



Cátedra Tecnología de Sistemas

[Introducción a la programación]

Código: [00831]

Proyecto. Valor 3%

Temas de Estudio

1. Tema 1 y Tema 2 vistos anteriormente.
2. Tema 4 Entrada y salida de datos e información

Subtemas

- a. Capítulo VI - Entrada /Salida de flujo en C++
- b. Capítulo VII - Procesamiento de Archivos
- c. Capítulo VIII - Manejo de excepciones

Objetivo

Resolver un problema, con un programa en el lenguaje de C++ aplicando lo aprendido en los temas de estudio

Software de Desarrollo

CodeBlocks, en la plataforma MOODLE está disponible las instrucciones para su instalación

Desarrollo

Para completar la **QUINIELA**, Juan desea que las estadísticas se almacenen en archivos, para cada grupo y para el rating de selecciones de futbol.

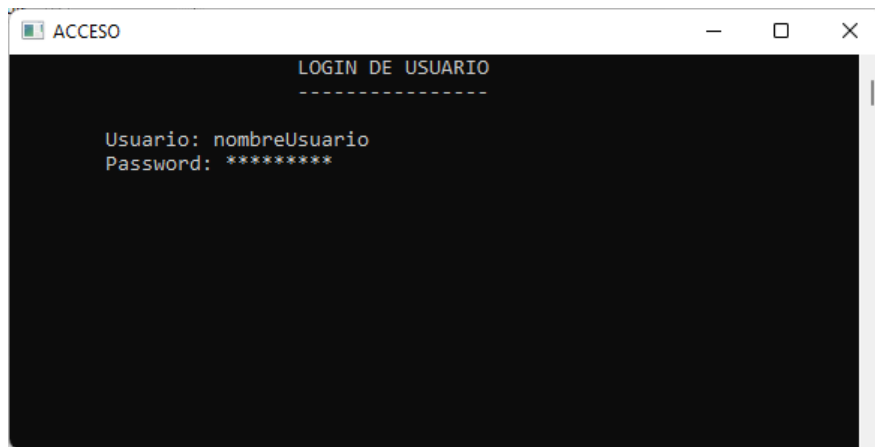
Instrucciones:

1. Agregar una ventana de seguridad para ingresar a la quiniela. Esta ventana de seguridad debe recibir el nombre del usuario y contraseña (encriptada con "*" asteriscos).
2. El nombre del usuario y la contraseña deben ser leídos de un archivo txt, creado específicamente para tal propósito y con los datos cargados manualmente al abrir el editor de texto.
3. En la tarea 1 y 2, se incluyeron dos grupos, debe asegurarse que estén creados o completar lo instruido en dichos enunciados.
4. Se mantienen las instrucciones para cada grupo, partidos y reglas generales.
5. Crear un archivo de tipo txt para cada grupo, total dos grupos

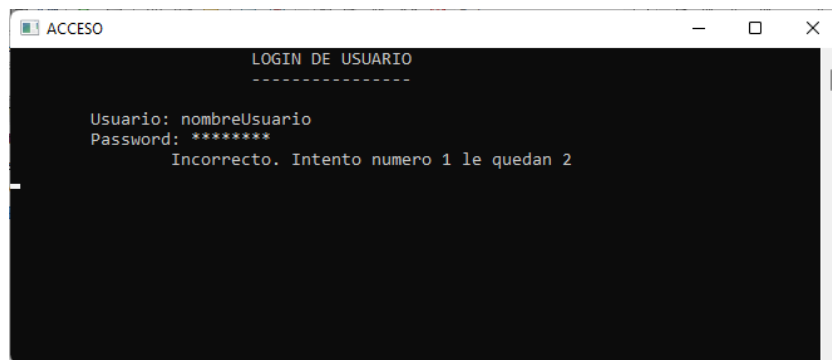
6. Crear un archivo de tipo txt para el raiting de todas las selecciones de futbol, ordenadas de mayor a menor, conforme al puntaje obtenido.
7. Crear un reporte en pantalla para desplegar los datos contenidos en el archivo para cada grupo, estos datos son: Nombre de selección de futbol y los datos según lo solicitado en las tareas.
8. Crear un reporte en pantalla obtenido del archivo del raiting de todas las selecciones de futbol, ordenado de mayor a menor.

Codificación:

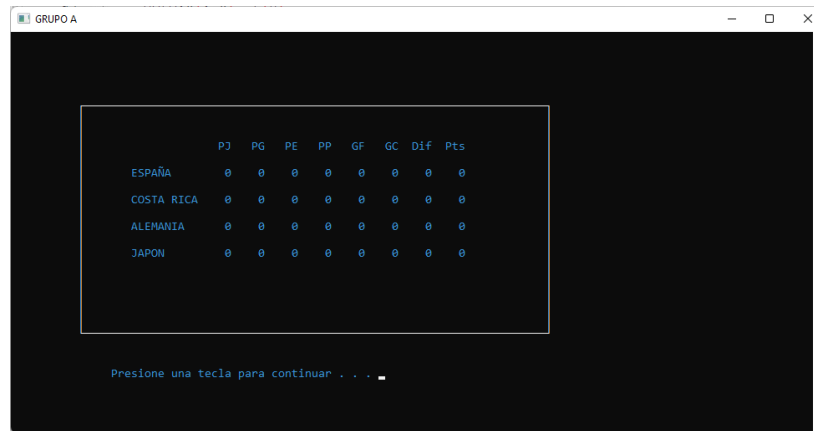
1. La ventana de seguridad debe recibir el nombre del usuario y la contraseña, obtenidos de un archivo txt, denominado "acceso".
2. Al digitar la contraseña debe mostrarse únicamente " * " asteriscos, como se muestra en el siguiente ejemplo:



3. Debe validar el usuario y contraseña, de tal manera que si es inválido muestre un mensaje y no permita entrar a la quiniela.
4. Debe programar este acceso para que tenga tres intentos, si falla, finaliza el programa.



