|  |  |  |
| --- | --- | --- |
| Dato a almacenar | Variable | |
| Identificador | Tipo de dato |
| * Placa de un vehículo * Tamaño del motor en centímetros cúbicos * Número de pasajeros * Número de baños de una casa * Área de la casa en metros ´ * Valor del alquiler * Valor del descuento de un producto * ¿Encendió el computador? * Número de matrícula del estudiante * Valor de la matrícula del estudiante | * Variable * Variable * Variable * Variable * Variable * Constante * Variable * Constante * Constante * Constante | * Entero * Flotante * Float * Float * Float * Float * Float * cadena * Float * Float |

Ejercicios propuestos

Capítulo 1

1. Imagine que se desea declarar variables para almacenar datos de acuerdo con el enunciado de la primera columna de la Tabla 1.16 que aparece a continuación. Diligencie la segunda y tercera columna, declarando un identificador válido y adecuado para la variable, con su respectivo tipo de dato.

2. Para las variables que se declaran en la Tabla 1.17, diga si el identificador utilizado es correcto o incorrecto y justifique por qué.

|  |  |  |
| --- | --- | --- |
| Variable | ¿Es válido? | Justifique |
| Nombre medico  @especialidad  generoAspirante  Valor para pagar  salarioEmpleado1  #CEDULA  Titulo Libro  títuloLibro  Años de Experiencia esCasado | No  No  Si  No  Si  No  Si  Si  No | Espaciado el cual genera dos variables  Uso carácter especial  Variable bien declarada  Espaciado el cual genera dos variables  Identificación completa del dato  Uso incorrecto de caracteres especiales  Identificación completa de la variable  Identificación correcta  Al involucrar varios espacios crea más de una variable |

3. Describa, de acuerdo con la precedencia de los operadores, en cual orden se ejecutan las siguientes expresiones:

a) resultado = PI \* radioˆ2

1er. Potenciación (radioˆ2) 2ndo. Multiplicación PI\*(radio^2)

b) resultado = 2 \* a + 3 \* b - c

1er. Multiplicación (2\*a) +(3\*b)-c 2ndo. Suma/resta

c) resultado = 2 \* ( a + 3 ) \* b - c

1er. Suma paréntesis (a+3) 2ndo. Multiplicaciones 2\*(3+a)\*b 3er.resta 2\*(3+a)\*b -c

d) resultado = aˆ2 - b \* 36ˆ(1/2)

1er. Potenciación a^2 y 36^(1/2) 2ndo.multiplicacion de b\*36^(1/2) 3er. Suma/resta a^2\*-b\*36^(1/2)

e) resultado = ( a + b ) / ( 2 \* c + 1 - a %3)

1er. Paréntesis (a+b) y (2\*c+1-a%3) 2nd. División a%3 3rd. Multiplicación 2\*c 4th. Sumas y restas dentro de paréntesis (2\*c+1-a%3) 5th. División de resultados de paréntesis = (a + b) / ( 2 \* c + 1 - a %3)

4. Resuelva paso a paso las siguientes expresiones, teniendo en cuenta la prioridad de ejecución de los operadores.

a) 3 \* (3 + 4) \* (5 - 2)

3\*(7) \*(3) à 63

b) (3ˆ (2 + 3) - 1) ˆ (1.0/2.0)

(3^ (5)-1) ^ (0.5) à (243-1)^(0.5) à 242^0.5 à 15.58

c) 5.0/2.0 \* 3 + 4 ˆ 3.0 \* 2/ (5.0 + 2.0) - 2

5.0/20\*3+4^3.0\*2/ (7)-2 à 5.0/2.0\*3+64\*2/7-2 à 2.5\*3+64\*0.4 à 7.5+25.6 à 33.1

d) 10 \* (7 + 7) % (9 + 2) /10

10\*(14) % (11) /10 à 140%1.1 à 1.54

e) 3 % 2 - (2 + 2) / 1 \* (3 + 1) + 3

3%2-(4)/1\*(4)+3 à 3%2/7 à 1.5/7 à 0.21

f) (5 + 8\*2 - 3.0/5.0) /(4\*1 - 2.0/3.0 + 8/2)

(5+16-3.0/5.0)/(4-2.0/3.0+8/2) à (19/5.0)/(2/3.0+8/2) à (3.8)/(0.6+4) à 3.8/4.06 à 0.82

g) (4 + 8) /(4 /(1 + 1)) - (3 + 2 + 1)/(5ˆ2+4)

(8)/(4/(2))-(6)/(14) à 8/(4)-0.42 à 2-0.42 à 1.58

5. Suponga que se requieren las variables: a, b, c, d de tipo real.

a) Declare estas variables. à Int a,b,c

b) Asígnele un número cualquiera entre 10 y 20 a cada variable. à a=12, b= 17, c=15 d=19

c) Evalúe las siguientes expresiones, teniendo en cuenta los valores asignados:

a = 3 \* b + d %5 à a=3\*17+19%5 à a= 51+0.95 à a=51.95

d = (4 \* a/2 - 3\*c) / (4 + b %3) à (4\*12/2-3\*15) à (48/2-45) à (48/43) à d=1.11

c = 3 \* bˆ2 - 5 \* c/3 à 3\*17^2-5\*15/3 à 3\*289-75/17 à 867-75/17 à 792/17 à c=46.5

d = 2 \* a + 3 \* b + 4 \* c/d à 2\*12+3\*17+4\*15/19 à 24+51+60/19 à 135/19 à 78.15

a = 3 \* a à 3\* 12 à a=36

b = 7 + a à 7+12 à b=19

c = b - a à 17-12 à c=5

d = c + a - b à 15+12-19 à d= 8

6. Se tienen las variables: a, b, c y d de tipo entero. A partir de las asignaciones que se dan enseguida, determine si las siguientes expresiones entregan como resultado un valor verdadero o falso:

a = 5 b = 4 c = 7 d = 3

a) 3 \* a <= 15 true

b) c - d == a + 2 true

c) (5 \* a + 3 \* b) >= (cˆd - 35) à 25+12>= (343- 35) à 37>=308 à false

d) (5 % c + 3 \* b/d) < (cˆd/3)/76 Fundamentos à 5%7+3\*4/3<(7ˆ3/3)/76 à 0.35+12/3<(343)/76 à 4.11<4.5 false

e) (32 % (4 \* c) + 3 \* (b + d)) < (cˆ(1 + a % 2)) à (32%(4\*7)+3\*(4+3))<(7^(1+5%2)) à (32%(28)+3\*(7))<(7^(0.12)) à 8.96+21<1.65 à 29.96<1.65 False

f) (5 \* d + b != 4ˆd) O (c + d >= a/b) à (5\*3 + 4 != 4ˆ3) O (7 + 3 >= 5/4) à (19 ¡= 12 ) O (10>=1.25)

True

g) (c \* a + 10 > b \* d - 6) Y ( (b + c) % 5 b\*<a) à (7 \* 5 + 10 > 4 \* 3 - 6) Y ( (4 + 7) % 5 \*4<5) à (45+10>12-6) Y (11%20<5) à 55>6 Y 2.2<5 True

h) (c\*(a + 10) > b\*(8\*d-6)) Y ((b + c) % 5 < a) à (7\*(5 + 10) > 4\*(8\*3-6)) Y ((4 + 7) % 5 < 5) à (7(15)>4\*(18) Y ((11)%5<5) à 105>72 Y (0.55<5) False

I) (c\*(a + 10) > b\*(8\*d-6)) Y ((b + c) % 5 < a) à (7\*(5 + 10) > 4\*(8\*3-6)) Y ((4 + 7) % 5 < 5) à 7\*15>4\*18 Y 0.55<5 à 105>72 Y 0.55<5 True

J) (c\*a+7 >b \* d - 3) Y ( (b+c)ˆ(1/2) à (7\*5+7 >4 \* 3 - 3) Y ( (4+7)ˆ(1/2) à (45+7 >12-3) Y ( (11)ˆ(0.5) à 52>9 Y 3.3 True

k) NO (a == b) à NO (5 == 4) True

l) NO (a \* b + c != d) à NO (5 \* 4 + 7 != 3) à NO (27 != 3) True

7. Escriba en notación algorítmica las siguientes expresiones matemáticas:

A picture containing diagram

Description automatically generated

int main() {

float a,b,c;

float division9,raíz,sima;

printf("%s\n",("El número para a:"));

scanf("%f",&a);

printf("%s\n",("El número para b:"));

scanf("%f",&b);

printf("%s\n",("El número para c:"));

scanf("%f",&c);

raiz = sqrtf((b\*2)-(4\*a\*c));

suma = -b+raiz;

division = suma/2\*a;

printf("El resultado es: %f\n",division);

return 0;

