9 > 0 4. NO 5. -9 < 0 6. SI 7. "El número es negativo".
1. "Ingresa el número a determinar". ***2. N = 8*** 3. 8 > 0 4. SI 5. "El número es positivo".	1. "Ingresa el número a determinar". ***2. N = -14*** 3. -14 > 0 4. NO 5. -14 < 0 6. SI 7. "El número es negativo".
1. "Ingresa el número a determinar". ***2. N = 0*** 3. 0 > 0 4. NO 5. 0 < 0 6. NO 7. 0 = 0 8. SI 9. "El número es neutro".	

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

1. 4, 5 = real ***2. N1 y N2.*** 3. 4 > 5 4. NO 5. 5 > 4 6. SI 7. 5 = mayor 8. "El número mayor es 5".	***1. -9, 16 = real*** ***2. N1 y N2.*** 3. -9 > 16 4. NO 5. 16 > 9 6. SI 7. 16 = mayor 8. "El número mayor es 16".

1. *127, 8+4i = real* 2. *N1 y N2.* 3. N2 no es número real.	1. *7, m = real* 2. *N1 y N2.* 3. M no es número real.

c) 5, 9, 0, -3.

1. *NUMERO POSITIVO = 5* 2. Real = f 3. F < -1 4. "Ingrese un número". 5. 5 = 0 6. NO 7. 5 = 1 8. NO 9. Multiplicar: 5 x (N+ (-1)) = 5 (5-1) 5 (5-2) 10. Hasta que (N+F) = 1 5 (4)(3)(2)(1) 11. "Factorial de n = 12"	1. *NUMERO POSITIVO = 9* 2. Real = f 3. F < -1 4. "Ingrese un número". 5. 9 = 0 6. NO 7. 9 = 1 8. NO 9. Multiplicar: 9 x (N+ (-1)) = 9 (9-1) 9 (9-2) 10. Hasta que (N+F) = 1 9 (8)(7)(6)(5)(4)(3)(2)(1) 11. "Factorial de n = 362880"
1. *NUMERO POSITIVO = 0* 2. Real = f 3. F < -1 4. "Ingrese un número". 5. 0 = 0 6. "Factorial de n = 1"	1. *NUMERO POSITIVO = -3* 2. Real = f 3. F < -1 4. "Ingrese un número". 5. N = -3 6. "No es un número positivo"

4. Desarrolle algoritmos propios de un procesador para:

a) Cambiar el signo de un número binario.

Precondiciones:

- Tener un número binario.
- 0 = +
- 1 = -

Algoritmo:

- Tomar número binario 1.
- Derecha – Izquierda ver bits.
- En forma de columna escribir de cada valor el bit opuesto.
- Ver si es positivo o negativo.
- Recopilar el resultado en registro 2.

Salidas:

- Número binario +
- Número binario –

b) Hacer una suma binaria larga.

Precondiciones:

- Tener dos números binarios largos.
- 0+0 = 0
- 1+0 = 1
- 0+1 = 1
- 1+1 = 10

Algoritmo:

- Tomar número binario F.
- Otro número binario K.
- Escribir alineadamente uno arriba y otro abajo.
- Derecha – Izquierda ver bits.
- El resultado ira en la tercera columna.
- Cuando el resultado tiene dos cifras un numero se pone abajo el de lado izquierdo se sube arriba de la siguiente cifra.
- En forma de columna escribir de cada valor el bit opuesto.
- Ver si es positivo o negativo
- Recopilar el resultado en registro Z.

Salidas:

- Nuevo número binario = 0