



ACTIVITAT 05 – ESTRUCTURES DE DADES

Ésta actividad la podéis realizar utilizando la técnica de desarrollo agile Pair Programming. Ésta técnica consiste en que dos programadores trabajen en un mismo terminal para desarrollar un proyecto. Se recomienda que el rol de codificador y navegador vaya alternándose.

Queremos crear un programa que permita realizar simulaciones de carreras entre distintos sujetos a partir de ciertas variables.

Como mínimo tendrás que desarrollar una clase Circuito, Participante, Vehículo, y una clase Motocicleta y Coche que hereden de Vehículo.

1. Las características a tener en cuenta en un Circuito son:
 - a. Nombre
 - b. Tiempo. Posibles valores: “lluvioso”, “húmedo”, “seco”.
 - c. Longitud. (podéis limitar la longitud entre un rango de valores)
2. Las características a tener en cuenta para un Participante son:
 - a. Nombre
 - b. Vehículo
 - c. Cuantas veces ha quedado 1º, 2º, 3º o fuera del podio
3. Las características a tener en cuenta para un Vehículo son:
 - a. Modelo.
 - b. Tracción: Posibles valores “blanda”, “mediana”, “dura”
 - c. Avance mínimo i máximo (podéis indicar un límite de valores)
4. Los tipos de Vehículos son:
 - a. Motocicleta:
 - i. su movimiento siempre será un número aleatorio entre la velocidad mínima y la máxima sumando 5 si la tracción es dura, y 2 si la tracción es mediana.
 - ii. debe tener implementada una función que controle si cae al suelo o no en cada avance teniendo presente que:
 1. Cuando el terreno es mojado y la tracción es dura el porcentaje es del 30%.
 2. En terreno húmedo y tracción dura así como en terreno mojado y tracción media el porcentaje es del 20%.

3. En terreno húmedo y tracción media el porcentaje es del 10%

4. En el resto de casos el porcentaje es del 5%.

iii. Si cae al suelo estará 5 movimientos sin avanzar.

b. Coche:

i. su movimiento siempre será un número aleatorio entre la velocidad mínima y la máxima sumando modificadores según la tracción i el tiempo. Por ejemplo:

	Lluvioso	Húmedo	seco
blanda	+4	+2	0
media	+2	+2	+2
dura	0	+2	+4

5. El usuario d poder crear , modificar i ver Participantes (el nombre es único) y sus estadísticas (número de veces que ha quedado 1º, 2º, 3º u otros)
6. El usuario debe poder crear distintos vehículos (el nombre de modelo es único)
7. El usuario debe poder asignar y quitar Participantes a un Circuito.
8. Cuando empiece la simulación de una carrera, cada medio segundo se deberá avanzar a cada participante un cierto valor obtenido a partir de las características de cada participante y del circuito.
9. El usuario ha de poder crear distintos circuitos y seleccionar qué carrera quiere simular.
10. Se debe mostrar el podio final de la carrear y añadir la información al participante.

A continuación se muestra una propuesta de interfaz:

Gestión de Participantes

Nombre (datalist) ▾

Vehículo ▾

Estadísticas

Guardar Participante

Cargar estadísticas

New Vehículo

Modelo (datalist) ▾

Min Vel

Max Vel

Tipo Traccion ▾

Tipo Vehículo ▾

Nuevo Vehículo

Cargar estadísticas

New Circuito

nombre

longitud

Tiempo ▾

Nueva Carrera

Asigna un participante al circuito

Quita un participante del circuito

circuitos ▾

Participants ▾

Asignar

circuitos ▾

Participants ▾

Quita

CARGA CIRCUITO

Circuito Cargado

La gran carrera

Lluvioso

150px

START!

Super Mario

Bowser

150px

FINISH!!

Comentarios: