PSEUDOCODIGOS:

EJERCICIOS RESUELTOS:

EJERCICIO 01:

Registro punto

Real x

Real y

Fin\_registro

Algoritmo estructura\_01

Punto p1,p2

Real m,b

Ingresopunto (p1,”Primer punto ”)

Ingresopunto (p2,”segundo punto”)

Escribir “la distancia entre los puntos es: ”,distancia (p1,p2)

Calculo (p1,p2,m,b)

Reporte (m,b)

Fin\_algoritmo

Ingresopunto (punto p(R),const cadena mensaje [])

Escribir ,mensaje,

Escribir “valor de x: ”

Leer p.x

Escribir “valor de y:”

Leer p.y

Fin\_ingresopunto

Real distancia (punto p1,punto p2)

Retorno raíz de (pow (p2.x-p1.x,2)+pow(p2.y-p1.y,2))

Fin\_distancia

Calculo (punto p1,punto p2,ral m(R),real b(R))

M🡨(p2.y-p1.y)/(p2.x-p1.x)

B🡨p1.y-m\*p1.x

Fin\_calculo

Reporte (real m, real b)

Escribir “la ecuación de la recta es: ”

Escribir ,m,”x”

Si (b>0) entonces

Escribir “+”,b

Sino

Escribir ,b

Fin\_reporte

EJERCICIO 02:

Registro alumno

Cadena nombre [40]

Real nota

Fin\_registro

Algoritmo matriz\_02

Alumno a[100]

Entero n

Numalumno (n)

Ingreso (a,n)

Ordenalfabetico (a,n)

Escribir “lista en orden alfabetico ”

Reporte (a,n)

Ordenmarito (a,n)

Escribir “lista en orden de merito”

Reporte (a,n)

Fin\_algoritmo

Numalumno (entero n(R))

Hacer

Escribir “numero de alumnos: ”

Leer n

Mientras n<=0

Fin\_numalumno

Ingreso (alumno a[],entero n)

Entero i

Para i🡨0 hasta n-1 inc 1 hcaer

Escribir “dato del alumno ”,i+1

Escribir “nombre: ”

Leer (a[i].nombre,40)

Hacer

Escribir “nota :”

Leer a[i].nota

Mientras (a[i].nota<0 o a[i].nota>20)

Fin\_ingreso

Reporte (alumno a[],int n)

Entero i

Para i🡨0 hasta n-1 inc 1 hacer

Escribir a[i].nombre"\t ",a[i].nota

Fin\_para

Fin\_reporte

Ordenalfabetico (alumno a[],int n)

Entero i,j

Alumno temp

Para i🡨0 hasta n-1 inc 1 hacer

Para j🡨i+1 hasta n-1 inc 1 hacer

Si (a[i].nombre>a[j].nombre) entonces

Temp🡨a[i]

A[i]🡨a[j]

A[j]🡨temp

Fin\_si

Fin\_para

Fin\_para

Fin\_ordenalfabetico

Ordenmerito (alumno a[],entero n)

Para i🡨0 hasta n-1 inc 1 hacer

Para j🡨i+1 hasta n-1 inc 1 hacer

Si (a[i].nota>a[j].nota) entonces

Temp🡨a[i]

A[i]🡨a[j]

A[j]🡨temp

Fin\_si

Fin\_para

Fin\_para

Fin\_ordenmerito

EJERCICIO 03:

Registro dato

Real valor

Real peso

Fin \_ Registro

Algoritmo estructura\_03

Dato v [100]

Entero n

Numdato (n)

Ingreso (v, n)

Reportedato (v. n)

Escribir “El promedio ponderado es: “, promedioponderado (v, n)

Fin \_ Algoritmo

Numdato (entero n (R))

Hacer

Escribir “Numero de datos: “

Leer n

Mientras (n<=0)

Fin \_ numdato

Ingreso (dato v [], entero n)

Entero i

Para i🡨0 hasta n-1 inc 1 hacer

Escribir “Dato”, i+1, “:”

Escribir “valor [“, i+1, “]: “

Leer v[i].valor

Escribir “pes [“, i+1, “]: “

Leer v[i].peso

Fin \_ Para

Fin \_ ingreso

Reportedato (d ato v [], entero n)

Entero i

Escribir “datos ingresados”

Para i🡨0 hasta n-1 inc 1 hacer

Escribir v[i].valor, “, “, v [i].peso

Fin \_ Para

Fin \_ reportedato

Promedioponderado (dato v [], entero n)

Entero i

Escribir “datos ingresados”

Para (i🡨0 hasta n-1 inc 1) hacer

S1🡨 s1 + v [i].valor \* v[i].peso

S2🡨s2+v[i].peso

Fin \_ Para

Fin \_ promedioponderado

EJERCICIO 04:

Algoritmo estructura 04

Entero op, n

Empleado v [100]

numEmpleados (n)

ingresoEmpleados (v, n)

Hacer

Escribir “Menu”