}



#include<iostream>

using namespace std;

int main() {

float a;

float b;

float c;

float division;

float division2;

float resta;

float resultado;

float suma;

float suma2;

float suma3;

cout << ("El número para a:") << endl;

cin >> a;

cout << ("El número para b:") << endl;

cin >> b;

cout << ("El número para c:") << endl;

cin >> c;

suma = 3\*a+4\*b;

suma2 = 5+c;

division = suma/suma2;

resta = 8\*c-10;

suma3 = 3\*a+b;

division2 = resta/suma3;

resultado = division+division2;

cout << "El resultado es: " << resultado << endl;

return 0;

}

Text

Description automatically generated with low confidence

#include<iostream>

#include<cmath>

using namespace std;

int main() {

float a;

float b;

float c;

float d;

float division;

float division2;

float raiz;

float raizc;

float resultado;

float suma;

float suma2;

float x;

cout << ("El número para a:") << endl;

cin >> a;

cout << ("El número para b:") << endl;

cin >> b;

cout << ("El número para c:") << endl;

cin >> c;

cout << ("El numero para d:") << endl;

cin >> d;

suma = 4\*b+7;

suma2 = 10+sqrtf(4\*c);

division = suma/suma2;

x = (4\*b + 8)

raiz = 3\*a\*sqrt(3,x);

raizc = sqrtf(8\*d);

division2 = rais/division;

resultado = division2+raizc;

cout << "El resultado es:" << resultado << endl;

return 0;

}



Algoritmo

Definir b,a,c,Suma,Division,Suma2 Como Real

Inicio

Escribir ("El número para a:")

Leer (a)

Escribir ("El número para b:")

Leer (b)

Escribir ("El número para c:")

Leer (c)

Suma = 3b + a

Division = 5ab7 / Suma

Suma2 = Division + 2ac3/5

Escribir (“El resultado es:”Suma2)

FinAlgoritmo

Diagram

Description automatically generated with medium confidence

#include<iostream>

using namespace std;

int main() {

float a;

float b;

float c;

float division;

float suma;

float suma2;

cout << ("El número para a:") << endl;

cin >> a;

cout << ("El número para b:") << endl;

cin >> b;

cout << ("El número para c:") << endl;

cin >> c;

suma = 3\*b+a;

division = 5\*a\*b\*7/suma;

suma2 = division+2\*a\*c\*3/5;

cout << "El resultado es:" << suma2 << endl;

return 0;

}

8. Preguntas sobre los conceptos vistos

a) Defina con sus propias palabras los siguientes términos, vistos en este capítulo:

* Variable: Una variable es una propiedad que puede adquirir diferentes valores en un conjunto determinado y cuya variación es susceptible de ser medida.
* Constante: Según cómo se midan, las variables podrán ser cualitativas o cuantitativas. Serán cualitativas aquellas que expresen características o cualidades diferentes; y serán cuantitativas cuando expresen argumentos numéricos. Las variaciones cualitativas podrán ser clasificadas en: Ordinales o cuasi cuantitativas.
* Dato: Es un valor o referente que recibe el computador por diferentes medios, los datos representan la información que el programador manipula en la construcción de una solución o en el desarrollo de un algoritmo.
* Tipo de dato: Un **tipo de datos** es la propiedad de un valor que determina su dominio (qué valores puede tomar), qué operaciones se le pueden aplicar y cómo es representado internamente por el computador.
* Operador: Un operador, es un elemento que indica el tipo de operación que se le va a aplicar a uno o más datos.
* Expresión: Una **expresión** es una combinación de valores y operaciones que, al ser evaluados, entregan un valor.

b) Dado que existen diferentes tipos de datos, operadores y expresiones, haga una clasificación para:

Tipos de datos:

* Operadores aritméticos.
* Operadores de comparación.
* Operadores lógicos.
* Operadores de concatenación.
* Operadores especiales.

Expresiones:

- aritméticas

- lógicas

- relacionales

- alfanuméricas

- asignación.

Algoritmos.

* Algoritmos de búsqueda.
* Algoritmos de ordenamiento.
* Algoritmos voraces.
* Programación dinámica.
* Algoritmos probabilísticos.

a) Para los siguientes enunciados realice el análisis y el diseño del algoritmo (utilizando la forma de descripción narrada) que permita realizar las acciones que a continuación se listan:

* Adquirir un libro a través de una librería virtual. -ingresar a la página donde se realizará la compra, búsqueda de libro, si se encuentra, buscar opción para compra, si no, buscar otra página de libros que lo venda, buscar carrito, verificar métodos de pago, ubicar información, realizar pago, esperar a por el envió del libro.
* Descargar un vídeo de YouTube. -ingresar a YouTube, buscar nombre del video, encontrar video, copiar URL del video, abrir nueva pestaña, buscar “descargar video YouTube”, hacer click en el primer resultado, ubicar barra para pegar URL, pegar URL, esperar, click en descargar, ubicar el archivo en espacio del ordenador.
* Calcular cuánto dinero se gastará el día de mañana. -declarar base, calcular precios de comida, calcular gastos aparte de la comida, calcular ingresos externos, sumar ingresos externos con base, restar gastos de la comida y gastos aparte
* Invitar a un amigo a desayunar en la cafetería. – Buscar contacto en móvil, abrir chat de la persona, escribir “Hola, ¿te gustaría ir a desayunar conmigo el día de mañana en la cafetería?”, click en enviar, esperar, si la respuesta es igual a “si” entonces, escribir “genial, te veo mañana”, presionar botón “enviar” si no, escribir “oh, creo que será en otra ocasión”, presionar enviar.
* Desplazarse desde su casa a un centro comercial. -salir de casa, ubicar centro comercial más cercano, ubicar medio de transporte, opciones,” bici”,” caminar”, “carro”, conducir/caminar/pedalear para dirigirse a centro comercial

b) Si el lector es un estudiante universitario, también realice los siguientes ejercicios, los cuales le permitirán familiarizarse con los procesos de su institución educativa. Tenga en cuenta que, si no los conoce, deberá realizar las consultas necesarias:

* Solicitar la homologación de un espacio académico. -contactar secretario/a, solicitar información sobre homologación, validar requisitos, comparar con documentos de los cuales posee, si todos aparecen continuar, si no, revisar documentos faltantes, continuar con petición, realizar documento con solicitud, generar solicitud de homologación, firmar solicitud, entregar solicitud a secretario/a.
* Realizar un proceso de validación de un espacio académico. – consultar horario con secretario/a, generar solicitud de revisión ante disponibilidad, evaluar franja de espacios disponibles para la fecha,
* Cancelar materias y semestre. Consultar materias, verificar materias inscritas, revisar validez de materias, seleccionar materias por descartar, escribir carta formal solicitando quitar las materias seleccionadas de la lista, enviar documento a área encargada.
* Realizar un préstamo de un libro en la biblioteca. Validar número y código del libro, tomar el libro, dirigirse a recepción de biblioteca, validar requisitos para pedir prestado el libro, revisar si cumple requisitos, si no, revisar elementos para solicitarlo, continuar con proceso de préstamo, entregar credenciales para préstamos, tomar libro, retirarse, darle uso al libro
* Realizar una consulta bibliográfica en las bases de datos de la biblioteca. – tener nombre del libro/dato/información, validar página de la biblioteca, ingresar a base de datos de libros en la página, suministrar nombre del ejemplar, realizar validación de información
* Solicitar una cita en el Centro Médico de Bienestar Institucional – solicitar información de contacto en secretaria/bienestar, tomar medios de comunicación, radicar correo institucional al correo de contacto solicitando la cita, validar opción de agendamiento, realizar cita, esperar hora y fecha de asignación, asistir.

