

## 2do parcial 2017 – Tema A

Ejercicio 1 [25 pts] Escriba una función genérica para cualquier tipo de elemento llamada `vector_a_lista(...)` que reciba un `std::vector` y retorne una `std::list` que contenga los elementos del vector de entrada, sin repetidos. Debe mantener el orden del vector. Escriba un programa cliente que genere un vector con 40 enteros aleatorios, utilice la función para pasarlos a una lista, y muestre la lista en pantalla.

Ejercicio 2 [25 pts] Se dispone de un archivo de textos `USUARIOS.TXT` donde cada línea contiene el código, sector, el apellido y el nombre . El código es un número entero y el sector es una sola palabra, y cada usuario tiene un solo nombre y apellido separado por un espacio en blanco. Escriba una clase que posea entre sus métodos: a) la función `GeneraPassword()`, encargada de crear un password concatenando el sector con la primer letra del nombre y el apellido completo de cada usuario y otra función `CreaBinario()` que genere un archivo binario que contenga los datos referentes al código, nombre, apellido y password generado de cada usuario. Programa los métodos adicionales que considere necesarios.

Ejercicio 3 [30 pts] Defina `Carta` (clase o struct) para representar a un naípe que tiene un número y un palo. Diseñe una clase llamada `Mazo` que permita manejar los datos de un grupo de cartas. La clase debe poseer métodos para: a) Guardar los datos de las cartas en un archivo binario. b) Cargar los datos de las cartas desde un archivo binario. c) Agregar una nueva carta . d) Eliminar una carta indicando su palo y número, e) retornar una carta dada su posición en el mazo. Tener en cuenta que no hay cartas repetidas en el mazo.

Ejercicio 4 [20 pts] Explique a) Puede abrir un archivo binario en modo texto? y viceversa? b) ¿Es posible escribir un objeto de tipo `std::string` en un archivo binario? Explique su respuesta. c) ¿Qué entiendo por “especializar/instanciar una plantilla”? ¿Cómo determina el compilador cuándo y cómo especializar la plantilla? d) ¿Qué es un iterador? ¿Para qué sirve? e) Explique cómo es posible ordenar un `std::vector` de structs según alguno de sus miembros.

## 2do parcial 2017 – Tema B

Ejercicio 1 [25 pts] Se dispone de un archivo de textos EQUIPO.TXT donde cada línea contiene el DNI, apellido y nombre correspondiente a un grupo de trabajo. Por persona hay un solo nombre y apellido y están separados por un espacio en blanco. Escriba una clase que posea entre sus métodos: a) la función `GeneraEmail()`, encargada de formar la dirección de correo electrónico de cada empleado del archivo, colocando la inicial del primer nombre y el apellido, seguido de `@gmail.com` y otra función `CreaBinario()` que genere un archivo binario que contenga los datos referentes al nombre, apellido, dni y email de cada persona. Programa los métodos adicionales que considere necesarios.

Ejercicio 2 [25 pts] Escriba una función genérica para cualquier tipo de elemento llamada `lista_a_vector(...)` que reciba un `std::list` y retorne un `std::vector` que contenga los elementos de la lista de entrada, sin repetidos. Escriba un programa cliente que genere una lista con 30 enteros aleatorios, utilice la función para pasarlos a un vector, y muestre el vector en pantalla.

Ejercicio 3 [30 pts] Defina `Cliente` (clase o struct) para representar a un cliente a partir de su nombre y apellido, DNI y Saldo. Diseñe una clase llamada `Facturas` que permita manejar los datos de un grupo de clientes. La clase debe poseer métodos para: a) Guardar los datos de los clientes en un archivo binario. b) Cargar los datos de los clientes desde un archivo binario. c) Agregar un cliente. d) Eliminar un cliente a partir de su DNI, e) mostrar un cliente dada su posición.