Escribir “[1] Consultar empleado”

Escribir “[2] Eliminar empleado”

Escribir “[3] Ordenar por salario descendentemente”

Escribir “[4] Mostrar empleados”

Escribir “[5] Mostrar empleados”

Escribir “[6] Salir”

Escribir “Ingrese opción (1-6): “

Leer op

Según \_ Sea\_op hacer

Caso 1: consultarEmpleado (v, n)

salir

Caso 2: eliminarEmpleado (v, n)

salir

Caso 3: ordenarPorSalario (v, n)

Sistema Salir

Escribir “Datos ordenador por salario”

reporteEmpleados (v, n)

sistema pausa

salir

case 4:

sistema salir

reporteEmpleados (v, n)

sistema pausa

caso 5:

sistema salir

Escribir “el promedio de sueldos es: “, promedioSalarios (v, n)

Salir pausa

Fin \_ Según \_sea

Mientras (op<>6)

Fin \_ Algoritmo

numEmpleados (entero n (R))

hacer

escribir “Numero de empleados : “

leer n

Mientras (n<=0)

Fin \_ numEmpleados

ingresoEmpleados (Empleado v[], entero n)

entero i

para (i🡨0 hasta n-1 inc 1) hacer

Escribir “Datos del empleado”, i+1

Escribir “código: “

Leer (v[i].c odigo)

Escribir “Apellidos: “

Leer (v[i].apellidos, 40)

Escribir “Nombres: “

leer (v[i].Nombres, 40)

Escribir “telefono: “

leer (v[i].telefono, 40)

Escribir “Sueldo: “

leer (v[i].salario, 40)

Fin \_ Para

Fin \_ ingresoEmpleados

reporteEmpleados (Empleado v[], entero n)

entero i

escribir “Lista de empleados”

para (i🡨0 hasta n-1 inc 1) hacer

escribir v[i].codigo, “,”, v[i].apellidos, “,”, v[i].nombres, “,”, v[i].telefono, “,”, v[i].salario

Fin \_ para

Fin \_ reporteEmpleados

buscarCodigo (empleado v[], entero n, carácter código [])

entero i

para (i🡨o hasta n-1 inc 1) hacer

si (v[i].código = código) entonces

retornar i

Fin \_ Si

Fin \_ Para

Fin \_ buscarCodigo

ConsultarEmpleado (Empleado v [], entero n)

Entero p

Carácter codigo [10]

Escribir “Ingrese codigo: “

Leer codigo

P = buscarCodigo (v, n, codigo)

Si (p!=1) entonces

Escribir “Datos del empleado : “

Escribir “Apellidos: “, v[p].apellidos

Escribir “Nombres: “, v[p].nombres

Escribir “Telefono: “, v[p].telefono

Escribir “Sueldo: “, v[p].salario

Sino

Escribir “Codigo no se encuentra”

Fin \_ Si

Fin \_ Para

eliminarEmpleado (Empleado v[], entero n (R))

Entero p

Carácter codigo [10]

Escribir “Ingrese codigo: “

Leer codigo

P = buscarCodigo (v, n, codigo)

Si (p!=1) entonces

Para (entero i🡨0 hasta n-1 inc 1) hacer

V[i]🡨 v[i+1]

Fin \_ Para

N🡨n-1

Escribir “Dato eliminado”

Sino

Escribir “El código no se encuentra”

Fin \_ Si

Fin \_ eliminarEmpleado

ordenarPorSalario (Empleado v[], entero n)

entero i, j

Empleado temp

Para (i🡨0 hasta n-1 inc 1) hacer

Para (j🡨i+1 hasta n-1 inc j) hacer

Si ((v[i].salario<v[i].salario)) entonces

Temp<- v[i]

V[i]🡨v[j]

V[j]🡨temp

Fin \_ Si

Fin \_ Para

Fin \_ Para

Fin \_ ordenarPorSalario

Promediosalarios (Empleado v[], entero n)

Entero i

Real s🡨0

Para (i🡨0 hasta n-1 inc 1) hacer

S🡨 s+ v[i].salario

Fin \_ Para

Retornar s/n

Fin \_ ordenarPorSalario

EJERCICIOS PROPUESTOS:

EJERCICIO 01:

Estructura alumno

Real e1

Real e2

Real tra

Real pro

Cadena nombre [60]

Fin\_estructura

Algoritmo alumnos\_nota

Alumno v[60]

Entero n

Real progre

Cadena resp

Hacer

Sistema salir

Ndatos n

Ingreso (v,n)

Promedio (v,progre,n)

Ordenmer (v,n)

Escribir “datos en orden de merito: ”

Reporte (v,n)

Escribir “PROMEDIO GENERAl: ”,progre

Resp🡨opcion

Mientras (resp=’s’ o resp=’S’)

Fin\_algoritmo

Ndatos (entero n(R))

Hacer

Escribir “Numero de alumnos: ”

Leer n

Sistemas salir

Mientras n<=0

Fin\_ndatos

Ingreso (alumno v[],entero n)

