

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJES DE PROGRAMACIÓN 1

3ra. práctica (tipo b)
(Primer Semestre 2018)

Indicaciones Generales:

- Duración: 110 minutos.
- Se podrá usar como material de consulta solo sus apuntes de clase.
- No se pueden emplear variables globales, cadenas de caracteres (ni como arreglos ni como punteros), objetos (con excepción de cin, cout y objetos de iostream), tampoco las bibliotecas cstdio, stdio.h o similares.
- Toda función propia deberá ser implementada obligatoriamente en archivo diferentes al main.cpp, debiendo crear los archivos .h y .cpp correspondientes y estas deben ser agrupadas por temas.
- Se tomará en cuenta en la calificación el uso de comentarios relevantes.
- Los programas que presenten errores de sintaxis o de concepto se calificarán en base al 40% del puntaje de la pregunta. Los que no den resultados coherentes en base al 60%.
- La presentación, la ortografía y la gramática de los trabajos influirá en la calificación.

Puntaje total: 20 puntos

Cuestionario:

Pregunta 1 (10 puntos)

Muchas veces, para probar algún algoritmo o para hacer alguna simulación se requiere contar con conjuntos de un número considerable de datos y muchas veces no se cuenta con esto datos. Por esta razón, los desarrolladores algunas veces los generan de manera aleatoria, ya sea de manera directa o componiendo datos a partir de otros que ya se tienen.

Este laboratorio busca que usted desarrolle una **biblioteca estática de funciones**, que permita generar grupos de datos de manera aleatoria ya sea directamente como también componiendo datos a partir de otros grupos de datos ya existentes. La biblioteca se denominará **GeneradorDeDatosAleatorios**.

Para lograr esto se debe saber que en la biblioteca **cstdlib** existen dos funciones que nos pueden ayudar en esta labor: **rand** y **srand**. La función **rand** no tiene argumentos y devuelve un valor pseudo-aleatorio entero entre 0 y **RAND_MAX**, donde **RAND_MAX** es una constante simbólica también definida en **cstdlib** y vale por lo general 32,767. La función **srand** no devuelve valores pero si recibe un valor entero llamado semilla que permite cambiar la secuencia de valores que entregará la función **rand**. La función **srand** se coloca una sola vez en el programa y se emplea de la siguiente manera: **srand (time (NULL))**, donde **time** es una función definida en la biblioteca **ctime**, que devuelve el número de segundos transcurridos desde las 00:00:00 horas de una fecha determinada.

La biblioteca deseada debe tener las siguientes funciones:

- a) **aleatorioRango(inicio, fin)**: devuelve un valor aleatorio entre **inicio** y **fin**. La función deberá verificar que **inicio** sea menor que **fin**, de no ser así deberá intercambiarlos. Para calcular ese valor obtenga el resto de la división entre **rand** y **(fin-inicio+1)** y a ese resultado súmele **inicio**. La función debe ser implementada como una plantilla para poder ser empleada tanto con valores de tipo **int** como **char**. La función devuelve el mismo tipo de dato que los argumentos.
- b) **valoresAleatorios(valores, numDatos, inicio, fin)**: esta función llena el arreglo **valores** con tantos datos como indica **numDatos**, los datos asignados deben ser valores aleatorios entre **inicio** y **fin**. La

función debe ser implementada como una plantilla para poder ser empleada tanto con valores de tipo *int* como *char*. La función no devuelve resultados.

- c) **valoresAleatorios(valores, numDatos, v1, nd1)**: esta función llena el arreglo **valores** con tantos datos como indica **numDatos**, los valores deben ser tomados de manera aleatorio del arreglo **v1**, por lo que el arreglo **valores** solo puede tener valores que estén en **v1**, pudiendo haber muchos repetidos.

Por ejemplo:

v1:	nd1 = 4		valores:	numDatos=9
1		==>	3	
3			5	
5			1	
7			1	
			5	
			7	
			3	
			7	
			7	

La función debe ser implementada como una plantilla para poder ser empleada tanto con valores de tipo *int*, *double* y *char*. La función no devuelve resultados.

valoresAleatorios(valores, numDatos, v1, nd1, v2, nd2): esta función hará algo similar a la función anterior pero los valores asignados al arreglo **valores** serán una composición de los dato de los arreglos **v1** y **v2**. El ejemplo siguiente muestra cómo:

v1:	nd1 = 6	v2:	nd2 = 5	valores:	numDatos=7
2222		3		1214007	
4444		11		2222135	
6666		135		2222015	
8888		7		8888015	
1046		15		1214007	
1214				4444135	
				6666011	

Esta función solo será implementada para valores enteros.