Capítulo 2

2.3. Ejercicios propuestos

Diseñe algoritmos utilizando tanto pseudocódigo como diagramas de flujo para los problemas que se enuncian a continuación. Una vez se hayan construido, elabore la prueba de escritorio con la tabla de verificación para determinar si el algoritmo se ejecuta adecuadamente y entrega los resultados esperados:

1. Suponga que, para el algoritmo del cálculo de la nota definitiva expuesto en el Ejemplo 2.4, el acuerdo al que llegaron profesor y 116 Estructura secuencial estudiantes consiste en darle un porcentaje a cada una de las 4 notas, de la siguiente manera:

Table

Description automatically generated

Tabla 2.4: Porcentaje de notas para el Ejercicio Propuesto 1

Reescriba el algoritmo, de tal forma que se pueda obtener la nota definitiva a partir de las notas y sus porcentajes. Compare los resultados obtenidos por este algoritmo con los que se obtienen en el algoritmo del Ejemplo 2.4.

////////////////////////////////////////////////////////////////////////////////////////////////////

#include <iostream>

using namespace std;

int main() {

float primera,segunda,tercera,cuarta,total;

float porcentaje\_primera,porcentaje\_segunda,porcentaje\_tercera,porcentaje\_cuarta;

double porcentaje\_total;

cout<<"---------------------------------------------------------------------------"<<endl;

cout<<"--------------- CALCULO DE NOTAS ------------------"<<endl;

cout<<"---------------------------------------------------------------------------"<<endl;

cout<<"ingrese notas de 1.0 a 5.0 con decimales: "<<endl;

cout<<"ingrese la primera nota: ";

cin>>primera;

cout<<"ingrese la segunda nota: ";

cin>>segunda;

cout<<"ingrese la tercera nota: ";

cin>>tercera;

cout<<"ingrese la cuarta nota: ";

cin>>cuarta;

cout<<"----------Tabla de equivalencias-------------"<<endl;

cout<<"---------------------------------------------"<<endl;

cout<<"-------nota parcial----------porcentaje------"<<endl;

cout<<" primera 15% "<<endl;

cout<<" segunda 15% "<<endl;

cout<<" tercera 30% "<<endl;

cout<<" cuarta 40% "<<endl;

cout<<"---------------------------------------------"<<endl;

cout<<"---------------------------------------------"<<endl;

porcentaje\_primera=primera\*0.15;

porcentaje\_segunda=segunda\*0.15;

porcentaje\_tercera=tercera\*0.30;

porcentaje\_cuarta=cuarta\*0.40;

porcentaje\_total=porcentaje\_primera+porcentaje\_segunda+porcentaje\_tercera+porcentaje\_cuarta;

total=primera\*segunda\*tercera\*cuarta/4;

cout<<"el resultado total de cada una de las notas es: "<<endl;

cout<<" "<<primera<<" "<<porcentaje\_primera<<" "<<endl;

cout<<" "<<segunda<<" "<<porcentaje\_segunda<<" "<<endl;

cout<<" "<<tercera<<" "<<porcentaje\_tercera<<" "<<endl;

cout<<" "<<cuarta<<" "<<porcentaje\_cuarta<<" "<<endl;

cout<<"el promedio de notas es: "<<total<<" su equivalente en porcentaje es: "<<porcentaje\_total<<" "<<endl;

return 0;

}

/////////////////////////////////////////////////////////////////////////////////////////////

2. Imagine que se conoce el valor del lado de un cubo. Elabore un algoritmo que permita calcular el ´área total de sus seis caras, el perímetro y el volumen del cubo. El ´área de una cara se obtiene por el producto de sus dos lados. El volumen se obtiene elevando al cubo su lado conocido y el perímetro es la suma de todos sus lados.

#include <iostream>

#include <math.h>

using namespace std;

int main()

{

int b,h,area\_total,area1,perimetro,volumen;

cout<<"introduzca la medida del lado: ";

cin>>h;

cout<<"introduzca la medida de la altura: ";

cin>>b;

area1= b \* h;

area\_total=(6 \* area1);

perimetro=(12 \* h);

volumen=(h \* h \* h);

cout<<"El area total de sus lados es: "<<area\_total<<" cm^2 "<<endl;

cout<<"su perimetro es: "<<perimetro<<" cm^2 "<<endl;

cout<<"su volumen es: "<<volumen<<" cm^2 "<<endl;

return 0;

}

3. Un empleado fue contratado durante un periodo específico en días y por un salario, ambos conocidos. Construya un algoritmo que permita calcular el valor de su liquidación al terminar el contrato. La liquidación se compone de prima, cesantías, intereses a las cesantías y vacaciones. Para calcular estos valores, se usan las siguientes formulas:

Prima: (salario \* diasLaborados) / 360

Cesant´ıas: (salario \* diasLaborados) / 360

Intereses cesant´ıas: cesant´ıas \* (12 % / diasLaborados)

Vacaciones: (salario \* diasLaborados) / 720

-------------------------------------------------------------------------------------------------------

#include<iostream>

using namespace std;

int main(){

float salario,diasLaborados;

double prima,cesantias,intereses\_cesantias,vacaciones,total;

cout<<"introduzca los dias que laboro: ";

cin>>diasLaborados;

cout<<"introduzca el salario dispuesto: ";

cin>>salario;

prima=(salario \* diasLaborados)/360;

cesantias=(salario\*diasLaborados)/360;

intereses\_cesantias=cesantias\*(cesantias\*0.12/(diasLaborados));

vacaciones=(salario\*diasLaborados)/720;

total=prima+cesantias+intereses\_cesantias+vacaciones+vacaciones;

cout<<"usted recibira: "<<endl;

cout<<"prima: " <<prima<< "."<<endl;

cout<<"cesantias: " <<cesantias<< "."<<endl;

cout<<"intereses de cesantias: " <<intereses\_cesantias<< "."<<endl;

cout<<"vacaciones: " <<vacaciones<< "."<<endl;

cout<<"para un total de: "<<total<< "." <<endl;

return 0;

}

4. Diseñe un algoritmo que, al ingresarle el costo de un producto y la cantidad de ese producto que compra un cliente, calcule y muestre el valor a pagar por el cliente, obteniendo igualmente, el valor del impuesto al valor agregado IVA, que corresponde al 16 % del valor de la compra y que también el cliente debe pagar.

#include<iostream>

using namespace std;

int main(){

int cliente,producto,desicion,SI,NO;

double precio,cantidad,total,total\_IVA,IVA;

cout<<"----------------------Caja---------------------"<<endl;

cout<<"-----------------------------------------------"<<endl;

cout<< "Digite su nombre: ";

cin>>cliente;

cout<<"-----------------------------------------------"<<endl;

cout<< "Digite el nombre del producto: ";

cin>>producto;

cout<<"-----------------------------------------------"<<endl;

cout<< "Digite el precio del producto: ";

cin>>precio;

cout<<"-----------------------------------------------"<<endl;

cout<< "Digite cuantos productos va a llevar: ";

cin>>cantidad;

cout<<"-----------------------------------------------"<<endl;

total=precio\*cantidad;

IVA=total\*0.19;

total\_IVA=total+IVA;

cout<<"-----------------------------------------------"<<endl;

cout<<"SR/SRA "<<cliente<< "usted debera pagar: "<<total\_IVA<<endl;

cout<<"-----------------------------------------------"<<endl;

cout<<" el subtotal al pagar es: "<<total<<endl;

cout<<"-----------------------------------------------"<<endl;

cout<<" con un cargo del IVA del 19% de: "<<IVA<<endl;

cout<<"-----------------------------------------------"<<endl;

cout<<"Desea probar con otro cliente? SI / NO ";

if desicion = SI{

cout<<"Permita un segundo mientras el programa se reinicia"<<endl;

else if desicion = NO{

cout<<"Gracias por usar nuestro producto"<<endl;

}

else if desicion !SI,NO{

cout<< "Su opcion no forma parte de nuestras opciones, intente de nuevo"<<endl;

}

}

return 0;

}

5. Construya un algoritmo al que se le ingresa el valor de un número determinado y el algoritmo realiza la suma de los números que le preceden desde

1. Para ello, utilice la siguiente formula:

Graphical user interface, application, Word

Description automatically generated

#include<iostream>

using namespace std;

int main(){

float a,N;

cout<<"ingrese un valor para 'n' "<<endl;

cin>>a;

N=a\*(a+1)/2;

cout<<"el valor de la operacion n\*n+1/2 es igual a: "<<N<<" "<<endl;

return 0;

6. Un agente inmobiliario necesita un algoritmo con el que pueda saber el precio al cual debe vender una finca o terreno; para ello, el algoritmo debe conocer el número de metros cuadrados que tiene el terreno, el valor del metro cuadrado y el valor de la comisión por la venta que equivale al 2 % del valor de los metros cuadrados de los que consta el terreno. Escriba un algoritmo que permita saber el valor en el que se debe vender una finca o terreno.