Entero i

Para i🡨0 hasta n-1 inc 1 hacer

Escribir “ALUMNO ”,i+1,”: ”

Escribir “nombre: ”

Leerletras(v[i].nombre)

Hacer

Escribir “nota del examen 1: ”

V[i].e1🡨leerreal

Escribir “nota del examen 2:”

V[i].e2🡨leerreal

Escribir “nota de trabajo: ”

V[i].tra🡨leerreal

Mientras (v[i].e1<0 y v[i].e2<0 y v[i].tra<0 o v[i].e1>20 y v[i].e>20 y v[i].tra>20 )

Fin\_ingreso

Promedio (alumno v[],real progre(R),entero n)

Entero i

Real se,s,no

Se🡨0

S🡨0

Para i🡨0 hasta n-1 inc 1 hacer

Se🡨v[i].e1\*0,25+v[i].e2\*0,25

No🡨v[i].tra\*0,5

V[i].pro🡨se+no

S🡨s+v[i].pro

Fin\_para

Fin\_promedio

Reporte (alumno v[],entero n)

Entero i

Para i🡨0 hasta n-1 inc 1 hacer

Escribir ,v[i].nombre,”, el promedio ponderado es: ”,v[i].pro

Fin\_para

Fin\_reporte

ordenmer (alumno v[], entero n)

Entero i ,j

Cadena temp1 [60]

Real temp2

Para i🡨0 hasta n-1 inc 1 hacer

Para j🡨i+1 hasta n-1 inc 1 hacer

Si (v[i].pro<v[j]pro)

Temp1🡨v[i].nombre

v[i].nombre🡨 v[j].nombre

v[j].nombre🡨temp1

temp2🡨v[i].pro

v[i].pro🡨v[j].pro

v[j].pro🡨temp2

fin\_si

fin\_para

fin\_para

fin\_promedio

EJERCICIO 02:

Estructura banco

Cadena código [60]

Cadena nombre [60]

Real saldo

Fin\_estructura

Algoritmo banco\_menu

Entero n,opción

Sistema salir

Banco v[100]

Ndatos (n)

Ingreso (v,n)

Hacer

Sistema salir

escribir"MENU DEL BANCO”

escribir "[1] Crear una cuenta"

escribir "[2] Consultar saldo"

escribir "[3] Depositar saldo"

escribir "[4] Retirar Saldo"

escribir "[5] Mostrar cuentas ingresadas"

escribir "[6] Salir"

escribir "Ingrese opcion (1-6): "

leer opcion

según\_sea\_opcion hacer

caso 1:

crearcuenta(v,n)

salir

caso 2:

consultarsaldo(v,n)

salir

caso 3:

depositarsaldo (v,n)

salir

caso 4:

retirarsaldo (v,n)

salir

caso 5:

mostrarcuenta (v,n)

salir

fin\_segun\_sea

mientras (opción !=6)

fin\_algoritmo

ndatos (entero n (R))

hacer

escribir “ingrese el total de clientes: ”

n🡨leerEntero

mientras n<=0

fin\_ndatos

busqueda (banco v[], entero n, cadena codbus [])

entero i

para i🡨0 hasta n-1 inc 1 hacer

si v[i]codigo=codbus

retorno i

fin\_si

fin\_para

retornio -1

fin\_busqueda

ingreso (banco v[], entero n)

entero i

para i🡨0 hasta n-1 inc 1 hacer

escribir “ingrese los datos del cliente ”,i+1,”: ”

escribir “ingrese su nombre: ”

leerLetras(v[i].nombre)

escribir “ingrese codigo: ”

leerLetrasYNumeros (v[i].codigo)

hacer

escribir “ingrese su saldo: ”

v[i].saldo🡨leerReal

mientras v [i].saldo<0

fin\_para

fin\_ingreso

crearcuenta (banco v[], entero n(R))

escribir “ingrese datos del nuevo cliente”

escribir “ingrese su nombre: ”

leerLetras(v[i].nombre)

escribir “ingrese codigo: ”

leerLetrasYNumeros (v[i].codigo)

escribir “ingrese su saldo: ”

v[i].saldo🡨leerReal

n🡨n+1

fin\_crearcuenta

consultarsaldo (banco v[], entero n)

cadena codbus [60]

entero p

sistema salir

escribri “ingrese codigo del cliente”

leerLterasYNumeros (codbus)

p🡨busqueda (codigo,n,codbus)

si p!=-1 entonces

escribir “DATOS DEL CLIENTE”

escribir ”Codigo: ”, v[p].codigo

escribri “Nombre”, v[p].nombre

escribir “el saldo del cliente es:”,v[p].saldo

sino

esribir “el codigo no se cuenta”

sistema pausa

fin\_consultarpersona

depositarsaldo (banco v[], entero n)

cadena codbus [60]

real d

entero p

sistema salir

escribir “ingrese codigo del cliente: ”