Consideraciones:

- Las funciones no leerán ni imprimirán datos, por lo que la información deberá ingresar a las funciones por los parámetros.
- La pregunta debe contemplar la elaboración de: un proyecto de implementación y prueba de las funciones, un proyecto que genere la biblioteca estática y un proyecto donde se pruebe las funciones en la biblioteca ya compilada. La prueba de las funciones debe ser hecha lo más simple posible pero que muestre claramente los resultados correctos.

Pregunta 2 (10 puntos)

Se desea hacer una simulación con datos de alumnos de una Universidad, como no se tienen los datos se tendrá que generar los datos de manera aleatoria y guardarlos en un archivo de textos, deben generarse entre 150 y 300 juegos de datos. El archivo debe similar al siguiente:

No colocar estos datos, solo es → referencial	Código del alumno	Número de créditos	Pago por crédito	Codigo2	Hora de reserva
	2018011234	18	974.50	LaQ54312	09:32:45
	2010146379	22	602.75	GyW88409	10:05:24

El código del alumno está formado por tres elementos, el año de ingreso, la modalidad de ingreso y el código propiamente dicho. Los años serán tomados de una lista colocada en un archivo de textos como se muestra al final de la pregunta. La modalidad de ingreso también será tomada del archivo. El código debe ser un número aleatorio de cuatro cifras.

El número de créditos debe estar entre un mínimo y un máximo. Estos valores también estarán en el archivo de textos.

El pago por crédito se ajusta a las escalas de pago, que aparecen también en el archivo.

El Código2 está formado por tres caracteres y un número de cinco cifras. El primer y el tercer carácter serán una letra mayúscula, la segunda minúscula. El número será un valor entero de cinco cifras. Como no se pueden usar cadenas de caracteres en este laboratorio, los elementos del código serán determinados por separado, y a la hora de emitir el reporte se imprimirán juntos.

La hora de reserva debe darse, como se ve, en horas, minutos y segundos. La forma cómo debe asignar estas horas será como sigue: el primer elemento debe ser una hora (hora, minutos y segundos) entre las 6:00:00 y las 10:00:00, las siguientes serán dadas secuencialmente con lapsos de tiempo dados aleatoriamente entre 15 y 25 minutos (minutos y segundos). Si se llegase a una hora después de las 22:30:00 la hora será reemplazada por las 6 y se seguirá asignado el resto a partir de esa hora.

Se pide desarrollar un proyecto denominado **Pregunta02** en el que, usando la biblioteca estática **GeneradorDeDatosAleatorios** desarrollada en la primera pregunta, lea los parámetros de un archivo similar al que se muestra a continuación y emita un reporte donde se coloquen los datos de los alumnos generados.

2000	2003	2004	2005	2009	2010	2015	...	← Años a tomar en cuenta
1	14	17	2	22	5	51	7 77...	← Modalidad de ingreso
3	24							← Mínimo y Máximo número de créditos
974.50	897.35	805.24	...					← Pago por créditos

Luego de generar los datos deberá guardarlos en el reporte ordenados por el código del alumno. El algoritmo para ordenarlos deberá hacerse mediante QuickSort u otro similar con eficiencia: $n \log n$.

Los datos leídos del archivo con los parámetros (según sea el caso) y los datos generados deben ser colocados en arreglos estáticos.

En la evaluación se tomará en cuenta el formato del reporte, no se podrá emplear el caracter de tabulación ('\t') para la emisión del reporte.

NOTAS:

- Cree en el computador una carpeta de trabajo con la siguiente ruta: c:\temp\Laboratorio3. En ella colocará los proyectos que den solución a las preguntas planteadas.
- En cada archivo que implemente en los proyectos (.h y .cpp) deberá colocar un comentario en el que coloque claramente su nombre y código, de no hacerlo se le descontará 0.5 puntos por archivo.

Al finalizar la práctica, comprima¹ la carpeta **laboratorio3** en un archivo con nombre <código del alumno con 8 dígitos>.<extensión del archivo comprimido> y súbalo a la Intranet del curso, en el enlace Documentos, en la carpeta \Laboratorio3\<código del horario>\<aula>.

Profesores del curso: Arturo Oncevay
Miguel Guanira

San Miguel, 21 de septiembre del 2018.

¹ Para evitar problemas en la corrección de la prueba, utilice el programa de compresión que viene por defecto en Windows (Zip).