#include<iostream>

using namespace std;

int main(){

float metros\_cuadrados,comision,valor,valor\_venta,vf;

cout<<"ingrese la cantidad de metros cuadrados del terreno"<<endl;

cin>>metros\_cuadrados;

cout<<"ingrese el precio por metro cuadrado"<<endl;

cin>>valor;

cout<<"la comision que llevara es de: 2%"<<endl;

valor\_venta=metros\_cuadrados\*valor;

comision=valor\_venta\*0.02;

vf=valor\_venta+comision;

cout<<"el coste de comision sera de: "<<comision<<" . "<<endl;

cout<<"debera vender la finca o terreno a: "<<vf<<" . "<<endl;

return 0;

}

7. Construya un algoritmo que reciba como entrada un ´Angulo expresado en grados, minutos y segundos y se entregue como resultado la medida del ´Angulo en radianes.

#include<iostream>

using namespace std;

int main(){

float grados,Minutos,segundosi,segundos,num;

double decimal;

cout<<"introduzca los grados: "<<endl;

cin>>grados;

decimal = grados - int(grados);

Minutos = decimal \* 60;

segundosi = Minutos - int(Minutos);

segundos = segundosi \* 60;

cout << "Grados: "<< int(grados) << " Minutos: "<< Minutos << " segundos: "<< segundos <<endl;

return 0;

}

8. Un artesano se enfrenta diariamente al problema de determinar cuál será el precio de venta de un producto fabricado. Para asignar el precio, el artesano identifica el valor de la materia prima utilizada, el valor de la mano de obra y a estos les incrementa la utilidad que desea ganar, que es del 30 %. Diseñe un algoritmo al que se le ingresa el nombre de un producto, el valor de las materias primas usadas en su elaboración y el valor de la mano de obra con que se construyó y el algoritmo determina el precio de venta del producto.

#include<iostream>

using namespace std;

int main(){

long double precio\_final,precio\_final\_comision, comision, valor\_materia\_prima, valor\_mano\_de\_obra;

cout<<"introduzca el valor de la mano de obra: "<<endl;

cin>>valor\_mano\_de\_obra;

cout<<"introduzca el valor de la materia prima: "<<endl;

cin>>valor\_materia\_prima;

precio\_final=valor\_materia\_prima+valor\_mano\_de\_obra;

comision=precio\_final\*0.3;

precio\_final\_comision=precio\_final+comision;

cout << "el precio final del precio de venta es: "<<precio\_final\_comision<<endl;

cout << "el precio sin la ganancia esperada seria de: "<<precio\_final<<endl;

cout << "lo estimado en ganancias del 30% es de: "<< comision <<endl;

return 0;

}

9. Construir un algoritmo que permita determinar cuál es la altura de un edificio del que se deja caer una pelota que tarda 3 segundos en tocar el piso y cuál es la velocidad con la que la pelota llega al piso. Para escribir el algoritmo, utilice las siguientes formulas:

Text

Description automatically generated

#include<iostream>

#include<conio.h>

using namespace std;

int main(){

int altura,velocidadinicial,tiempo,gravedad,velocidadfinal;

cout<<"Introduzca cuanto tiempo dura en caer la pelota: "<<endl;

cin>>tiempo;

cout<<"Introduzca su velocidad inicial: "<<endl;

cin>>velocidadinicial;

gravedad=9.8;

altura=velocidadinicial\*tiempo+ 0.5 \* gravedad\*(tiempo^2);

velocidadfinal= (velocidadinicial^2)+ 2 \*gravedad\*altura;

cout<<"la altura desde la cual se lanzo el objeto fue desde: "<<altura<<" metros/segundos"<<endl;

cout<<"y la velocidad con la que la pelota llego al suelo fue: "<<velocidadfinal<<" metros/segundos"<<endl;

getch ();

return 0;

}

10. Dada una temperatura en grados cent´ıgrados, construya un algoritmo que convierta esa temperatura tanto a grados Kelvin como a grados Fahrenheit. Use las siguientes f´ormulas.ç

Text, letter

Description automatically generated

#include <stdio.h>

#include <stdlib.h>

int main(int argc, char\*\* argv) {

float grados\_C; /\* Temperatura en grado Celsius \*/

float grados\_F; /\* Temperatura en grado Farenheit \*/

float grados\_K; /\* Temperatura en Kelvin \*/

printf("Introduzca una temperatura en grados Celsius: ");

scanf("%f", &grados\_C);

grados\_F = (9.0/5.0)\* grados\_C + 32;

grados\_K = grados\_C + 273.15;

printf("%.2f grados Celsius = %.2f grados Fahrenheit = %.2f Kelvin\n", grados\_C, grados\_F,grados\_K);

system("PAUSE");

return (0);

}

Capítulo 3

3.4. Ejercicios propuestos

1. Crear un algoritmo que indique el valor del descuento de un artículo el cual es del 5% solo si el artículo tiene un costo superior al $150.000.

#include<iostream>

using namespace std;

int main() {

float descuento;

float precio;

float total;

cout << ("Escriba el precio del producto que quiere comprar: ") << endl;

cin >> precio;

if (precio>=150000) {

descuento = 150000\*15/100;

cout << ("El descuento del producto es: ") << endl;

total = precio-descuento;

} else {

cout << "Este producto no tiene descuento" << endl;

}

return 0;

}

|  |  |  |
| --- | --- | --- |
| Precio | Descuento | Total |
| 80000  170000 | 0  25500 | 80000  144500 |

1. Crear un algoritmo que indique si la llave de un tanque de agua debe ser abierta o cerrada. El tanque debe estar siempre entre 250 y 450 litros.

#include<iostream>

using namespace std;

int main() {

int abrir;

int agua;

int cerrar;

int ma;

int me;

cout << "¿Abrir o Cerrar?" << endl;

cout << ("¿Que cantidad de agua tiene el tanque?: ") << endl;

cin >> agua;

me = 250;

ma = 450;

if (agua<=250) {

cout << ("Debe abrir la llave") << endl;

}

if (agua>=450) {

cout << ("Debe cerrar la llave") << endl;

}

return 0;

}

**Punto 3 (Capitulo 3)**

#include<iostream>

using namespace std;

int main() {

float cosc;

float primo;

cout << "Inserte un numero del 0 al 20: " << endl;

cin >> primo;

cosc = primo%2;

if (primo>20 && primo<0) {

cout << ("El numero no es valido") << endl;

}

if (cosc==0) {

cout << "El numero es primo" << endl;

} else {

cout << "El numero no es primo" << endl;

}

return 0;

}

Punto 4 (Capitulo 3)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| N1 | N2 | N3 | N4 | N5 | Definitiva |
| 4,3  5 | 3,9  2,8 | 3,1  3,3 | 4,5  3,8 | 4  2,5 | 3,96  3,48 |

1. Crear un algoritmo que dado un número entero entre 0 y 20 diga si es o no un número primo. Recuerde que los números primos menores o iguales a 20 son: 2, 3, 5, 7, 11, 13, 17, 19.

#include<iostream>

using namespace std;

int main() {

float cosc;

float primo;

cout << "Inserte un numero del 0 al 20: " << endl;

cin >> primo;

cosc = primo%2;

if (primo>20 && primo<0) {

cout << ("El numero no es valido") << endl;

}

if (cosc==0) {

cout << "El numero es primo" << endl;

} else {

cout << "El numero no es primo" << endl;

}

return 0;

}

1. Crear un algoritmo que indique si un estudiante ganó o perdió un curso después de presentar los cinco trabajos asociados al curso (Notas entre 0.0 y 5.0). Los trabajos tienen igual peso sobre la nota final y se gana el curso si la nota definitiva es superior a 3.5.