leerLterasYNumeros (codbus)

p🡨 busqueda(codigo,n,codbus)

si p!=-1 entonces

escribir “DATOS DEL CLIENTE”

escribir ”Codigo: ”, v[p].codigo

escribri “Nombre”, v[p].nombre

escribir “el saldo del cliente es:”,v[p].saldo

escribir “cuanto de saldo depositara: ”

d🡨leerReal

v[p].saldo🡨v[p].saldo+d

escribir “el saldo nuevo es: ”,v[p].saldo

fin\_si

sino

escribir “el codigo no se encuentra”

fin\_si

sistema pausa

fin\_depositarsaldo

retirarsaldo (banco v[], entero n)

cadena codbus [60]

real r

entero p

sistema salir

escribir “ingrese codigo del cliente: ”

leerLterasYNumeros (codbus)

p🡨 busqueda(codigo,n,codbus)

si p!=-1 entonces

escribir “DATOS DEL CLIENTE”

escribir ”Codigo: ”, v[p].codigo

escribri “Nombre”, v[p].nombre

escribir “el saldo del cliente es:”,v[p].saldo

hacer

escribir “ingrese cuanto saldo retirara: ”

r🡨leerReal

r🡨leerReal

si (r<=0 o r>v[p].saldo)entonces

escribir “el retiro debe ser una cantidad positiva y menor al saldo actual:”

fin\_si

mientras r<=0 o r>v[p].saldo

v[p].saldo🡨v[p].saldo-r

escribir “el saldo nuevo es: ”,v[p].saldo

sino

escribir “el codigo no se encuentra”

fin\_si

sistema pausa

fin\_retirarsaldo

reporte (banco v[],entero n)

entero i

para i🡨0 hasta n-1 inc 1 hacer

escribir ,v[i].nombre,”, “v[i].codigo”, ”,v[i]saldo

fin\_para

fin\_reporte

mostrarcuentas (banco [],entero n)

sistema salir

escribir “DATOS DE TODAS LAS PERSONAS: ”

reporte (v,n)

sistema pausa

fin\_mostrarcuentas

EJERCICIO 03:

Estructura polinomio

Entero coef

Fin\_estructura

Algoritmo\_polinomio

Polinomio v[100]

Entero x,n

Cadena resp

Hacer

Sistema salir

Ngrado (n)

Valor (x)

Poli (v,n)

Reportep (v,n)

Escribir “el valor del polinomio: ”,calculo (v,n,x)

Resp🡨opcion

Mientras (resp=’s’ o resp=’S’)

Fin\_algoritmo

Ngrado (entero n(R))

Hacer

Escribir “el grado del polinomio: ”

N🡨leerEntero

Mientras n<=0

Fin\_ngrado

Valor (entero x(R))

Escribir “ingrese un valor para x: ”

X🡨leerentero

Fin\_valor

Poli (polinomio v[],entero n)

Entero i

Para i🡨n hasta 0 dis 1 hacer

Si (i=0) entonces

Escribir “ULTIMO TERMINO”

Sino

Escribir “TERMINO”

Escribir “ingrese un valor para el coeficiente: ”

V[i].coef🡨leerReal

Fin\_para

Fin\_poli

Reportep (polinomio v[],entero n)

Entero i

Escribir “la ecuación polinómica es: ”

Para i🡨n hasta 0 dis 1 hacer

Si (i != 0) entonces

Si (i=n) entonces

Si (v[i].coef>0 || v[i]<0)

Escribir ,v[i].coef,

Sino

Escribir “”

Fin\_si

Fin\_si

Fin\_si

Fin\_para

Sino

Si (v[i].coef>0)

Escribir “+”,v[i].coef

Sino

Si (v[i].coef>0) entonces

Escribir “+”,v[i].coef,”x^”,i

sino

Si (v[i].coef<0)

Escribir ,v[i].coef

Sino

Escribir “”

Fin\_si

Fin\_si

Fin\_si

Sino

Si (v[i].coef>0)entonces

Escribir “+”,v[i].coef

Sino

Si (v[i].coef<0) entonces

Escribir ,v[i].coef

Sino

Escribir “”

Fin\_si

Fin\_si

Fin\_si

Fin\_si

Escribir “cambio de linea”

Fin\_reportep

Real calculo (polinomio v[],entero n,entero x)

Entero i,s

S🡨0

Para i🡨n hasta 0 dis 1 hacer

S🡨s+v[i].coef\*x^i

Fin\_para

Retorno s

Fin\_calculo

EJERCICIO 04:

Estructura trabajadora

Cadena cod [60]

Char nombre [60]

Real valor

Real horastrab

Real t

Fin\_estructura

Algoritmo trabajador

Trabajador v[100]

Entero n,op

Ndatos (n)

Ingresarrabajador (v,n)

Hacer

Sistema salir

Escribir “[1]Mostrar trabajador”

Escribir “[2]Mostrar trabajadores ordenados por nombre”

