

ALGORITMOS Y ESTRUCTURAS DE DATOS

2023/1

PROFESOR: Manuel Alejandro Moscoso Domínguez
manuel.moscoso.d@gmail.com

Laboratorio Semana 1

En esta oportunidad realizaremos actividades que nos permitan refrescar contenidos vinculados al trabajo con distintos tipos de datos en C y C# además de revisar la gestión de un repositorio a través de GIT.

Objetivos

- Crear una cuenta en GITHUB junto con un repositorio para los ejercicios a desarrollar en el laboratorio.
- Resolver ejercicios que involucren el uso de tipo de datos básicos.
- Desarrollar algoritmos que permitan entregar una solución a los problemas entregados.

Ejercicios

Nomenclatura para nombre de archivos fuentes

El nombre del archivo en el cual se almacena el código fuente debe considerar el siguiente formato: labsemanaX_ejercicioY.EXT donde; X corresponde a la semana, la Y al número de ejercicios y EXT a la extensión del lenguaje de programación utilizado.

Ejercicio número 1

Escriba un programa en C que permita evidenciar el uso de todos los tipos de datos básicos.

Ejercicio número 2

Escriba un programa en C que reciba un número desde el teclado y aplique a lo menos 4 operaciones aritméticas para demostrar su funcionamiento.

Ejercicio número 3

Es necesaria la implementación de un prototipo que nos permita visualizar una barra de carga o esperar durante la ejecución de un juego. Como el juego en su etapa temprana de desarrollo se ejecuta desde una consola, la barra debe aparecer antes que se presente la interfaz gráfica de usuario. Para eso es necesario crear un programa en C o C# que permita dibujar en la consola la barra de carga.

Ejercicio número 4

Es necesario para partir con nuestro piloto de juego el GRAN_GATO crear en C o C# un programa que permita almacenar las jugadas ingresadas por cada usuario. En esta etapa no es necesario detectar las jugadas sino que evidenciar con datos precargados como sería el almacenamiento y presentación del tablero.

Ejercicio número 5

Para la elaboración del juego VRVTetrix es necesario generar de manera aleatoria que pieza es la que se presenta. En nuestro caso existen 7 piezas del juego y la única regla adicional es que no pueden repetirse la misma pieza si en la jugada anterior ya salió. Para poder validar esta función es necesario desarrollar en C un programa que permita esta implementación.

Ejercicio número 6

Estamos interesados en generar la base de la lógica que nos permita tener un juego de rol interactivo durante la espera o pantalla de carga. Para esto es necesario poder crear la estructura que permita que cada persona pueda reflejar su tipo, nombre, vida, poder, velocidad. Parte importante de lo que pueda hacer cada jugador en su turno es descansar para recuperar energía (vida), atacar o cargar un hechizo o poder según

corresponda. Es necesario presentar una clase que permita resolver y apoyar la implementación de esta fase inicial del juego.

Ejercicio número 7

Estamos interesados en poder trabajar en la pantalla que permita seleccionar un grupo de vehículos disponibles para ser utilizados en carreras dentro de un videojuego. Cada vehículo se define a través del fabricante, modelo, año, % de velocidad y % de maniobrabilidad. Es necesario poder presentar una clase que permita trabajar con esta información y tres ejemplos que nos permitan evidencia la usabilidad.