#include<iostream>

using namespace std;

int main() {

float definitiva;

float n1;

float n2;

float n3;

float n4;

float n5;

cout << "Calificaciones" << endl;

cout << "Digite la primera nota: " << endl;

cin >> n1;

cout << "Digite la segunda nota: " << endl;

cin >> n2;

cout << "Digite la tercera nota: " << endl;

cin >> n3;

cout << "Digite la cuarta nota: " << endl;

cin >> n4;

cout << "Digite la quinta nota: " << endl;

cin >> n5;

definitiva = (n1+n2+n3+n4+n5)/5;

if (definitiva>=3.5) {

cout << "Felicidades, logro pasar con: " << definitiva << endl;

} else {

cout << "Desafortunadamente usted no logro pasar, su nota fue: " << definitiva << endl;

}

return 0;

}

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| N1 | N2 | N3 | N4 | N5 | Definitiva |
| 4,3  5 | 3,9  2,8 | 3,1  3,3 | 4,5  3,8 | 4  2,5 | 3,96  3,48 |

1. 5. Crear un algoritmo que permita saber si una ecuación cuadrática tiene o no solución. Recuerde que una ecuación cuadrática se define como:

A picture containing text

Description automatically generated

Y se dice que tiene solución si el valor del discriminaste (que corresponde al cálculo interno de la raíz cuadrada b 2 −4ac) es mayor o igual a cero y el valor de a es diferente de cero.

#include<iostream>

#include<cmath>

using namespace std;

int main() {

int a;

int b;

int c;

int division1;

int division2;

int rais;

cout << "Digite el valor para a: " << endl;

cin >> a;

cout << "Digite el valor para b: " << endl;

cin >> b;

cout << "Digite el valor para c: " << endl;

cin >> c;

rais = sqrtf(b\*2-4\*a\*c);

division1 = -b+rais/(2\*a);

division2 = -b-rais/(2\*a);

if (rais>=0 && a!=0) {

cout << "Esta ecuación si tiene solución y es: " << division1 << " - " << division2 << endl;

} else {

cout << "Esta ecuación no tiene solución." << endl;

}

return 0;

}

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | C | Raiz | Division1 | Division2 |
| 2  3 | 9  8 | 1  4 | 8,54  4 | -0,11  -0,66 | -4,385  -2 |

1. Crear un algoritmo que indique si un numero entero x, ingresado por el usuario, se encuentra por dentro o por fuera del intervalo cerradocerrado [minimoV alor, maximoV alor)] también ingresados por el usuario.

Por ejemplo: si los valores mínimos y máximo son 3 y 7 respectivamente, el valor 5 está dentro, mientras que el valor de 8 está por fuera del intervalo.

#include<iostream>

using namespace std;

int main() {

int limit1;

int limit2;

int nm;

cout << ("Agregue el limite inferior del intervalo: ") << endl;

cin >> limit1;

cout << "Agregue el limite superior del intervalo: " << endl;

cin >> limit2;

cout << "Digite el numero que quiere conocer: " << endl;

cin >> nm;

if (nm>=limit1 && nm<=limit2) {

cout << ("El numero si esta dentro del intervalo") << endl;

} else {

cout << ("El numero no se encuentra dentro del intervalo") << endl;

}

return 0;

}

1. Crear un algoritmo que indique si un número entero x se encuentra por dentro o por fuera de tres intervalos abierto-abierto cuyo rangos no se interceptan entre sí y sus límites son ingresados por el usuario.

#include<iostream>

using namespace std;

int main() {

int limit1;

int limit2;

int limit3;

int limit4;

int limit5;

int limit6;

int x;

cout << "Escriba el numero que va a determinar: " << endl;

cin >> x;

cout << "Escriba el limite inferior de la primera variable: " << endl;

cin >> limit1;

cout << "Escriba el limite superior de la primera variable: " << endl;

cin >> limit2;

cout << "Escriba el limite inferior de la segunda variable: " << endl;

cin >> limit3;

cout << "Escriba el limite superior de la segunda variable: " << endl;

cin >> limit4;

cout << "Escriba el limite inferior de la tercera variable: " << endl;

cin >> limit5;

cout << "Escriba el limite superior de la tercera variable: " << endl;

cin >> limit6;

if (x>limit1 && x<limit2) {

cout << "El valor se encuentra dentro de la primera variable" << endl;

}

if (x>limit3 && x<limit4) {

cout << "El valor se encuentra dentro de la segunda variable" << endl;

}

if (x>limit5 && x<limit6) {

cout << ("El valor se encuentra dentro de la tercera variable") << endl;

} else {

cout << ("Este numero no se encuentra en ninguna variable") << endl;

}

return 0;

}

1. Crear un algoritmo que indique el valor del descuento de un art´ıculo dependiendo de su tipo:

#include<iostream>

using namespace std;

int main() {

float descuento;

float precio;

float tipo;

float total;

cout << ("Elija el tipo de producto (Textil = 0), (Electrodoméstico = 1), (Elementos de cocina = 2) y (Videojuego = 3 : ") << endl;

cin >> tipo;

cout << ("Digite el precio del producto: ") << endl;

cin >> precio;

if (tipo==0) {

descuento = precio\*0/100;

total = precio-descuento;

cout << "El descuento del producto es: " << descuento << endl;

cout << "El precio final es: " << total << endl;

}

if (tipo==1) {

descuento = precio\*3.7/100;

total = precio-descuento;

cout << "El descuento del producto es: " << descuento << endl;

cout << "El precio final es: " << total << endl;

}

if (tipo==2) {

descuento = precio\*4.2/100;

total = precio-descuento;

cout << "El descuento del producto es: " << descuento << endl;

cout << "El precio final es: " << total << endl;

}

if (tipo==3) {

descuento = precio\*7.8/100;

total = precio-descuento;

cout << "El descuento del producto es: " << descuento << endl;

cout << "El precio final es: " << total << endl;

}

return 0;

}

Table

Description automatically generated

1. Crear un algoritmo que indique el valor del descuento de un artículo dependiendo de su valor:

#include<iostream>

using namespace std;

int main() {

float descuento;

float precio;

float total;

cout << ("Escriba el precio del producto: ") << endl;

cin >> precio;

if (precio>=0 && precio<=100000) {

descuento = precio\*0/100;

total = precio-descuento;

cout << "El descuento para el producto es del 0% que equivale a: " << descuento << endl;

cout << "El precio final es: " << total << endl;

}

if (precio>100000 && precio<=225000) {

descuento = precio\*1.5/100;

total = precio-descuento;

cout << "El descuento para el producto es del 1,5% que equivale a: " << descuento << endl;

cout << "El final es: " << total << endl;

}

if (precio>225000 && precio<=375000) {

descuento = precio\*3.8/100;

total = precio-descuento;

cout << "El descuento para el producto es del 3,8% que equivale a: " << descuento << endl;

}

if (precio>375000) {

descuento = precio\*10.3/100;

total = precio-descuento;

cout << "El descuento para el producto es del 10,3% que equivale a: " << descuento << endl;

cout << "El final es: " << total << endl;

}

return 0;

}

Table

Description automatically generated

Capítulo 4

1. Responda las siguientes preguntas:
2. ¿Cuáles son las estructuras repetitivas condicionadas al comienzo?

* **Mientras – FinMientras**
* **Haga – MientrasQue**
* **Para – Finpara**

b) ¿De las estructuras repetitivas estudiadas en este capítulo, ¿cuál o cuáles de ellas pueden llegar a no ejecutarse y por qué?

* Mientras – FinMientras no puede llegar a ejecutase si no se inicializan bien las variables que serán utilizadas incluyendo los contadores o los acumuladores (el resultado de la condición debe evaluarse al inicio del ciclo).
* En las estructuras anidadas no se ejecuta ni se pasa el control a la estructura externa hasta que la estructura interna no se termine de ejcutar.
* La estructura Para-FinPara no se puede llegar a ejecutar si al evaluar por primera vez la condición esta resulte falsa.
* Algunos casos que se solucionan con Mientras-FinMientras o Haga-Mientrasque no pueden ser solucionados con Para-FinPara.

c) De las estructuras repetitivas estudiadas en este capítulo, ¿cuál o cuáles de ellas se ejecutan por lo menos una vez y por qué?

Todas las estructuras estudiadas en este capitulo se pueden llegar a ejecutar, cada una tienen que cumplir ciertas condiciones para que puedan ser ejecutadas.

