|  |  |
| --- | --- |
| Colegio Universitario **IES***Siglo 21* | |
| **EXAMEN IEFI** | |
| **Materia:** Programación Lógica 2 | **Carrera:** Analista de Sistemas |
| **Modalidad:** Presencial/Distancia | **Centro de Extensión:** Córdoba |

**Reservado para el alumno**

|  |  |
| --- | --- |
| **Alumno**: Sebastian Barcovsci | **Docente:** |
| **DNI**: | **Fecha**: |

**Reservado para el docente**

|  |
| --- |
| NOTA |
|  |

**CONSIDERACIONES GENERALES**

* **Temas:**
* Algoritmia computacional. Análisis de los requerimientos, tratamiento de los datos de entrada y producción de la salida de un programa.
* Estructuras de programación: secuencial, alternativa y repetitiva.
* Datos Simples: variables y constantes (contadores, acumuladores y otros usos particulares).
* Estructuras homogéneas y heterogéneas de datos: Vectores, Registros, Agregados de Datos.
* Métodos de búsqueda y ordenamiento de datos.
* Procesamiento de Agregados de Datos.
* **Objetivos:**
* Aplicar lógica algorítmica para la resolución de problemas computaciones concretos.
* Desarrollar en lenguaje de pseudocódigo los algoritmos necesarios para procesar los datos en orden a los requerimientos.
* **Criterio de Evaluación:**

Se tendrá en cuenta para la evaluación:

* Correcto funcionamiento de los procedimientos desarrollados.
* Estructura general de un programa especificado en pseudocódigo.
* Nomenclatura de variables del programa.
* Precisión en las respuestas y la sintaxis de los pseudocódigos.
* Uso adecuado de las estructuras de programación y el manejo de datos y variables para la producción de los resultados requeridos.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **%** | 0-19 | 20-39 | 40-59 | **60-65** | **66-71** | **72-77** | **78-83** | **84-89** | **90-95** | **96-100** |
| **Nota** | 1 | 2 | 3 | **4** | **5** | **6** | **7** | **8** | **9** | **10** |

**CONSIGNAS**

***OBSERVACION: El parcial debe remitirlo MEDIANTE “enviar archivos”, Tome en cuenta cortes de luz e internet, el examen está desarrollado para hacerse en 90 minutos.***

**Actividad 1:**

Recibe un vector llamado RC[25], tipo numérico cargado con fechas. Deberá cargar en una grilla, fecha, nombre, detalle del tipo de cuenta y nombre de la sucursal.

**Actividad 2:**

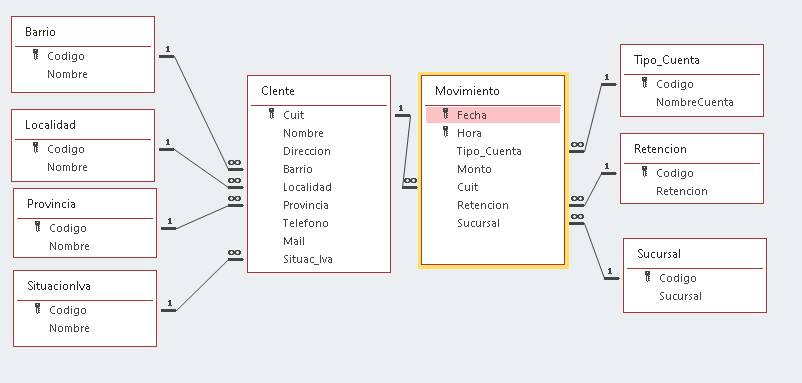
En una grilla mostrará seleccionando un cuit, el total de movimientos en cantidad y total por sucursal, validará si existe dicho cuit.

**Actividad 3:**

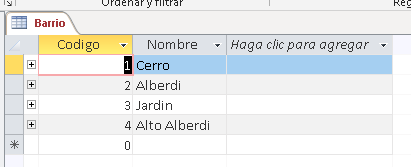
Ingresara una sucursal y mostrara en una grilla solo el total de cada tipo de cuenta para la sucursal elegida.

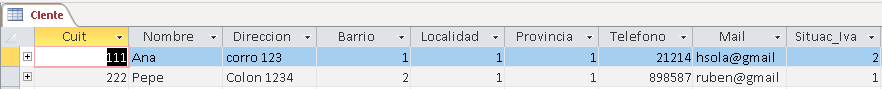
***Obs: se adjunta base de datos y tablas de ejemplo, recuerde que son muestras.***