Ejercicio 4 [20 pts] Explique a) ¿Qué entiende por programación genérica? b) Al instanciar una plantilla, ¿cuándo se puede omitir el tipo de dato, y cuándo debe explicitarse? Dé ejemplos. c) ¿Qué es un iterator? d) ¿Es correcto escribir un arreglo `int V[10]` en un archivo binario B con la sentencia `"B.write(reinterpret_cast<char*>(V),10*sizeof(int));"`? ¿Y si V fuese un `"vector<int> V"`, sería correcto utilizar `"B.write(reinterpret_cast<char*>(&V),V.size()*sizeof(int));"`? e) ¿Cuáles son las ventajas y desventajas de utilizar archivos binarios frente a utilizar archivos de texto?

## Recuperatorio 2do parcial 2017

Ejercicio 1 (25 pts) Un archivo de código fuente es en realidad un archivo de texto. Escriba una función “ListarIncludes” que reciba el nombre de un archivo de código c++ y retorne una lista (std::list) de strings con todos los includes del mismo (incluyendo < y >, o las comillas; pero sin la palabra #include). Por ejemplo, una lista resultante podría contener: <iostream>, <fstream> <vector>, “utils.h” y “MiClase.h”. Para simplificar, considere que las líneas de los includes siempre comienzan con #include, luego puede o no haber un espacio entre medio, y finalmente el nombre del archivo, que puede estar entre <> o entre comillas.

Ejercicio 2 (30 pts) Implemente una clase AdminSeries para almacenar la lista de series que ya vio y que está viendo actualmente un usuario de un sistema de streaming. Por cada serie, debe almacenar el nombre y el número de capítulos ya visualizados. La clase debe tener: a) un método Cargar que reciba el nombre de un archivo binario y lea y cargue en memoria los datos de las series que el usuario está viendo actualmente. b) Un método RegistrarEpisodio que reciba un nombre de serie, la busque entre los datos del usuario e incremente en 1 su cantidad de capítulo visualizados; o si la serie no estaba, la agregue y setee la cantidad en 1. c) Un método EpisodiosVisualizados que reciba el nombre de una serie y retorne el contador actual de la misma (o 0 si no está en la lista). d) Un método Guardar que guarde los cambios producidos por b) y c) sobrescribiendo el mismo archivo binario indicado en a).

Ejercicio 3 (25 pts) a) Escriba un función genérica del \_maxmin que reciba un vector de elementos del tipo genérico y elimine del mismo tanto el máximo elemento como el mínimo. Si alguno de estos valores a eliminar está repetido, debe eliminarlo todas las veces que aparezca. b) Implemente además un programa cliente que genere dos vectores, uno de enteros y otro de strings, y pruebe la función con ambos, mostrando en pantalla los vectores antes y después de invocar a la función.

Ejercicio 4 (20 pts) Explique: a) ¿Cómo determina el compilador con qué tipo en particular debe especializar un template de clase? ¿Y uno de función? b) ¿Qué es un cstring (cadena estilo C)? c) ¿Qué es un iterador? ¿Qué diferencia hay entre un iterador de lista y uno de vector? d) ¿Es correcto escribir un arreglo “int V[10]” en un archivo binario B con la sentencia “B.write(reinterpret\_cast<char\*>(V),10\*sizeof(int));”? ¿Y si V fuese un “vector<int> V”, sería correcto utilizar “B.write(reinterpret\_cast<char\*>(&V),V.size()\*sizeof(int));”?

## 2do parcial 2016 – Tema A

Ejercicio 1 [30 pts] Escriba una función genérica (template) llamada ampliar() que reciba como argumentos un objeto llamado archname de tipo string y un arreglo de la clase vector (STL) llamado x que contiene structs de 2 miembros: valor (genérico) y pos (char). El string archname corresponde al nombre de un archivo binario. La función debe agregar al archivo el dato (valor) de cada elemento del arreglo x. Los agregados se hacen al principio del archivo si pos contiene 'f' y al final del archivo si pos contiene 'b'.

