


Programación	
2º Evaluación	
Prueba práctica	

Jorge, nuestro compañero de trabajo, nos ha dejado un problemón. Se ha puesto malo y esta tarde tenía que realizar la entrega de un proyecto.

El proyecto es para una empresa de videojuegos. Tenía que codificar la lógica del juego, la interfaz gráfica no hace falta. Nos ha mandado un video de como funcionaría el juego una vez integrado en la interfaz gráfica que está desarrollando su compañero.

<https://www.youtube.com/watch?v=s90Nfebl8FU>

Requisitos funcionales

El juego consiste en generar una serie de pares de operaciones entre las que el jugador elegirá cada turno. El objetivo es que elija siempre la operación que genere el resultado mayor, de manera que al finalizar el juego, el número obtenido (puntuación) sea lo mayor posible.

Clase Operacion

- Esta clase define UNA operación.
- Cada operación se realiza entre dos números y puede ser de 4 tipos: suma, resta, multiplicación y división entera.
- El tipo de operación se genera de manera aleatoria.
- Las operaciones se realizan entre dos números: un número aleatorio (1-99) y un número que representa la puntuación del jugador.

Clase JuegoMates

- Esta clase define el programa principal (ejecutable).
- Generará un número aleatorio (1-9) que representa la puntuación inicial del jugador.
- El juego consta de 5 turnos, en cada turno se le ofrece al jugador dos operaciones a elegir.
- La puntuación debe actualizarse a cada turno.
- Debemos llevar un registro de la puntuación máxima que era posible conseguir.
- Al iniciar se debe mostrar un mensaje de preparado, el jugador debe indicar "S" para iniciar.
- Al finalizar se debe mostrar la puntuación del jugador y la puntuación máxima posible que se podía alcanzar.
- Si la puntuación del jugador es igual a la puntuación máxima, se debe indicar con un mensaje.
- Si el jugador alcanza o supera la puntuación máxima que puede procesar el juego, se parará automáticamente y se mostrará un mensaje (hacer uso de la librería Math y los métodos addExact, subExact y multiplyExact que lanzan una excepción ArithmeticException en caso de overflow).

Ejemplo de ejecución:

Bienvenido a MatesTron !!!

En cada turno debes elegir la operación que obtenga mayor puntuación

¿Estas preparado? S / N

S

AL TURRONNNNN!!!!

Elige entre...

1.->4*2

2.->4/100

1

Elige entre...

1.->8/75

2.->8/5

2

Elige entre...

1.->1+62

2.->1*61

2

Elige entre...

1.->61/93

2.->61*28

2

Elige entre...

1.->1708/59

2.->1708+94

2

Has conseguido un total de 1802 de 1802

Muy bien calculín!!!!

Resultados de aprendizaje:

3. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.

Criterios de evaluación:

d) Se han reconocido las posibilidades de las sentencias de salto.

Resultados de aprendizaje:

4. Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.

Criterios de evaluación:

a) Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.

b) Se han definido clases.

c) Se han definido propiedades y métodos.

d) Se han creado constructores.

e) Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.