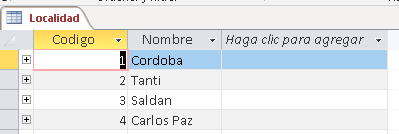
***Base de Datos CARC.***

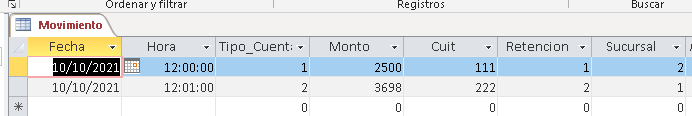


Muestras de Tablas

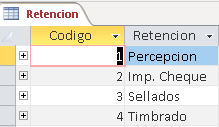


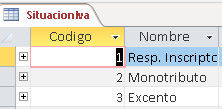


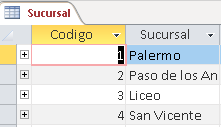


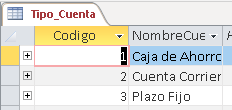












**Definicion de registros, a modo de referencia, para todas las actividades:**

//Registro declarado solo con los elementos necesarios

Registro Movimiento

Variable Fecha tipo numérica

Variable Cuit tipo numérica //Clave foránea

Variable TipoCuenta tipo numérica //Clave foránea

Variable Sucursal tipo numérica //Clave foránea

Fin Registro

//Registro declarado solo con los elementos necesarios

Registro Cliente

Variable Cuit tipo numérica //Clave foránea (Movimiento.Cuit)

Variable Nombre tipo cadena de caracteres

Fin Registro

Registro TipoCuenta

Variable Codigo tipo numérica //Clave foránea (Movimiento.TipoCuenta)

Variable NombreCuenta tipo cadena de caracteres

Fin Registro

Registro Sucursal

Variable Codigo tipo numérica //Clave foránea (Movimiento.Sucursal)

Variable Sucursal tipo cadena de caracteres

Fin Registro

Agregado de Datos ADMovimiento tipo Registro Movimiento

Agregado de Datos ADCliente tipo Registro Cliente

Agregado de Datos ADTipoCuenta tipo Registro TipoCuenta

Agregado de Datos ADSucursal tipo Registro Sucursal

**Actividad 1:**

Procedimiento cmdImprimir(Vector[] RC tipo numérica)

Abrir ADMovimiento

Abrir ADCliente

Abrir ADTipoCuenta

Abrir ADSucursal

Leer ADMovimiento

Leer ADCliente

Leer ADTipoCuenta

Leer ADSucursal

Mientras RC[] != FIN DEL ARREGLO

Variable IND tipo numérica

IND = 0

Mientras ADMovimiento != EOF && ADMovimiento.Fecha == RC[IND]

Variable Nombre tipo cadena de caracteres

Variable TipoCuenta tipo cadena de caracteres

Variable Sucursal tipo cadena de caracteres

Mientras ADCliente != EOF && ADMovimiento.Cuit == ADCliente.Cuit

Nombre = ADCliente.Nombre

Leer ADCliente

Fin Mientras

Mientras ADTipoCuenta != EOF && ADMovimiento.TipoCuenta == ADTipoCuenta.Codigo

TipoCuenta = ADTipoCuenta.NombreCuenta

Leer ADTipoCuenta

Fin Mientras

Mientras ADSucursal != EOF && ADMovimiento.Sucursal == ADSucursal.Codigo

Sucursal = ADSucursal.Sucursal

Leer ADSucursal

Fin Mientras

Grilla.AgregarFila( RC[IND], Nombre, TipoCuenta, Sucursal )

IND++

Leer ADMovimiento

Fin Mientras

Fin Procedimiento

**Actividad 2:**

Procedimiento cmdImprimir(Variable Cuit tipo numérica )

Variable Cantidad tipo numérica

Abrir ADCliente

Abrir ADMovimiento

Abrir ADSucursal

Leer ADCliente

Leer ADMovimiento

Leer ADSucursal

Mientras ADCliente != EOF

Si ADCliente.Cuit == Cuit Entonces

Mientras ADSucursal != EOF

Cantidad = 0

Mientras (ADMovimiento != EOF && ADMovimiento.Cuit == ADCliente.Cuit &&

ADMovimiento.Sucursal == ADSucursal.Codigo)

Cantidad++

Leer ADMovimiento

Fin Mientras

Grilla.AgregarFila( ADSucursal.Sucursal, Cantidad )

Leer ADSucursal

Fin Mientras

Sino

Mensaje(“Cliente no encontrado”)

Fin Si

Leer ADCliente

Fin Mientras

Fin Fin Procedimiento

**Actividad 3:**

Procedimiento cmdImprimir(Variable Sucursal tipo Cadena de texto )

Variable Cantidad tipo numérica

Abrir ADTipoCuenta

Abrir ADMovimiento

Abrir ADSucursal

Leer ADTipoCuenta

Leer ADMovimiento

Leer ADSucursal

Mientras ADSucursal != EOF

Si ADSucursal.Sucursal == Sucursal

Mientras ADTipoCuenta != EOF

Cantidad = 0

Mientras (ADMovimiento != EOF && ADMovimiento.Sucursal == ADSucursal.Codigo

&& ADMovimiento.TipoCuenta == ADTipoCuenta.Codigo)

Cantidad++

Leer ADMovimiento

Fin Mientras

Grilla.AgregarFila( TipoCuenta.NombreCuenta, Cantidad )

Leer ADTipoCuenta

Fin Mientras

Sino

Mensaje(“Sucursal no encontrada”)

Fin Si

Leer ADSucursal

Fin Mientras

Fin Fin Procedimiento