Ejercicio 2 [30 pts] Se dispone de un archivo de textos donde cada línea contiene 2 strings con nombres y apellidos de personas, pero en algunos casos se ubica primero el nombre y en otros el apellido, aunque el apellido siempre está en mayúsculas (todas sus letras. no así el nombre). Hay un solo nombre por persona y no hay apellidos dobles ni compuestos. Escriba una clase que posea: a) un constructor que reciba el nombre de un archivo de texto y almacene las líneas del archivo en un vector de la STL.: primero el apellido y luego el nombre; b) Un método ordenar () para ordenar la lista por apellidos y a su vez eliminar personas repetidas (idéntico apellido y nombre); c) un método guardar() que no reciba ningún parámetro y sobrescriba el archivo original con los nombres modificados.

Ejercicio 3 [30 pts] Un archivo binario contiene una base de datos de registros de tipo ficha como el que se indica en el recuadro generada por un programa C++. Cuando un usuario borra un registro desde el programa, marca el registro como borrado colocando true en el campo borrado, pero no lo quita del archivo. El archivo es ahora demasiado extenso. Escriba una clase con algún método que acceda al archivo y lo 'limpie' eliminando definitivamente todos los registros con miembro borrado en true. Use la clase en un programa C++.

```
struct ficha {  
    bool borrado;  
    string dni, ape, nom;  
    float h;  
};
```

Ejercicio 4 [10 pts] a) Puede abrir un archivo binario en modo texto? y viceversa? b) ¿Es posible escribir un objeto de tipo std::string en un archivo binario? Explique su respuesta. c) Explique las diferencias entre archivos de texto y archivos binarios. Por qué usaría list o vector en un programa? Explique. d) ¿Qué significa instanciar una plantilla o template?

## 2do parcial 2016 – Tema B

Ejercicio 1 [30 pts] En un programa C++ se define la siguiente instancia: `GestionaSecuencia <int> x;`. Escriba la clase llamada `GestionaSecuencia` que posea como atributo una secuencia dinámica de la STL y su longitud. La clase debe poseer los siguientes métodos para: a) construir la secuencia; b) incorporar datos a la secuencia desde un archivo binario; c) buscar y eliminar un elemento de la secuencia (se pasa el dato a eliminar como parámetro), d) ordenar la secuencia en forma creciente a partir de la posición de un elemento que se pasa como parámetro (entero); e) informar los elementos de la secuencia que están dentro de un rango (parámetros: 2 enteros que definen el rango). Elija la secuencia que crea más eficiente para las operaciones (list, vector, deque).

Ejercicio 2 [30 pts] En un archivo de texto llamado `PROSA.TXT` se almacena un texto con una introducción a la tecnología informática. Se desea mediante un programa C++, reemplazar en el archivo la palabra “software” por “programa”. Considere que “software” puede aparecer en una línea más de una vez. Informe la cantidad de palabras reemplazadas. El archivo `PROSA.TXT` debe quedar modificado con los reemplazos.

Ejercicio 3 [30 pts] . Existe un archivo binario `CURSO.DAT` con los datos de un grupo de estudiantes que realizaron un curso. Los datos fueron almacenados empleando una acción como la siguiente: `archi.write((char *) &x, sizeof(x));` donde `x` es una variable struct cuyos miembros son: `dni`, `nota1`, `nota2`, `nota3`, `Apellido`, `Nombres`. Los registros se hallan almacenados ordenados en forma creciente por `dni`. Escriba el método `agregar_en_orden()` de la clase `Curso` que acceda al archivo e inserte un nuevo registro (struct) manteniendo el orden por `DNI`. No utilice arreglos ni secuencias de la STL. Codifique solo el método para insertar.

Ejercicio 4 [10 pts] a) Puede abrir un archivo de texto como binario? Si responde sí, para qué serviría? b) la siguiente función genérica tiene el prototipo: `bool compara intercambia(T &x, T &y);` y determina qué dato es mayor e intercambia ambos parámetros; es posible pasarle 2 objetos de una clase? explique; c) Explique la clase contenedora `map` de STL y ejemplifique; d) Puede escribir una programa C++ que acceda a un archivo de textos que solo sabe su nombre y que no sabe cómo están dispuestos los datos? explique.