Escribir “[3]Mostrar trabajadores ordenados por el sueldo”

Escribir “[4] Salir”

Escribir “ingrese una opcion (1-4)”

Op 🡨 leerentero

Según\_sea\_op

Caso 1:

Mostrartrabajadores (v,n)

Salir

Caso 2:

Trabajadoresordenadospornombre (v,n)

salir

Caso 3:

Trabajadoresordenadosporsueldo (v,n)

Salir

Fin\_segun\_sea

Mientras op!=4

Fin\_algoritmo

Ndatos (entero n(R))

Hacer

Escribir “ingrese el numero de trabajadores: ”

Leer n

Mientras n<=0

Fin\_ndatos

Ingresartrabajador (trabajador v[],entero n)

Entero i

Para i🡨0 hasta n-1 inc 1 hacer

Escribir “trabajador”,i+1,”: “

Escribir “ingrese código: ”

leerLetras (v[i].cod)

escribir “ingrese el nombre: ”

leerLetras (v[i].nombre)

escribir “el valor por horas: ”

v[i].horastrab🡨leerEntero

escribir “ingrese las horas trabjadas: ”

v[i].horastrab🡨leerEntero

v[i].t🡨v[i].horastrab\*v[i].valor

sistemas salir

fin\_para

fin\_ingresartrabajador

entero búsqueda (trabajador v[],entero n,codbus[])

entero i

para i🡨0 hasta n-1 inc 1 hacer

si (v[i].cod=codbus)

retorno i

fin\_si

fin\_para

retorno -1

fin\_busqueda

mostrartrabajador (trabajador v[],entero n)

cadena codbus [20]

entero p

sistema salir

escribir “código a buscar: ”

leerLetrasYNumeros (codbus)

p🡨busqueda (v,n,codbus)

si (p!=-1)

escribir “DATOS DE LA PERSONA”

escribir “código ”, v[p].cod

escribir “nombre ”,v[p].nombre

escribir “valor por hora”,v[p].valor

escribir “horas trabajadas”,v[p]horastrab

escribir “el salario es: ”v[p].t

sino

escribir “el nombre no se encuentra”

sistema pausa

fin\_mostrartrabajador

reporte (trabajador v[],entero n)

entero i

para i🡨0 hasta n-1 inc 1 hacer

escribir ,v[i].cod,”, ”,v[i].nombre,”, valor de hora: ”,v[i].valor,”, horas trabajadas”,v[i].horastrab,”, sueldo: ”,v[i].t

fin\_para

fin\_reporte

trabajadoresordenadospornombre (trabajador v[],entero n)

entero i,j

cadena temp1 [20],temp2 [60]

real temp3,temp4,temp5

para i🡨0 hasta n-2 inc 1 hacer

para i🡨j+1 hasta n-1 inc 1 hacer

si (v[i].nombre>v[j].nombre)

temp🡨v[i].cod

v[i].cod🡨v[j].cod

v[i].cod🡨temp

temp2🡨v[i].nombre

v[i].nombre🡨v[j].nombre

v[i].nombre🡨temp2

temp3🡨v[i].valor

v[i].valor🡨v[j].valor

v[j].valor🡨temp3

temp4🡨v[i].horastrab

v[i].horastrab🡨v[j].horastrab

v[i].horastrab🡨temp2

temp5🡨v[i].t

v[i].t🡨v[j].t

v[j].t🡨temp5

fin\_si

fin\_para

fin\_para

fin\_trabajadoresordenadospornombre

mostrarpersonas (trabajador v[],entero n)

sistema salir

escribir “DATOS ORDENADOS”

reporte (v,n)

sistema pausa

fin\_mostrarpersonas

trabajadoresordenadosporsueldo (trabajador v[],entero n)

entero i,j

cadena temp1 [20],temp2 [60]

real temp3,temp4,temp5

para i🡨0 hasta n-2 inc 1 hacer

para i🡨j+1 hasta n-1 inc 1 hacer

si (v[i].t<v[i].t) entonces

temp🡨v[i].cod

v[i].cod🡨v[j].cod

v[i].cod🡨temp

temp2🡨v[i].nombre

v[i].nombre🡨v[j].nombre

v[i].nombre🡨temp2

temp3🡨v[i].valor

v[i].valor🡨v[j].valor

v[j].valor🡨temp3

temp4🡨v[i].horastrab

v[i].horastrab🡨v[j].horastrab

v[i].horastrab🡨temp2

temp5🡨v[i].t

v[i].t🡨v[j].t

v[j].t🡨temp5

fin\_si

fin\_para

fin\_para

fin\_trabajadoresordenadosporsueldo

EJERCICIO 05:

Estructura fracción

Entero numerador

Entero denominador

Fin\_estructura

Algoritmo fracción

Fracción v[100]

Entero n,num,dem

Real v1[60]

Hacer

Sistema salir

Ndato (n)

Ingreso (v,v1,n)

Escribir “reportamos la fraccion”

Reporte (v,n)