5. Se deben crear dos grupos A y B, según lo indicado en las tareas 1 y 2, pero con los datos registrados en un archivo de texto para cada grupo, ejemplo:
 - a. grupoA.txt
 - b. grupoB.txt
6. Los datos deben ser registrados desde el programa a cada uno de esos dos archivos.
7. El reporte para cada grupo, debe mostrarse similar al siguiente ejemplo:



	PJ	PG	PE	PP	GF	GC	Dif	Pts
ESPAÑA	0	0	0	0	0	0	0	0
COSTA RICA	0	0	0	0	0	0	0	0
ALEMANIA	0	0	0	0	0	0	0	0
JAPON	0	0	0	0	0	0	0	0

Presione una tecla para continuar . . .

8. Crear un archivo para el rating de los equipos de futbol o selecciones ordenados de mayor a menor, utilizando la columna de puntaje y en caso de empate, puede utilizar la columna de diferencia de goles.
9. El reporte debe mostrar las ocho selecciones o equipos de futbol, ordenados de mayor a menor según el puntaje obtenido y mostrar el puntaje.

Honestidad Académica



<https://audiovisuales.uned.ac.cr/play/player/23048>

Nota Importante

Cada estudiante es responsable del contenido que entrega, si no es el archivo correcto, no podrá entregarlo posterior a la fecha establecida.

Si el contenido del archivo coincide con algún otro estudiante, o se comprueba que no es de su autoría, se aplicaría lo indicado en la plataforma en el documento [Lineamientos ante casos de plagio](#)

Indicaciones Importantes

- Es obligatorio que incluya todo el directorio donde se encuentra el Proyecto.
- El **Proyecto** debe estar desarrollado en **CodeBlocks** que es la herramienta oficial del curso.
- El programa debe ser modular, utilizando de la mejor manera funciones definidas por usted.
- Los trabajos deben realizarse en forma individual. Dentro del código del programa debe de indicar la documentación que explique cómo fue realizado el programa.
- Si utiliza código de algún ejemplo del libro, o de otra fuente que no sea de su autoría, debe de indicarlo.
- Comprima todos los archivos en un solo archivo .zip o .rar.
- **Nombre del archivo que envía:** debe ser nombre y primer apellido del estudiante, y nombre de la tarea. **Ejemplo: JuanRojas-tarea1.**
- La entrega del **Proyecto** en las fechas establecidas en la plataforma de aprendizaje en línea Moodle en el apartado que se indique.
- Si no concluyó a tiempo el proyecto, debe entregar lo que pudo hacer e incluir una carta explicando las razones por las cuales no finalizó.

Rúbrica de Evaluación

Criterio	Cumple a satisfacción lo indicado en la evaluación	Cumple medianamente en lo indicado en la evaluación	Cumple en contenido y formato pero los aportes no son significantes	No cumple o no presenta lo solicitado
Formato: Nitidez y presentación del código, incluyendo Redacción-Ortografía // Documentación interna dentro del código	5	3	2	1
Orden y claridad en el planteamiento (lógica). Cómo ordena las ideas para lograr la mejor solución, aplicando correctamente los conocimientos y herramientas vistos hasta el momento en el curso.	15	10	3	1
Estructuras de control – Secuenciales. Utiliza <i>if</i> , <i>if/else</i> y <i>switch</i> en la solución de forma adecuada. Ejemplo: Menús, Preguntas, validaciones de datos, evaluación de ganador, etc.	9	5	2	1
Estructuras de control - iterativas. Utiliza <i>while</i> , <i>do/while</i> y <i>for</i> en la solución de forma adecuada. Ejemplo: recorrido de arreglos o datos de archivos, validaciones, repetición del programa, etc.	9	5	2	1
Funciones General. Utiliza las necesarias y suficientes adicionales a la o las solicitadas en el enunciado. Con un correcto nombre, parámetro (si aplica) y llamado de las mismas.	9	5	2	1
Funciones Solicitadas. Creación de la ventana y función para la validación de seguridad o acceso al programa.	9	5	2	1

Archivos. Utiliza manejo de archivos en su solución, en particular para llevar control de los jugadores.	9	5	2	1
Archivos. Utiliza manejo de archivos en su solución, en particular para preguntas y respuestas.	15	10	3	1
Impresión de información en pantalla (Calidad-validez datos/presentación tabulada). Uso correcto de entrada y salida de datos por pantalla. Solicitud de información, validaciones, presentación de lo mínimo solicitado.	15	10	3	1
Interfaz de usuario en general. (NO GUI) - aplicación fácil usar e intuitiva. Se refiere a evaluar la distribución y uso de la pantalla, menús y dinámica de uso de la solución. Incluyendo lo mínimo según el ejemplo del enunciado o lo adicional que el estudiante entienda necesario.	5	3	2	1
TOTAL	100			