* Mientras-FinMientras solo se puede llegar a ejecutar si se determinan bien desde un principio las variables que se van a usar.
* Las anidadas se ejecutan completamente cuando se ejecuta correctamente primero la estructura interna y luego la externa.
* Haga-Mientrasque solo se ejecutara cuando se cumpla la condición que se le impuso con anterioridad.
* Para-FinPara se ejecuta cuando se le da el numero de limitantes y la cantidad con la que se va a disminuir o aumentar.

d) ¿A qué se le llama iteración?

Se le denomina iteración a cada ejecución de un ciclo del programa.

e) Escriba dos situaciones en las que usaría una variable bandera.

La primera situación en la que usaría esta variable seria en una calculadora para que el usuario determine si quiere continuar realizando operaciones o si quiere terminar el proceso, debido a que esta variable controla el sistema de iteración. La segunda situación en la que usaría la variable bandera es la programación de un cajero para que el usuario determine si quiere seguir retirando dinero o finalizar la operación.

f) ¿Cuál es la diferencia entre un acumulador y un contador?

Un acumulador es una variable que se utiliza para sumar valores. Al igual que el contador, se utiliza normalmente dentro de un ciclo, pero cambiamos su valor sumándole una variable, es decir, no siempre se le suma la misma cantidad. En el siguiente programa, utilizamos el mismo arreglo del programa contador.

g) ¿Qué sucedería si Usted no escribe la instrucción modificadora de condición dentro de un ciclo Mientras-FinMientras?

El archivo no detectaría la función ni el fin de la operación usando el condicional declarado con anterioridad, en resumen, no se ejecuta.

h) ¿Por qué dentro del cuerpo del ciclo Para, no está explicita la instrucción modificadora de condición?

Porque si se planteara de esta manera, el ciclo aplicaría la instrucción directamente a su proceso y modificaría el valor dentro de su mismo ciclo, no en torno a todo el programa, por ende, es mejor declarar todas las variables al inicio de un programa determinado para evitar algún conflicto en el mismo.

1. ¿Para validar la entrada de un dato (lectura) en un algoritmo, cual es el ciclo ideal? Justifique su respuesta.

Si es una lectura de datos sin un límite especifico al que se quiera llegar, el ciclo más adecuado para esto es el Mientras-FinMientras, debido a que este se rige principalmente por los números booleanos por lo cual no se terminara hasta que el programa se encuentre con una variable que considere falsa, el usuario podrá ingresar los datos que desee y el mismo podrá decidir cuándo terminar este ciclo sin necesidad de cumplir con un numero especifico de repeticiones.

j) Dentro de la cultura popular siempre se ha dicho que, si alguien no puede dormir, debe contar ovejas hasta que logre conciliar el sueño. Suponga que le piden a Usted, que mediante un diagrama de flujo represente esta situación. De las 3 estructuras repetitivas estudiadas en este capítulo, ¿cuál es la menos indicada para hacerlo? Justifique su respuesta.

El algoritmo menos indicado para esta tarea en mi opinión seria el Para-FinPara debido a que este se le debe poner unos limitantes específicos, de la cantidad de veces que se va a repetir, por lo cual si se le coloca un número muy grande este seguirá contando ovejas aunque el usuario ya se “durmiera” y si le ponemos un número muy pequeño lo más probable es que se finalice antes de que el usuario se “duerma”, es muy difícil determinar el número exacto en el que va a resultar, por esta razón es mejor usar los otros dos ciclos que no necesitan de un número determinado, sino que necesitan de una condición para finalizar su funcionamiento.

2. Tome cada uno de los ejemplos de este capítulo y reescríbalos usando una estructura de ciclo diferente. En caso de no ser posible el cambio, justifique su respuesta.

* Instrucción de inicialización

Haga {

Instrucción-1

Instrucción-2

/\* Cuerpo del ciclo \*/

Instrucción – n

Instrucción modificadora de condición

MientrasQue (condición)

Instrucción externa

* Numero = 1

Para ( numero == 1; numero <= 10; numero++) {

Imprimir (numero)

}

FinPara

* Numero = 0

Para (numero == 0; numero <= 10; numero++){

Imprimir (numero)

}

FinPara

* Algoritmo Serie

/\* Este algoritmo imprime la siguiente serie, de acuerdo al número de términos que se le especifique: 1, 3, 5, 7, 9, 11, …, n

\*/

// Declaración de variables

Entero cantidadTerminos, contadorNumeros, termino

//Dato disponible

Imprimir (“Ingrese la cantidad de términos a generar: “)

Leer (cantidadTerminos)

Haga {

contadorNumeros = contadorNumeros + 1

termino = termino + 2

Imprimir (termino, “, ”)

}

Mientrasque (contadorNumeros < cantidadTerminos – 1)

Imprimir (termino)

FinAlgoritmo

* Algoritmo Estudiantes

Real notadefinitiva, sumaDefinitivas, promedioGrupo

Entero cantidadEstudiantes, contadorEstudiantes

Entero aprobaron, reprobaron

Cadena codigoEstudiante

Imprimir (“Ingrese la cantidad de estudiantes: “)

Leer ( cantidadEstudiantes )

ContadorEstudiantes = 0

Aprobaron = 0

Reprobaron = 0

sumaDefinitivas = 0

Haga {

Imprimir (“ Ingrese el código del estudiante: “)

Leer ( codigoEstudiante )

Imprimir (“Ingrese la nota definitiva: “)

Leer (notaDefinitiva)

Si ( notaDefinitiva >= 3.0 ) Entonces

Aprobaron = aprobaron + 1

SiNo

Reprobaron = reprobaron + 1

FinSi

sumaDefinitivas = sumaDefinitivas + notaDefinitiva

contadorEstudiantes = contadorEstudiantes + 1

MientrasQue (contadorEstudiantes < cantidadEstudiantes)

Imprimir (“La cantidad que aprobaron es: “, aprobaron)

Imprimir (“La cantidad que reprobaron es: “, reprobaron)

Imprimir (“El promedio es: “, promedioGrupo)

FinAlgoritmo

* Algoritmo Cifras Numero

Entero numero, cifra, contadorCifras, sumaCifras, copiaNumero

Imprimir (“Escribir un numero entero: “)

Leer (numero)

Si (numero > 0) Entonces

copiaNumero = numero

contadorCifras = 0

SumaCifras = 0

Haga {

Cifra = copiaNumero % 10

copiaNumero = copiaNumero / 10

sumaCifras = sumaCifras + cifra

contadorCifras = contador Cifras + 1

Mientras que ( copiaNumero > 0)

Imprimir (“La cantidad de cifras de: “, numero)

Imprimir (“son: “, contadorCifras)

Imprimir (“La sumatoria es: “, sumaCifras)

SiNo

Imprimir (“No es un numero positivo”)

FinSi

FinAlgoritmo

* Algoritmo Armstrong

Entero numero, copiaNumero, contadorCifras, cifra, sumaCifras

Imprimir (“Escriba el número a analizar: “)

Leer (numero)

contadorCifras = 0

copiaNumero = numero

Hacer {

Cifra = copiaNumero % 10

copiaNumero = copiaNumero / 10

contadorCifras = contadorCifras + 1

MientrasQue (copiaNumero > 0)

sumaCifras = 0

copiaNumero = numero

Hacer {

Cifra = copiaNumero % 10

copiaNumero = copiaNumero / 10

sumaCifras = sumaCifras + cifra ^ contadorCifras

MientrasQue (copiaNumero > 0)

Si (numero == sumaCifras) Entonces

Imprimir (numero, “es un número de Armstrong.”)

SinNo

Imprimir (numero, “No es un número de Armstrong.”)