Escribir “cambio de linea”

Sumafraccion (v,n,v1,num,dem)

Escribir “la suma de las fracciones es: ”,num,”/”,dem

Simplificar (num,dem)

Escribir “fracción simplificada: ”,num,”/”,dem

Resp🡨opcion

Mientras (resp=’s’ o resp=’S’)

Fin\_algoritmo

Ndato (entero n(R))

Hacer

Escribir “numero de términos (>0)”

N🡨leerEntero

Mientras n<=0

Fin\_ndato

Reporte (fracción v[],entero n)

Para i🡨0 hasta n-1 inc 1 hacer

Si (i=0) entonces

Escribir ,v[i].numerador,” / ”,v[i].denominador

Sino

Escribir “ + ”,v[i].numerador,” / ”,v[i].denominador

Fin\_si

Fin\_para

Fin\_reporte

Sumafraccion (fracción v[],entero n,real v1[],entero num (R),entero dem(R))

Entero i,x,j,y,s,may

Real p

May🡨v[0].denominador

Para i🡨0 hasta n-1 inc 1 hacer

Si (v[i].denominador>may) entonces

May🡨v[i].denominador

Fin\_si

X🡨2

P🡨1

Mientras x<=may hacer

Para i🡨0 hasta n-1 inc 1 hacer

Si (v[i].denominado%x=0)

P🡨p\*x

Salir

Fin\_si

Fin\_para

Para i🡨0 hasta n-1 inc 1 hacer

Si (v[i].denominador%x=0) entonces

V[i].denominador🡨v[i].denominador/x

Fin\_si

Fin\_para

Y🡨x

X🡨x+1

Para i🡨0 hasta n-1 inc 1 hacer

Si (v[i].denominador%y=0) entonces

X🡨x-1

salir

Fin\_si

Fin\_para

Dem🡨p

S🡨0

Para j🡨0 hasta n-1 inc 1 hacer

S🡨s+((dem/v1[j])\*v[j].numerador)

Fin\_para

Num🡨s

Fin\_mientras

Fin\_sumafraccion

Simplificar (entero num(R), entero (R))

Entero i

I🡨2

Mientras i<=num y i<=dem hacer

Si (num%i=0 y dem%i=0) entonces

Num🡨num/i

Dem🡨dem/i

Sino

I🡨i+1

Fin\_mientras

Fin\_simplificar

EJERCICIO 06:

Estructura cadena

Cadena pal [60]

Entero num

Fin\_estructura

Algoritmo cadena\_Ascendente

Cadena v[100]

Entero n

Cadena resp

Hacer

Sistema salir

Ndato (n)

Ingresar (v,n)

Ascendente (v,n)

Escribir “FORMA ASCENDENTE: ”

Reporte (v,n)

Resp🡨opcion

Mientras resp=’s’ o resp=’S’

Fin\_algoritmo

Ndato (entero n(R))

Hacer

Escribir “ingrese el numero de cadenas a ingresar: ”

N🡨leerEntero

Mientras n<=0

Fin\_ndato

Ingresar (cadena v[],entero n)

Entero i

Para i🡨0 hasta n-1 inc 1 hacer

leerLetras (v[i].pal)

v[i].num🡨strlen (v[i].pal)

fin\_para

fin\_ingresar

reporte (cadena v[], entero n)

entero i

para i🡨0 hasta n-1 inc 1 hacer

escribir ,v[i].pal,” = ”,v[i].num

fin\_para

fin\_reporte

ascendente (cadena v[],entero n)

entero i,j,temp1

cadena temp [60]

para i🡨0 hasta n-2 inc 1 hacer

para j🡨 i+1 hasta n-1 inc 1 hasta

si (v[i].num > v[j].num)

temp🡨v[i].pal

v[i].pal🡨v[j].pal

v[i].pal🡨temp

temp1🡨v[i].num

v[i].num🡨v[j].num

v[j].num🡨temp1

fin\_si

fin\_para

fin\_para

fin\_ascendente

EJERCICIO 07:

Estructura alumnos

Cadena código [60]

Cadena nombre [60]

Cadena apellidos [60]

Real d

Real m

Real a

Fin\_estructura

Algoritmo alumno\_orden\_fecha de inscripción

Alumnos v[100]

Entero n

Cadena resp

Hacer

Sistema salir

Ndatos (n)

Ingreso (v,n)

Ordenarfecha (v,n)

Escribir “DATOS ORDENADOS POR FECHA DE INSCRIPCION”

Reporte (v,n)

Resp🡨opcion

Mientras (resp=’s’ o resp=’S’)

Fin\_algoritmo

Ndatos (entero n(R))

Hacer

Escribir “ingrese el numero de alumnos: ”

N🡨leerEnteros

Mientras n<=0

Fin\_ndatos

Ingreso (alumno v[],entero n)

Entero i

Para i🡨0 hasta n-1 inc 1 hacer

Escribir “ALUMNO”,i+1,”: ”

