**I. Identificación**

|  |  |  |
| --- | --- | --- |
| **Materia** | : | Estructura de Datos |
| **Universidad** | : | Universidad Columbia del Paraguay. |
| **Carrera** | : | Ingeniería Informática |
| **Curso/Año** | : | Segundo/2019 |
| **Responsable** | : | Ing. Adolfo Arteta |

**II. Objetivos**

Al culminar el trabajo práctico el estudiante estará capacitado para:

* **Conocer** diseños de programas con el enfoque de orientación a objetos en el lenguaje de programación C++.
* **Implementar** tipos de datos abstractos lineales con memoria dinámica.
* **Reconocer** y **Valorar** la importancia de los tipos de datos abstractos lineales con memoria dinámica en un enfoque orientado a objetos para la resolución de problemas.

**III. Descripción**

Escriba un programa orientado a objetos en C++ que lea datos de una agenda de personas desde un archivo ***agenda.xml*** comprobando primeramente que las etiquetas (no se sabe la cantidad) coincidan correctamente. Se puede escribir una función que lea la línea del teclado o un archivo, extraiga todas las etiquetas como cadenas, y las vaya almacenando (***Push***) en una pila (solo etiqueta de apertura), y va sacando (***Pop***) a medida que encuentra su etiqueta de cierre siempre y cuando coincida con la etiqueta de apertura que se encuentra en el tope de la pila.

Una vez que se verificó que el documento es válido, se debe almacenar en una lista doblemente enlazada circular los datos del archivo ordenado por el campo Apellido.

A continuación se listan **los campos de la agenda**:

* Código (este valor es único y no se puede modificar),
* Nombres
* Apellidos
* Fecha de nacimiento
* Sexo (1- Femenino, 2-Masculino)
* Número de documento
* Tipo de documento: (1-CI, 2-RUC, 3-Otro)
* Estado Civil(1-Soltero/a, 2-Casado/a, 3-Viudo/a, 4-Divorciado/a, 5-Separado/a)
* Nacionalidad
* Lista de Direcciones:
  + Ciudad
  + Calle
  + Número de Casa
  + Edificio/Piso/Nro. Dpto.
  + País
  + Región
* Dirección de correo electrónico
* Lista de teléfonos:
  + Tipo:1-Celular,2-Linea Baja, 3-Fax, 4-Otro
  + Número
  + Interno.

Se debe **crear un menú principal** con las siguientes **opciones**:

* + **Navegación** en pantalla de registros:
    - **Siguiente** (S), **Anterior** (A), **Primero** (P) y **Último** (U).
  + Operaciones de:
    - **Insertar** (I). {Insertar un registro. El programa debe generar automáticamente el código y mantener la lista ordenada por el campo apellido}
    - **Borrar** (B). Buscar por código de persona o sobre el registro actual.
    - **Modificar** (M). Buscar por código o sobre el registro actual. No se puede modificar el código.
    - **Consultar** (C) (buscar por código de persona).
  + Finalmente dar la opción de **salir** al usuario (Q).

**IV. Cronograma**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Trabajo Práctico** | **Puntos** | **Fecha** | **Horas** | |
| **Presentación del tema/Definición de grupos** |  | 15/10/2019 | 2 hs. | 20:10 a 22:10 |
| **Desarrollo y Corrección de avances.** | 0 | 22/10/2019 | 2 hs. | 20:10 a 22:10 |
| **Desarrollo y Corrección de avances.** | 0 | 29/10/2019 | 2 hs. | 20:10 a 22:10 |
| **Desarrollo y Corrección de avances.** | 0 | 31/10/2019 | 4 hs. | 18:00 a 20:10 |
| **Entrega Final** | 10 | 05/11/2019 | 4 hs. |  |
| **TOTAL** | **10** |  | **14 hrs** | |

**V. Sistema y Criterios de Evaluación**

1. Total de Puntos: 10.
2. Máximo de tres integrantes por grupo.
3. La fecha de entrega final será el día 05 de noviembre de 2019.
4. La evaluación final se realizará a base del código fuente de la aplicación.
5. La defensa es presencial/obligatoria y se hará en horario de clases con la presencia de los integrantes del grupo.