FinSi

FinAlgoritmo

* Algoritmo Euclides

Entero a, b, dividiendo, divisor, resto, mcm

Caracter seguir

Seguir = ‘ S ’

Haga {

Imprimir (“Ingrese el primer número: “)

Leer (a)

Imprimir (“Ingrese el segundo numero: “)

Leer (b)

Si (a > b) Entonces

Dividendo = a

Divisor = b

Sino

Dividendo = b

Divisor = a

FinSi

MientrasQue (seguir == ‘ S ‘ O seguir == ‘ s ‘)

Hacer

Dividendo = divisor

Divisor = resto

Resto = dividendo % divisor

MientrasQue (resto ¡= 0)

Mcm = (a \* b) / divisor

Imprimir (“El máximo común divisor de : “, a, “ y “, b)

Imprimir (“es: “, divisor)

Imprimir (“El mínimo común múltiplo es: “, mcm)

Imprimir (“Desea Realizar nuevos cálculos [S] o [N]?: “)

Leer (seguir)

FinAlgoritmo

* Algoritmo Multiplicacion

Entero multiplicando, multiplicador, producto, contador

Imprimir (“Ingrese el multiplicando: “)

Leer (multiplicando)

Imprimir (“Ingrese el multiplicador: “)

Leer (multiplicador)

Producto = 0

Contador = 0

Hacer {

Producto = producto + multiplicando

Contador = contador + 1

MientrasQue (contado < multiplicando)

Imprimir (“El producto es: “, producto)

FinAlgoritmo

* Algoritmo Serie

Entero contodorNumeros, cantidadTerminos, termino

Imprimir “Ingrese la cantidad de términos a generar: “)

Leer (cantidadTerminos)

contadorNumeros = 0

termino = 1

Mientras (contadorNumeros < cantidadTerminos – 1)

Imprimir (termino, “, ”)

Termino = termino + 1

contadorNumeros = contadorNumeros + 1

FinMientras

Imprimir (termino)

FinAlgoritmo

3. Usando la instrucción Para, represente los siguientes enunciados:

a) La variable x con un valor inicial de 4, un valor final de 40 e incrementos de a 1.

Entero x

Para (x == 4; x == 40; x++)

Imprimir (“La variable es: “ x)

FinPara

FinAlgoritmo

1. La variable x que va desde 100 a 20, disminuyendo de 1 en 1.

Entero x

Para (x == 100; y ==20; x--)

Imprimir (“La variable es: “ x)

FinPara

FinAlgoritmo

1. La variable x que inicia en 10, incrementando de 5 en 5, hasta llegar a 200.

Entero x

Para (x == 10; x == 200; x + 5)

Imprimir (“La variable es:” x)

FinPara

FinAlgoritmo

4. Dado los extremos de un intervalo [M, N], halle la sumatoria de los números pares y de los impares que pertenezcan a ´el.

Algoritmo

Entero M, N, i, pares, impares

Imprimir (“Escriba el primer limite: “)

Leer (M)

Imprimir (“Escriba el segundo limite: “)

Leer (N)

Para (i == M; i == N; i++) {

Pares = i % 2 == 0

Imprimir (“Los pares son: “ pares)  
impares = i % 2 ¡= 0

Imprimir (“Los impares son: “ impares)

FinPara

FinAlgoritmo

5. Dada una población, máximo de 500 habitantes, determinar cuántos son mayores y cuántos menores de edad. De cada uno de ellos se conoce la edad en años.

Algoritmo

Entero edad, mayor, menor, i, adulto, niño

Para( i = 0; i = 500; i++)

Imprimir (“Coloque la edad del habitante: “)

Leer (edad)

Mayor = edad >= 18

Menor = edad < 18

SI ( edad == mayor) {

Adulto = adulto + 1

}

SiNo {

Niño = niño + 1

}

FinPara

FinMientras

Imprimir (“La cantidad de mayores es: “ adulto)

Imprimir (“La cantidad de menores es: “ niño)

FinAlgoritmo

6. Genere e imprima los múltiplos de 3 que se encuentren entre 6 y n, donde n tiene que ser superior a 6.

Algoritmo

Entero multiplos, i, n

Imprimir (“Múltiplos del numero 3”)

Imprimir (“Ingrese el numero al que quiere llegar: ” n)

Para (i = 2; i = n; i++)

Multiplos = i \* 3

Imprimir (“Los multiplos son: “ multiplos)

FinPara

FinAlgoritmo

7. En 1937, el matemático alemán Lothar Collatz, enunció la conjetura de Collatz, también conocida como el problema de Ulam, conjetura 3n + 1, entre otros.

Collatz enunció que, a partir de cualquier número natural, siempre se obtiene la unidad. Para ello se hace el siguiente procedimiento:

Tome un número n y ejecute las siguientes operaciones: Si n es par, halle la división entera entre 2. Si n es impar, multiplíquelo por 3 y súmele 1.

Con el resultado que obtenga, repita las operaciones anteriores, hasta obtener 1 como respuesta. Ejemplos:

n = 13, se obtienen los siguientes resultados:

40, 20, 10, 5, 16, 8, 4, 2, 1.

n = 6, se obtienen los siguientes resultados:

3, 10, 5, 16, 8, 4, 2, 1.

Otra de las curiosidades de esta conjetura, es que cuando se llegue a 1 y se apliquen nuevamente las fórmulas, obtendrá la secuencia 4, 2, 1 de forma infinita.

El algoritmo que Usted diseñe debe solicitar un número y aplicar el anterior concepto, imprimiendo los resultados que se obtienen hasta llegar a la unidad.

Algoritmo

Entero pares, impares, n, n2

Imprimir (“Ingrese el numero: “)

Leer (n)

Pares = n % 2 == 0

Impares n % 2 ¡= 0

Si (n == pares)

Mientras (n ¡= 1)

N = n / 2

Imprimir ( n“, ”)

FinMientras

FinSi

Si (n == impar)

Mientras ( n!= 1)

N = ( n \* 3 ) + 1

Imprimir ( n“, ”)

FinMientras

FinSi

FinAlgoritmo

8. Lea un número entero positivo, descompóngalo en cada una de sus cifras y con ellas genere el número invertido. Por ejemplo, si el número a leer es el 5432, el resultado será 2345.

Algoritmo

Entero n, inverso, resultado

Imprimir (“Ingrese el numero que quiere invertir: “)

Leer (n)

Mientras (n > 0)

Resultado = resultado\*10

Resultado = resultado + n% 10

N = n / 10

FinMientras

Retornar a inverso (n)

Imprimir (“El numero invertido es: “ inverso (n))

FinAlgoritmo

1. Dado un número menor o igual a 50, calcule su factorial mediante sumas sucesivas.

Algoritmo

Entero n, i, factorial

Para (i =1; i = n; i ++)

Factorial = 1 + 1

FinPara

Imprimir (“El fatorial de este numero es: “ factorial)

FinAlgoritmo

10. El cajero de un restaurante desea controlar el flujo de caja en un día de trabajo cualquiera. Antes de abrir al público, el gerente del establecimiento le entrega la base para el día, la cual consiste en una suma de dinero que debe registrar en la caja y con la cual se espera pueda desempeñarse sin contratiempos. Durante su jornada tendrá ingresos por concepto de las ventas que se realicen, también habrán salidas de caja para la compra de insumos o gastos eventuales que deban realizarse.

Se espera un algoritmo, que reciba el registro de cada una de las operaciones a medida que vayan sucediendo. El cajero también requiere un informe del saldo en caja después de cada registro. El algoritmo deberá dar un mensaje de alerta en el caso que el saldo sea inferior o igual al 15 % de la base asignada. Al cierre del restaurante, se requiere los saldos finales (saldo en la caja, ingresos y egresos) y la cantidad de cada una de las operaciones realizadas.

**Algoritmo** Caja

Definir dinero, ganancia, gastos, descuento, aumento, porcentaje, final como real

Definir opcion, salir como real

Imprimir (“Ingrese el saldo con el que inicia”)

Leer (dinero)

Imprimir (“Elija la si son gastos o ganancia (gastos = 1) (ganancia = 2): ”)

Leer (opcion)

Imprimir (“Si quiere finalizar oprima (0): “)

Leer (x)

Hacer {

Segun opcion hacer

Caso 1: Imprimir (“Ingrese la cantidad de gastos: “)

Leer (gastos)

Descuento = dinero – gastos

Imprimir (“EL saldo actual es: “ descuento)

Caso 2: Imprimir (“Ingrese la cantidad de ganancia: “)

Leer (ganancia)

Aumento = dinero + ganancia

Imprimir (“El saldo actual es: “ aumento”)

}

MientrasQue (x == 0)

Si (descuento >= (dinero 15%)

Escribir (“¡CUIDADO! Su saldo es igual o menor que el 15%”)

FinSI

Final = aumento – descuento

Imprimir (“Los gastos son: “ gastos)

Imprimir (“Las ganancias son: ” ganancias)

Imprimir (“El saldo final es: “ final)

FinAlgoritmo

1. En el Ejemplo 4.15 se hizo un algoritmo que informaba si un número es o no perfecto. Basado en esa solución, diseñe un nuevo algoritmo que lea un número n e imprima los números perfectos entre 1 y n.