Escribir “Codigo: ”

leerLetrasYNumeros (v[i].codigo)

escribir “Apellidos: ”

leerLetras(v[i].apellidos)

escribir “Nombre: ”

leerLetras(v[i].nombre)

hacer

escribir “FECHA DE INGRESO”

escribir “Dia: ”

v[i].d=leerEntero

escribir “Mes: ”

v[i].m=leerEntero

escribir “Año: ”

v[i].a=leerEntero

mientras (v[i].d<0 y v[i].m<0 y v[i].a<0)

fin\_para

fin\_ingreso

reporte (alumnos v[],entero n)

entero i

para i🡨0 hasta n-1 inc 1 hacer

escribir ,v[i].codigo,”, ”,v[i].apellidos,”, ”,v[i].nombre,”, ”v[i].d,”-”,v[i].m,”-”,v[i].a

fin\_para

fin\_reporte

ordenarfecha (alumno v[],entero n)

entero i,j

cadena temp [60],temp1 [60],temp2 [60]

real temp3,temp4,temp5

para i🡨0 hasta n-2 inc 1 hacer

para j🡨i+1 hasta n-1 inc 1 hacer

si (v[i].a>v[j].a) entonces

temp🡨v[i].codigo

v[i].codigo🡨v[j].codigo

v[i].codigo🡨temp

temp1🡨v[i].apellidos

v[i].apellidos🡨v[j].apellidos

v[i].apellidos🡨temp1

temp2🡨v[i].nombre

v[i].nombre🡨v[j].nombre

v[i].nombre🡨temp2

temp3🡨v[i].d

v[i].d🡨v[j].d

v[j].d🡨temp3

temp4🡨v[i].m

v[i].m🡨v[j].m

v[i].m🡨temp4

temp5🡨v[i].a

v[i].a🡨v[j].a

v[j].a🡨temp5

fin\_si

si (v[i]a=v[j].a) entonces

si (v[i].m>v[i].m) entonces

temp🡨v[i].codigo

v[i].codigo🡨v[j].codigo

v[i].codigo🡨temp

temp1🡨v[i].apellidos

v[i].apellidos🡨v[j].apellidos

v[i].apellidos🡨temp1

temp2🡨v[i].nombre

v[i].nombre🡨v[j].nombre

v[i].nombre🡨temp2

temp3🡨v[i].d

v[i].d🡨v[j].d

v[j].d🡨temp3

temp4🡨v[i].m

v[i].m🡨v[j].m

v[i].m🡨temp4

temp5🡨v[i].a

v[i].a🡨v[j].a

v[j].a🡨temp5

fin\_si

fin\_si

si (v[i].a=v[j].a y v[i].m=v[j].m) entonces

si (v[i].d>v[i].d) entonces

temp🡨v[i].codigo

v[i].codigo🡨v[j].codigo

v[i].codigo🡨temp

temp1🡨v[i].apellidos

v[i].apellidos🡨v[j].apellidos

v[i].apellidos🡨temp1

temp2🡨v[i].nombre

v[i].nombre🡨v[j].nombre

v[i].nombre🡨temp2

temp3🡨v[i].d

v[i].d🡨v[j].d

v[j].d🡨temp3

temp4🡨v[i].m

v[i].m🡨v[j].m

v[i].m🡨temp4

temp5🡨v[i].a

v[i].a🡨v[j].a

v[j].a🡨temp5

fin\_si

fin\_si

fin\_para

fin\_para

fin\_ordenarfecha

EJERCICIO 08:

Estructura empleados

Entero cod

Cadena nombre [60]

Real ventas

Real salario

Finb\_estructura

Algoritmo empleado

Empleados (n)

Entero n

Cadena resp

Hacer

sistema salir

nempleados (n)

ingreso (v,n)

escribir “Datos del empleados quien obtuvo mayor ventas anuales”

mayventas (v,n)

escribir “Incrementemos al 10% de salartio a los empleados que obtuvieron ventas anuales mayores a 10000”

escribir “nuevos salarios: ”

incrementosalario (v,n)

escribir “Empleados que obtuvieron ventas inferiores al promedio de las ventas”

maypromedio (v,n)

resp🡨opcion

mientras (resp=’s’ o resp=’S’)

fin\_algoritmo

empleados (entero n(R))

hacer

escribir “cantidad de empleados: ”

n🡨leerEntero

mientras n<=0

fin\_empleados

ingreso (empleados v[],entero n)

entero i

para i🡨0 hasta n-1 inc 1 hacer

escribir “datos del empleado “,i+1

escribir “ingrese su código: ”

v[i].cod🡨leerEntero

escribir “ingrese su nombre: ”

leerLetras(v[i].nombre)

hacer

escribir “ingrese ventas anuales: ”

v[i].ventas🡨leerReal

mientras v[i].ventas<=0

hacer

escribir “ingrese el salario de este empleado: ”

