



UNIVERSIDAD DE
SAN BUENAVENTURA
CALI

Universidad de San Buenaventura Cali

Facultad de Ingeniería

Departamento de Ciencias y Tecnologías de la Información

Laboratorio de Software I

Parcial Primer Corte

Objetivos Pedagógicos

El objetivo de este parcial es comprobar que el estudiante haya adquirido las competencias que hasta este momento se esperan de una activa participación en el curso y realización de los talleres.

Instrucciones

Este examen supletorio está diseñado para trabarse de manera individual.

Se debe presentar desde el salón de clase y firmando el control de asistencia. Durante la prueba se pueden consultar los apuntes propios en el cuaderno físico, el sitio web de referencia oficial del curso <https://cplusplus.com/>, y los Replit de elaboración propia de los integrantes de la pareja de trabajo. Se usará un único computador (portátil o de la sala de cómputo) por cada equipo de trabajo. Por otra parte, el consultar foros, en busca de código que pueda ser copiado y pegado en el examen, el uso de dispositivos de almacenamiento externo, celulares, servicios de mensajería, correo electrónico, de desarrollo de código de manera colaborativa por fuera del equipo de trabajo, está totalmente prohibido, se considerará como intento de fraude y se penalizará drásticamente, y en concordancia con el reglamento estudiantil.

La fecha y plazo de entrega es jueves 08 de Septiembre a las 09:05 horas. No se recibirán entregas por correo electrónico, ni después de ese plazo.

Se debe enviar, en un archivo comprimido de extensión .zip, el código fuente que da solución al problema (i.e., main.zip), los archivos de prueba que apliquen para el caso, y un reporte en PDF con la captura de pantalla de la ejecución del código, de acuerdo con las especificaciones. En su reporte debe indicar cuáles desafíos resolvió, y adjuntar captura de pantalla del código que lo resuelve listando la función o funciones que diseñó e implementó para resolver cada uno de ellos, junto a evidencias de ejecución exitosa. En el reporte se debe incluir la dirección del Replit empleado, si el código se desarrolló empleando dicha plataforma. Tanto el archivo .PDF, como el de código fuente deben indicar los nombres de los integrantes del equipo (i.e. nombre completo, código, correo electrónico). No se deben enviar ejecutables, ni archivos pertenecientes a la configuración de Replit, ni de ningún otro entorno de desarrollo.

El lenguaje de codificación puede ser cualquiera de los aceptados en el curso (i.e., C, C++, Python, Java, Kotlin, Go).

Cuestionario

1. Problema Uno – fiboArreglo, 15 puntos

La serie de Fibonacci es una de las series más famosas de la matemática. Los dos primeros valores de esta serie, para los términos 0 y 1, son respectivamente, 0 y 1. Esto es, $\text{Fibonacci}(0)=0$ y $\text{Fibonacci}(1)=1$. El valor de la serie, para cualquier otro término se define como la suma de los valores de los dos términos inmediatamente anteriores. Esto es, $\text{Fibonacci}(n)=\text{Fibonacci}(n-1)+\text{Fibonacci}(n-2)$. Así pues, por ejemplo, $\text{Fibonacci}(2)$, será igual a la suma de $\text{Fibonacci}(1)$ con $\text{Fibonacci}(0)$, que es igual a 1. Los diez primeros valores de la serie de Fibonacci, para sus respectivos términos, se ilustran en la Tabla 1, a continuación.

Tabla 1. Diez primeros términos de la serie Fibonacci.

n	Fibonacci(n)
0	0
1	1
2	1
3	2
4	3
5	5
6	8
7	13
8	21
9	34

Desafío: Diseñe e implemente una función llamada `fiboArreglo`, que calcule mediante una instrucción de ciclo **while** o **do-while**, y almacene en un arreglo, los valores de los primeros 50 (cincuenta) términos de la serie de Fibonacci.

La función debe ser de tipo *void*, y puede, bien sea recibir el arreglo como un parámetro, o conocer la dirección del arreglo debido a que este está declarado de manera global.

Restricciones: Desviaciones de los lineamientos de diseño se penalizarán descontando hasta 15 puntos.

2. Problema Dos – fiboVerificador, 35 puntos

Dado un archivo de texto de entrada (“input.txt”), conteniendo números enteros positivos en el rango [0, 7778742049], se debe determinar si cada uno de los valores presentes en el archivo corresponde a un valor de los primeros 50 términos de la serie de Fibonacci, y cuál es el término que lo genera. Esto es, por cada valor presente en el archivo, que pertenezca a los primeros 50 términos de la serie, la serie, se debe agregar a un archivo de salida (“output.txt”), el valor mismo, la palabra *true* y el término que lo genera. Por ejemplo, si el archivo tiene el valor 21, la salida sería 21 true 8. Si el archivo tiene el valor 8, la salida sería 8 true 6. En el caso especial que el valor sea 1, serán válidas cualquiera de sus dos posiciones. Por lo contrario, si el archivo tiene un valor que NO pertenece a los primeros valores de la 50 serie, se debe agregar al mismo archivo de

salida el valor y la palabra *false*. Por ejemplo, si el archivo tiene el valor 4, la salida sería 4 false. Los nombres de los archivos de entrada y de salida serán “input.txt” y “output.txt”, respectivamente. La estructura del archivo “input.txt” incluye en la primea línea, la cantidad de números que deben ser verificados, seguido por la misma cantidad de valores, cada uno en una línea del archivo. El archivo “salida.txt” contiene una línea por cada valor analizado. Un ejemplo de entradas y de salidas se ilustra en la Tabla 2.

Desafío: Diseñe y desarrolle un programa que, apoyado en la función del Problema-Uno, calcule y genere las salidas para el archivo de entrada, acorde con lo anteriormente descrito.

Restricciones: Desviaciones de los lineamientos de diseño se penalizarán descontando hasta 35 puntos.

Tabla 2. Ejemplo de archivos de entrada y de salida, respectivamente.

entrada.txt	salida.txt
20	500085958 false
500085958	5 true 5
5	24157817 true 37
24157817	0 true 0
0	89 true 11
89	0 true 0
0	454295286 false
454295286	1768964533 false
1768964533	864730169 false
864730169	317811 true 28
317811	963747242 false
963747242	55 true 10
55	801148510 false
801148510	1470635321 false
1470635321	1584845502 false
1584845502	390538214 false
390538214	2178309 true 32
2178309	1059681908 false
1059681908	160427405 false
160427405	2178309 true 32
2178309	