## Recuperatorio 2do parcial 2016

Ejercicio 1 (30 pts) Se desea calibrar un termómetro digital, para lo cual se realizaron 50 lecturas con 2 decimales de precisión, y que se almacenaron en un archivo de texto, una lectura por línea. Escriba un programa C++ que procese la lista de datos registrada realizando lo siguiente: a) Ordenar en forma ascendente los valores leídos. b) Obtener la temperatura promedio c) Obtener el mayor y el menor valor de todos los datos del archivo d) Guardar en un archivo binario cada valor medido y su diferencia con el promedio.

Ejercicio 2 (30 pts) Escriba una clase llamada ManejadorMultas la cual se encargará de administrar un archivo de multas de tránsito. Las multas son almacenadas en un solo archivo binario llamado “multas.dat”, donde cada multa será una instancia del struct Multa del recuadro. Deberá implementar, a) un método condonar que reciba un código de multa y establezca a cero todos los montos de multas que tienen ese código de infracción. b) Un método getDeuda que reciba un DNI y devuelva la suma de todas las multas que posee esa persona. c) Por último, un método llamado obtenerInfracciones que reciba un código de infracción y que devuelva en un vector de la STL todas las multas que poseen ese código de infracción.

```
struct Multa {  
    int dni;  
    int codigoInfraccion;  
    float monto;  
};
```

Ejercicio 3 (30 pts) Diseñe e implemente una clase templatizada llamada Arreglo que permita representar un arreglo de datos de tipo simple (enteros, punto flotante o carácter). Implemente los métodos que considere necesarios para: a) inicialización o carga de datos. b) Mostrar los elementos en pantalla. c) devolver el valor que se encuentra en una posición que se pasa por parámetro, d) Dado un valor pasado por parámetro, indicar en qué posición se encuentra en el arreglo

Ejercicio 4 (10 pts) Responda: a) ¿Qué es una plantilla? ¿Qué significa instanciar una? b) Explique cómo es posible ordenar un std::vector de structs según alguno de sus miembros. c) ¿Es posible almacenar objetos de tipo string en un archivo binario? Explique. d) ¿Qué es un iterador y para qué sirve?

## 2do parcial 2015 – Tema A

Ejercicio 1 (25 pts) Se desea desarrollar una función llamada `preproceso` que analizará datos de un acelerómetro, siendo los datos del sensor son de tipo entero. a) Leer de un archivo llamado “`acelerometro.txt`” una serie de datos y cargarlos en alguna secuencia de la STL (elija la que considere optima para este caso). b) Se sabe que la frecuencia con la que se obtienen los datos del sensor no es suficiente, por lo cual se debe insertar entre cada par de datos adyacentes su promedio, por ejemplo si se tiene los valores [6, 8, 4] se obtendrá [6, 7, 8, 6, 4] (redondear los resultados). c) Elimine todos valores duplicados de la secuencia d) A continuación, deberá buscar el máximo valor de la secuencia y moverlo al principio de la estructura. e) Guarde la secuencia modificada en un archivo llamado “`procesado.txt`”. . Para los puntos b, c y d solo DEBE USAR ALGORITMOS DE LA STL.

Ejercicio 2 (30 pts) Escriba una clase llamada `ManejadorSaves` la cual se encargará de administrar archivos con el progreso del usuario en un videojuego. Todos los saves se almacenarán en un solo archivo binario llamado “`saves.dat`”, donde cada uno será una instancia del struct `Save`. Deberá implementar, a) un método `actualizar` que reciba un objeto `Save` y que busque en el archivo para reemplazar la instancia vieja por la nueva que se recibió. La búsqueda para comparar si es el mismo save se debe realizar por el campo `codigoPlayer`. b) Un método llamado `getMayorPuntaje`, que devuelva el `Save` que tenga el mayor puntaje (si hubiera más de uno, devolver el último), comparando por el campo `puntos`. c) Por último, un método llamado `nuevoSave` que reciba una instancia de `Save` y que guarde la misma en el archivo. El archivo está ordenado de forma ascendente por el campo `ultimoNivel`, por lo que debe insertar el objeto nuevo en la posición correcta para que el archivo siga estando ordenado y no se pierda ningún objeto `Save`.

```
struct Save
{
    int codigoPlayer;
    int ultimoNivel;
    int puntos;
};
```