Algoritmo FactorialNumero

Entero n, factorial, inferiores, limit, i

imprimir("Ingrese el numero para el factorial: " )

leer(n)

imprimir (“Ingrese el limite al que quiere llegar: ”)

Leer (limit)

Si(n < 0)Entonces

imprimir( "No se puede calcular el factorial" )

SiNo

factorial = 1

inferiores = 1

Por (i = 1; i = limit; i++)

factorial = factorial\*inferiores

inferiores = inferiores + 1

imprimir("Los números perfectos son: ”, factorial)

FinSi

FinPara

FinAlgoritmo

1. Simule el funcionamiento de un temporizador, que reciba como entrada una cantidad de minutos y segundos. En el momento que falten 5 minutos para cumplir el tiempo, deber´a dar un mensaje de alerta, cuando finalice mostrara el siguiente mensaje “Tiempo fuera”. Debe funcionar Maximo para 1 hora.

Algoritmo

Entero i, segundos, minutos, conversion, total, fin

Imprimir (“Ingrese los minutos que desee (máximo 60): ”)

Leer (minutos)

Imprimir (“Ingrese los segundos que quiera medir: ”)

Leer (segundos)

Si (minutos > 60) {

Imprimir (“Este numero no es valido”)

Conversion = minutos \* 60

Total = conversion + segundos

Para (i = total; i = 0; i - 1)

Si (total <= 300)

Imprimir (“Solo faltan 5 minutos para que acabe”)

FinSi

FinPara

Si (total <= 0)

Imprimir (“TIEMPO FUERA”)

FinSi

FinAlgoritmo

1. Usando estructuras repetitivas, elabore un algoritmo, que genere e imprima las letras del abecedario de la siguiente forma:

Shape, arrow

Description automatically generated

Algoritmo Piramide

Carácter borde, letra

Para borde = ´ A ‘ Hasta ‘ Z ‘ incremento 1

Para letra = ‘ A ‘ hasta borde incremento 1

Imprimir (“ “, letra)

FinPara

FinPara

FinAlgoritmo

1. A un amigo que vive en el norte de la ciudad, su terapeuta le recomendó caminar mínimo 3 d´ıas a la semana desde su apartamento hasta el centro, lo cual ´el hace sin falta alguna; el recorrido tiene aproximadamente 22 cuadras y debe hacerlo con ropa cómoda. Este ejercicio lo debe realizar durante 4 meses. Al momento de volver a consulta, el amigo debe informarle a su terapeuta lo siguiente:

Promedio de tiempo por semana, por mes y por los 4 meses. Adicionalmente, cuál fue el menor y el mayor tiempo empleado en el recorrido. Para esta tarea, el amigo lleva un registro del tiempo que invierte en cada caminata.

Algoritmo

Definir i, semana, ,m, m2, promedio

Imprimir (“Lleve el registro del tiempo”)

Imprimir (“Tiempo por semana: “)

Leer (semana)

Para (i = 0; i = 4; i++)

Imprimir (Escriba el tiempo por mes: )

Leer (m)

M2 = m + m

FinPara

Promedio = m2 / 4

Imprimir (“El promedio de estos 4 meses es: “ promedio)

FinAlgoritmo

1. Imprima los 10 múltiplos sucesivos de 3 en orden descendente, a partir de un número n que será ingresado por el usuario y que representara el menor valor. Si el n, no es múltiplo de 3, debe llevarse al múltiplo más cercano superior.

Por ejemplo, si el número ingresado es el 28, al no ser múltiplo de 3 debe llevarse al siguiente múltiplo superior, o sea, 30. Entonces la salida del algoritmo se visualizaría así:

57, 54, 51, 48, 45, 42, 39, 36, 33, 30.

Algoritmo

Declarar n, multi, x, i enteros

Imprimir (“Escriba el numero con el que quiere finalizar”)

Leer (n)

X = n / 3

Hacer

i = x + 1

multi = i \* 3

MientrasQue (i >= (x+10))

Imprimir decremento (multi, ”, ”)

FinAlgoritmo

1. En una compañía que tiene varias sucursales a nivel nacional, una o varias en cada departamento y solo una por ciudad, se requiere de un censo que permita conocer la siguiente información de sus empleados:

Porcentaje total de personas que tienen estudios de primaria, secundaria, profesional, maestría o doctorado. Se tiene en cuenta solamente, el nivel más alto de estudio.

Porcentaje total de mujeres con posgrado, con relación a todas las mujeres de la compañía.

Cantidad de hombres con estudios de solo primaria en cada departamento.

Cantidad de hombres y mujeres en cada sucursal que hayan recibido su título profesional antes de cumplir 25 años.

De cada empleado se conoce el número de identificación y el nombre, adicional a los datos que se requieren en la solución del problema.

Algortimo

Declarar edad, sexo, nivel, p, s, pr, m, d, f, porcentajen, total, porcentajef, porcentajem, totaled, porcentajeed como Reales

Imprimir (“Porcentaje de nivel educativo”)

Imprimir (“Ingrese su nivel educativo máximo (primaria = 1), (secundaria = 2), (profesional = 3), (maestría = 4) o (doctorado = 5)”

Leer (nivel)

Según nivel hacer

Caso 1: p = p + 1

Caso 2: s = s + 1

Caso 3: pr = pr + 1

Caso 4: m = m + 1

Caso 5: d = d + 1

Total = caso 1 + caso 2 + caso 3 + caso 4 + caso 5

Porcentaje = (caso 1 / total) \* 100

Imprimir (“El promedio de personas que estudiaron hasta la primaria es: “ promedio)

Porcentaje = (caso 2 / total) \* 100

Imprimir (“El promedio de personas que estudiaron hasta la secundaria es: “ promedio)

Porcentaje = (caso 3 / total) \* 100

Imprimir (“El promedio de personas que estudiaron hasta el profesional es: “ promedio)

Porcentaje = (caso 4/ total) \* 100

Imprimir (“El promedio de personas que estudiaron hasta la maestría es: “ promedio)

Porcentaje = (caso 5/ total) \* 100

Imprimir (“El promedio de personas que estudiaron hasta el doctorado es: “ promedio)

Imprimir (“Digite su sexo (f) o (m): “)

Leer (sexo)

Si (sexo = f && nivel == 4 && nivel == 5)

M = m + 1

D = d + 1

Porcentajef = ((m + d) / total) \* 100

Imprimir (“El porcentaje con mujeres con una educación superior a la secundaria es: ”, porcentaje f)

FinSi

Si (sexo == m && nivel == 1)

P = p + 1

Porcentajem = (p / total) \* 100

Imprimir (“El porcentaje de hombres con una educación máxima igual a la primaria es: “, porcentajem)

FinSi

Imprimir (“Ingrese su edad: “)

Leer (edad)

SI (edad < 25 && nivel >= 3 )

Pr = pr + 1

M = m + 1

D = d + 1

Totaled = pr + m + d

Porcentajeed = (totaled / total) \* 100

Imprimir (“El porcentaje de personas que obtuvieron su profesional es: “, porcentajeed)

FinSi

FinAlgoritmo

1. Para los siguientes algoritmos, realice la respectiva prueba de escritorio o tabla de verificación:ç

Graphical user interface, text, application

Description automatically generated

1. El algoritmo fue probado y no funciona debido a que el ciclo Hasta y el ciclo Para se encuentran dentro de la misma variable unidos, generando que el programa no los pueda leer debido a que las sintaxis de ambos está mal y solo se puede usar uno a la vez
2. El algoritmo ya se probó y funciona a la perfección debido a que la sintaxis de los ciclos esta correcta y aunque se usaron varios ciclos, estos se usaron en diferentes variables, permitiendo que todos se ejecuten.
3. Este último algoritmo fue probado y funciono correctamente, en este también se usaron diferentes ciclos, sin embargo, funciona debido a que están en diferentes líneas para que no intervengan con los demás.