UNIVERSIDAD DISTRITAL



FRANCISCO JOSÉ DE CALDAS

### CIENCIAS DE LA COMPUTACIÓN 1

*PROYECTO FINAL 2018-3*

Una empresa desea recoger información acerca de sus empleados con el fin de hallar perfiles que les permitan enfocar el desarrollo de las labores de manera más eficiente.  Para ello, los gerentes ordenan la elaboración de formularios que son entregados en cada una de las sucursales ubicadas a lo largo y ancho del país (puede haber más de una sucursal en la misma ciudad). La información que se debe consignar allí es la siguiente:

* Nombre
* Apellido
* Tipo de identificación: puede ser uno de los siguientes valores: Cédula de ciudadanía, cédula de extranjería o tarjeta de identidad.
* Número de identificación
* Sexo: F o M
* Teléfono celular
* Teléfono fijo
* e-mail
* Fecha de Nacimiento
* Ciudad de Nacimiento
* País de Nacimiento
* Ciudad de Residencia
* Dirección
* Barrio
* Actividad laboral: Artes, ciencias sociales, ingenierías, áreas de la salud, otros
* Tiene hijos?: como respuesta se acepta S ó N
* Número de hijos
* Listado con el nombre y fecha de nacimiento de cada uno de los hijos: 0-5, 6-10, 11-15, mayor de 16
* Nombre de la sucursal en la que trabaja

Datos de la sucursal.

* Nombre de la sucursal
* Dirección de la sucursal
* Barrio
* Nombre del gerente de la sucursal.

Para evitar que la solicitud de información se haga demasiado grande, el programa debe estar en capacidad de calcular la edad de cada una de las personas encuestadas.

**Estructura de datos:**

**En memoria secundaria:**

La información deberá estar almacenada, de manera permanente, en archivos los cuales deben ser actualizados al final del día o en su defecto al terminar la utilización del programa. Debe tenerse la posibilidad de modificar la información ya sea agregando, eliminando o cambiando registros.

Existen por lo menos los archivos planos para los empleados y las sucursales.

**En memoria Principal:**

Con el fin de optimizar las búsquedas deberán utilizarse listas, multilistas, arreglos, pilas, colas o árboles en memoria principal y para la realización de dichas búsquedas, deberá evitarse la utilización total de la información. Es decir se espera que apliquen los criterios y conocimientos correspondientes a la eficiencia de los algoritmos. Complejidad en tiempo y memoria. En cuanto a listas se dan ejemplos de la forma como se pueden estructurar:

Nodo Feligreses:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Los campos con la información de cada miembro de la comunidad. | | | | |
| sigPorSexo | sigPorLocalidad | sigPorActividad Laboral | sigPorLocalidad | sigPorNumeroDeHijos |

Las cabezas de las listas que se deben crear son:

Cabezas de Lista

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| CabezaPorSexo | |  | CabezaPorEdad | | | | |
| masculino | femenino |  | 18 a 24 | 25 a 35 | 36 a 45 | 45 a 60 | Mas de 60 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CabezaPorActividadLaboral | | | | | |
| Artes | técnico | Ingeniero | Sociales | Salud | * - - - |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CabezaPorNúmeroDeHijos | | | |  | cabezaPorLocalidad | | | | |
| Sin hijo | 1-2 | 3-4 | Mas de 4 |  | Nombre1 | Nombre 2 | Nombre 3 | Nombre 4 | - - - - |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| cabezaPorIglesia | | | | |
| Nombre1 | Nombre 2 | Nombre 3 | Nombre 4 | - - - - |

**Consultas**

 El programa debe permitir a los miembros de la comunidad religiosa obtener la siguiente información:

1. Número total de personas que trabajan en una sucursal dada, clasificándolos por rangos de edades de los hijos de la siguiente forma: sin hijos, de 0-5, 6-10, 11-18, mayor de 18.
2. Listado de los nombres y apellidos de aquellos que tienen  un número de hijos dado (en rangos), clasificándolos por ciudad y país en que viven.
3. Nombre y apellidos de las personas que viven en una ciudad dada,  clasificándolos por ciudad de nacimiento y actividad laboral.
4. Número de sucursales en las que trabaja un número de personas superior a un número dado, mostrando la cantidad de personas de cada sucursal junto con el nombre del gerente, el nombre de la sucursal y el barrio en que se encuentra ubicada dicha sucursal.
5. Obtener el número de hombres y el número de mujeres que trabajan en las diferentes sucursales, clasificando la información por país y ciudad, mostrando el nombre de la sucursal y del gerente.
6. Dado un rango de edad y una actividad laborar mostrar la lista de trabajadores de esa edad, clasificados por barrio y sucursal a la que pertenecen.

**Requisitos funcionales:**

Con el fin de optimizar las búsquedas deberán utilizarse listas y la multilista definida. Las cuales se construyen al iniciar el programa y a partir de la información almacenada en los archivos.

Debe garantizar la eficiencia en las búsquedas aprovechando las estructuras de datos que considere pertinentes.

El programa presenta un menú y submenús que facilitan las consultas.

Adicionalmente, las listas deben crearse de manera genérica, de tal forma que puedan agregarse fácilmente otras consultas adicionales por parte de la empresa.

El programa debe estar en capacidad de calcular la edad de cada una de las personas registradas.

Para la entrega y sustentación final deben incluir los archivos planos indispensables con no menos de 30 registros.

Debe aplicar programación orientada a objetos y el principio de alta cohesión y bajo acoplamiento.

Debe aplicar los conceptos de optimización estudiados en el curso. No limitarse a garantizar el funcionamiento del programa.

Al finalizar el programa se actualizan los archivos incluyendo los cambios realizados durante la ejecución. Se agregaron o retiraron feligreses.

**Condiciones de entrega**

1. Se puede elaborar en grupos de por lo menos dos y máximo tres personas (Es un trabajo en grupo). Se sustentará individualmente.

2. Se entrega a través de aula virtual una carpeta comprimida con: código fuente, ejecutable, archivos planos y documento soporte. Sencillo manual para ejecución (si se requiere).

3. El programa debe correr en DevC++,sin requerir ningún tipo de ajuste.

4. Documento soporte incluye: Diseño de la solución, Contenido y estructura de archivos planos, Estructuras de datos en memoria principal definidas, incluyendo diagrama.

5. Archivo fuente debidamente autodocumentado.

6. Los archivos planos contendrán por lo menos: 10 sucursales, 10 empleados por sucursal y 10 ciudades.

Docente*: Deicy Alvarado.*