Ejercicio 3 (30 pts) Diseñe una clase templizada `Conjunto` que permita contener cualquier tipo como un conjunto. Implemente: a) una sobrecarga para el operador `+` que reciba otro `Conjunto` y que genere un tercer `Conjunto` que contenga los elementos de ambos conjuntos sin repetidos (ej: si el tipo de conjunto fuera entero, y tendrían {1,3,5} y {2,3,9}, el resultado debe contener {1,2,3,5,9}). b) Una sobrecarga del operador `-` que reciba otro `Conjunto` y genere un tercer `Conjunto` con los elementos del primero que no están en el segundo (ej: para los mismos operandos del ejemplo anterior, el resultado debe contener {1,5}). c) Una sobrecarga del operador `==` que permita verificar si dos conjuntos son iguales, es decir si tienen exactamente los mismos elementos. Nota: en un conjunto no importa el orden de los elementos, por lo que su clase puede ordenarlos como más le convenga según su implementación.

Ejercicio 4 (15 pts) Explique: a) Porqué la secuencia `list` tiene su propio método `sort` (ya que el algoritmo `sort` no está disponible para listas) b) ¿A qué se denomina “instanciación” o “especialización” de una plantilla? c) ¿Se puede almacenar cualquier objeto (de cualquier tipo) en un archivo binario? d) ¿Cuáles son las diferencias entre `list<T>::iterator` y `vector<T>::iterator`?

## 2do parcial 2015 – Tema B

Ejercicio 1 (25 pts) Se desea desarrollar una función que procese los puntajes de un videojuego (almacenados como enteros). La función se denomina `analizarPuntos` y debe: a) Leer de un archivo llamado "puntos.txt" los puntos del videojuego y cargarlos en una lista de la STL. b) Por un error en el juego se almacenó -3 en lugar de 10, por lo que se debe recorrer y reemplazar todos los -3 por 10, por ejemplo si los puntos son [6, -3, 4] se obtendrá [6, 10, 4]. c) Busque el puntaje mínimo de la lista e inserte una copia al principio, y el máximo en insertelo al final de la lista. d) Ordene la lista. Responda ¿Podría haberse ordenado la mitad de la lista? ¿Justifique por el sí o por el no? e) Guarde la lista modificada en un archivo llamado "analizado.txt". Para los puntos b, c y d solo DEBE USAR ALGORITMOS DE LA STL.

Ejercicio 2 (30 pts) Escriba una clase llamada `ManejadorMultas` la cual se encargará de administrar un archivo de multas de tránsito. Las multas son almacenadas en un solo archivo binario llamado "multas.dat", donde cada multa será una instancia del struct `Multa`. Deberá implementar, a) un método `condonar` que reciba un código de multa y establezca a cero todos los montos de multas que tienen ese código de infracción. b) Un método `getDeuda` que reciba un DNI y devuelva la suma de todas las multas que posee esa persona. c) Por último, un método llamado `obtenerInfracciones` que reciba un código de infracción y que devuelva en un vector de la STL todas las multas que poseen ese código de infracción.

```
struct Multa
{
    int dni;
    int codigoInfraccion;
    float monto;
};
```

Ejercicio 3 (30 pts) Implemente dos funciones templatizadas. a) Primero la función `leer` la cual recibe el nombre de un archivo de texto y almacena su contenido en un vector de la STL. Se puede asumir que el tipo de dato se puede leer con el operador `>>`. b) Otro método llamado `guardar` que recibe un vector de la STL de tipo genérico y almacena su contenido en un archivo de texto, ingresando en cada línea un registro diferente. Se puede asumir que el tipo de dato puede ser escrito usando el operador `<<`. d) Implemente una función `main` donde se cree un vector de flotantes de 100 elementos, generados aleatoriamente en el rango [0,1], y lo almacene con el método `guardar`, y luego lo vuelva a leer con el método `leer` y muestre su resultado en la consola.