v[i].salario🡨leerReal

mientras v[i].salario<=0

fin\_para

fin\_ingreso

mayventas (empleados v[],entero n)

entero i

real may

may🡨v[0].ventas

para i🡨0 hasta n-1 inc 1

si (v[i].ventas>may) entonces

may🡨v[i].ventas

fin\_si

fin\_para

para i🡨0 hasta n-1 inc 1

si (may=v[i].ventas) entonces

escribir “su código es:”,v[i].cod

escribir “su nombre es:”,v[i].nombre

fin\_si

fin\_para

fin\_mayventas

incrementosalario (empleados v[],entero n)

entero i

para i🡨0 hasta n-1 inc 1

si (v[i].salario>100000) entonces

v[i].salario🡨v[i].salario+0,1\*v[i].salario

fin\_si

fin\_para

para i🡨0 hasta n-1 inc 1 hacer

escribir “empleado ”,i+1,”, salario:”,v[i].salario

fin\_para

maypromedio (empleado v[],entero n)

entero i

real su,pro

su🡨0

para i🡨0 hasta n-1 inc 1 hacer

su🡨su+v[i].ventas

fin\_para

pro🡨su/n

para i🡨0 hasta n-1 inc 1 hacer

si (v[i].ventar<pro) entonces

escribir “empleados ”,i+1,”nombre: ”,v[i].nombre, “, código:”,v[i].cod,”ventas anuales: ”,v[i].ventas

fin\_si

fin\_para

fin\_maypromedio

EJERCICIO 09:

Estructura departamento

Cadena código [60]

Cadena nombre [60]

Cadena depa[60]

Cadena grado [60]

Cadena nac[60]

Real salario

Real monto

Fin\_estructura

Algoritmo profesores\_departamento

Departamento v[100]

Entero n

Nprofesores (v,n)

Ingreso (v,n)

Escribir “el profesor que gano mayor monto el año anterior: ”

Maymonto (v,n)

Montoanual (v,n)

Escribir “El monto total pagado a los profesores extrajeros es:”

Depay (v,n)

Fin\_algoritmo

nprofesores (entero n(R))

hacer

escribir “ingrese el numero de profesores: ”

leer n

mientras n<=0

fin\_nprofesores

ingreso (departamento v[],entero n)

entero i

para i🡨0 hasta n-1 inc 1 hacer

escribir “ingrese daots del profesor: ” ,i+1

escribir “ingrese su código: ”

leerletrasYNumeros(v[i].codigo)

escribir “ingrese su nombre del profesor”

leerLetras(v[i].nombre);

escribir “ingrese el departamento”

leerLetras (v[i).depa)

escribir “ingrese el grado académico: ”

leerLetras (v[i].grado);

escribir “ingrese nacionalidad: ”

leerLetras (v[i].nac);

hacer

escribir “ingrese salario del profesor: ”

v[i].salario🡨leerReal

v[i].monto🡨v[i].salario\*12

mientras v[i].salario<=0

fin\_para

fin\_ingreso

maymonto (departamento v[],entero n)

entero i

real may

may🡨v[0].monto

para i🡨0 hasta n-1 inc 1 hacer

si (v[i].monto>may)entonces

may🡨 v[i].monto

fin\_si

fin\_para

para i🡨0 hasta n-1 inc 1 hacer

si (may=v[i].monto)

escribir “código: ”,v[i].codigo

escribir “nombre: ”,v[i].nombre

escribir “nacionalidad: ”,v[i].nac

fin\_si

fin\_para

fin\_maymonto

montoanual (departamento v[],entero n)

entero i

real s

cadena depa 1[60]

s🡨0

escribir “ingrese el departamento: ”

leerLetras (depa1)

para i🡨0 hasta n-1 inc 1 hacer

si (depa1=v[i].depa)entonces

s🡨s+v[i].monto

fin\_si

fin\_para

escribir “El monto anual pagado en el departamento de “,depa1,” es:”,s

fin\_monto anual

real extranjeros (departamento v[],entero n)

entero i

real s

cadena nac1,nac2

nac1🡨”Peru”

nac2🡨”peru”

s🡨0

para i🡨0 hasta n-1 inc 1 hacer

s🡨s+v[i].monto

si (v[i].nac=nac1 o v[i].nac=nac2)

s🡨s-v[i].monto

fin\_si

fin\_para

si (s<0)entonces

retorno 0

retorno s

fin\_para

fin\_extranjeros

depay (departamento v[],entero n)

entero i

real may

cadena depa2 [60]

escribir “ingrese el departamento: ”

leerLetras (depa2)

para i🡨0 hasta n-1 inc 1 hacer

si (depa2=v{i]depa)

si v[i].monto>may entonces

may🡨v[i].monto

fin\_si

fin\_si

fin\_para

para i🡨0 hasta n-1 inc 1 hacer

si may=v[i].monto entonces

escribir “El que mas gano en el departamento ”,depa2, “el año anterior fue: ”,v[i].nombre

fin\_si

fin\_para

fin\_depay