Ejercicio 4 (15 pts) a) Indique 3 formas de acceder al último elemento de un vector dinámico de la STL, señalando las ventajas de cada una. b) ¿Puede almacenarse un objeto `string` en un archivo? Justifique su respuesta. c) Si se necesitara almacenar el valor numérico 13789 en un archivo, ¿de qué manera ocuparía menos espacio? ¿guardando los datos como texto o como binarios?. Explique por qué. d) Defina y explique los 3 componentes principales de STL: contenedores, iteradores y algoritmos.



## Recuperatorio 2do parcial 2015

Ejercicio 1 (25 pts) Se tienen dos archivos binarios. El primero almacena datos de un conjunto de candidatos a presidentes del CPP (Consejo Provincial de Programadores), utilizando un struct Candidato como el del recuadro. En este archivo, hay una candidato por cada partido, y todas las cantidades de votos se encuentran en 0. El segundo archivo almacena los votos de todas las mesas escrutadas mediante el struct Voto. Escriba una función RecuentoDefinitivo(...) que reciba los nombres de ambos archivos, y actualice la información de cada candidato (el campo cant\_votos), según la cantidad de votos de su partido. La función debe retornar el dni del candidato ganador.

```
struct Candidato {  
    int dni,cod_partido,cant_votos;  
};  
struct Voto {  
    int nro_mesa,cod_partido;  
};
```

Ejercicio 2 (25 pts) Escriba una función genérica vector\_a\_list(...) que reciba un std::vector y retorne un std::list que contenga la misma secuencia de elementos que el std::vector de entrada, para cualquier tipo de elemento. Escriba un programa cliente que genere un vector con 10 enteros aleatorios, utilice la función para pasarlos a una lista, y muestre la lista en pantalla.

Ejercicio 3 (25 pts) Se tiene un archivo como el del recuadro, donde se encuentran datos de atletas que compiten en una carrera de 100m llanos (un atleta por línea, primero su mejor registro, y luego su nombre completo). Se desea organizar las series de forma tal que no se agrupen los mejores ni los peores en una misma serie. Para ello, se propone el siguiente algoritmo: 1) ordenar la lista de competidores según sus tiempos, de menor a mayor, 2) determinar cuántas series (N) se deben organizar en base a la cantidad de competidores (pueden correr hasta 8 competidores por serie), y 3) asignar el 1er competidor de la lista a la 1er serie, el 2do competidor a la 2da lista, el N-ésimo competidor a la N-ésima serie, el N+1 nuevamente a la 1er serie, el N+2 nuevamente a la 2da, y así hasta asignarlos todos. Escriba una función generar\_series(...) que reciba un nombre de archivo. La función debe leer los datos de los competidores desde ese archivo y generar N archivos serie1.txt, serie2.txt, ..., serieN.txt, cada uno conteniendo la lista de competidores (solo nombres, ordenados alfabéticamente) de cada una de las series. La función debe retornar el número de series generadas.

9.86 Boldon Ato

9.58 Bolt usain

9.74 Gatlin Justin

9.69 Gay Tyson

9.79 Greene Maurice

9.86 Lewis Carl

9.72 Powell Asafa

Ejercicio 4 (25 pts) Explique a) ¿Qué entiende por programación genérica? b) Al instanciar una plantilla, ¿cuándo se puede omitir el tipo de dato, y cuándo debe explicitarse? Dé ejemplos. c) ¿Qué es un iterator? ¿Por qué las funciones de la biblioteca <algorithm> reciben iterators en lugar de contenedores? d) ¿Es correcto escribir un arreglo “int V[10]” en un archivo binario B con la sentencia “B.write(reinterpret\_cast<char\*>(V),10\*sizeof(int));”? ¿Y si V fuese un “vector<int> V”, sería correcto utilizar “B.write(reinterpret\_cast<char\*>(&V),V.size()\*sizeof(int));”? e) ¿Cuáles son las ventajas y desventajas de utilizar archivos binarios frente a utilizar archivos de texto?

