

SilverCoders

POTENZIAMENTO DELL'ALFABETIZZAZIONE DIGITALE ATTRAVERSO LE
ESPERIENZE DI APPRENDIMENTO PER ADULTI



CHALLENGE #22 IL CACCIATORE DI FORME GEOMETRICHE- LIVELLO AVANZATO

PROGRAMMA DI FORMAZIONE
SUL CODING **PER**
ADULTI +55



SILVER CODERS

Finanziato da n. 2020-1-SE01-KA227-ADU-092582



Co-funded by
the European Union

*Questo documento riflette solo il punto di vista dell'autore e l'Agenzia Nazionale e la
Commissione Europea non sono responsabili dell'uso che può essere fatto delle
informazioni in esso contenute.*

STRUTTURA DELLA CHALLENGE

DESCRIZIONE

In questa challenge viene fornita una configurazione che riprende gli elementi più importanti dell'ambiente Gdevelop: la scena e il foglio degli eventi. Gli eventi disponibili permettono al giocatore di muovere il mostro e di guadagnare i pezzi geometrici che cadono dall'alto. La challenge chiede di migliorare il gioco, rendendolo più dinamico – aggiungendo, ad esempio, delle bombe per eliminare il mostro –.

OBIETTIVO GENERALE

In questa challenge verranno apportate delle migliorie al gioco “Il cacciatore di forme geometriche”, conosciuto come Geometry Hunter, rendendolo così più interattivo e dinamico.

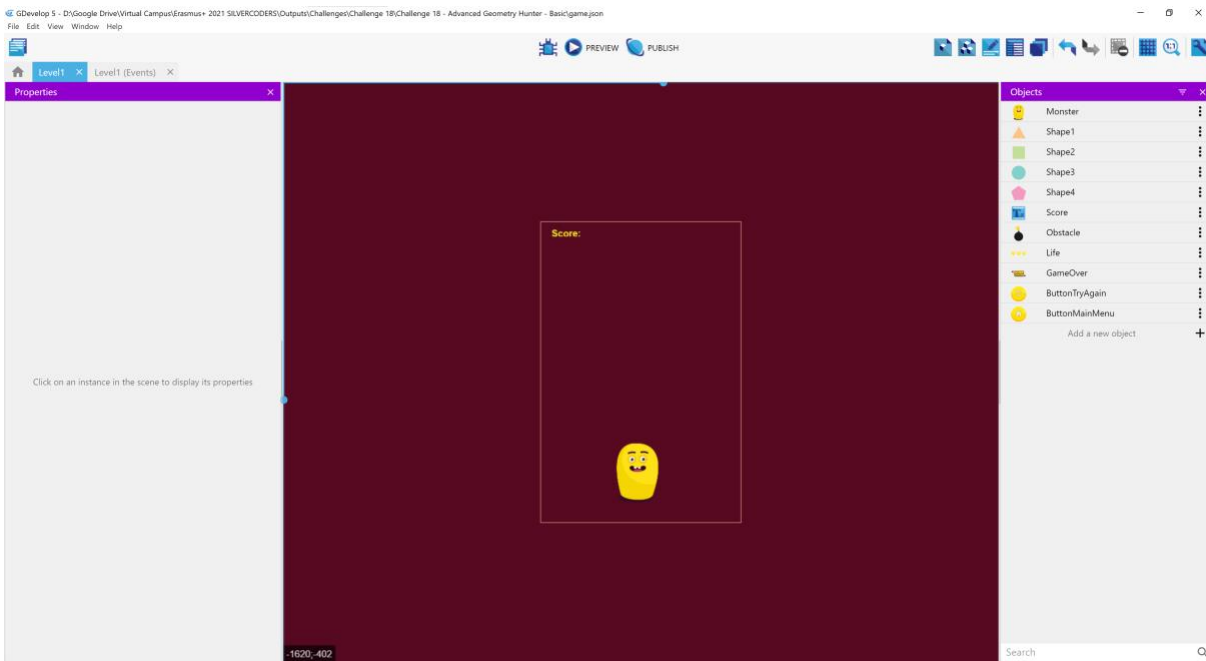
OBIETTIVI DI APPRENDIMENTO

Alla fine di questa challenge, sarai in grado di:

- Acquisire esperienza con una suite di programmazione visuale e codificare con essa semplici software standard.
- Conoscere le dichiarazioni, le linee di comando e il loro significato per il compilatore.
- Scrivere istruzioni utilizzando una sintassi corretta con minimi errori.
- Conoscere gli operatori, la loro funzione e i loro operandi (quali simboli corrispondono a quali operatori).
- Comprendere l'assegnazione di valori alle variabili e come modificarli.
- Conoscere tutte le operazioni aritmetiche di base e come utilizzarle.
- Riconoscere e utilizzare tutte le strutture dati relative ai numeri.
- Identificare le strutture legate all'uso del testo, come stringhe e caratteri.
- Utilizzare correttamente le istruzioni condizionali per eseguire il codice in base ad una condizione fissa definita.
- Utilizzare l'editor di Gdevelop.
- Comprendere i concetti di scene, eventi e oggetti.

ISTRUZIONI

- Inizia aprendo l'editor di Gdevelop.
- Utilizza il menu **File** per **aprire** il gioco Challenge 22 - Base.
- Dovresti visualizzare questo:



- Premi il pulsante **Anteprima** per iniziare il gioco. Puoi spostare il mostro a sinistra e a destra con i tasti freccia della tastiera per prendere le forme geometriche che cadono dall'alto. Per ogni forma geometrica catturata, si acquisisce un punto.
- Ripeti il gioco tutte le volte che desideri. Per iniziare una nuova partita, è necessario chiudere la finestra di gioco e premere il pulsante **Anteprima** nell'editor.
- Ora che conosci i meccanismi del gioco (cosa si può fare nel gioco), approfondiamo come è fatto, concentrandoci sulle differenze rispetto alla challenge precedente.

GAME START		
At the beginning of the scene Add condition	Start (or reset) the timer "ShapeCreation" Add action	

- Quando il gioco inizia, crea un **timer** chiamato **ShapeCreation**, un oggetto che tiene sempre il tempo in secondi.

SHAPES	
@ The timer "ShapeCreation" is greater than 2 seconds Add condition	Among objects Shapes , create object named "Shape" + ToString(RandomInRange(1,4)) at position RandomInRange(80, 640-80);-100 (layer:) Change the angle of Shapes : set to RandomInRange(0,360) Change the scale of Shapes : set to RandomFloatInRange(0.8, 1.6) Start (or reset) the timer "ShapeCreation" Add action

- Quando il **timer ShapeCreation** raggiunge i 2 secondi, viene creata casualmente una delle quattro forme disponibili. Per rendere il gioco più divertente, queste forme hanno grandezza diversa e ruotano su sé stesse mentre cadono dall'alto. Il timer viene riportato a 0 per ricominciare il conteggio.

Move shape according to the game speed	
Add condition	Add to Shapes an instant force, angle: 90 degrees and length: Variable(GameSpeed) pixels Rotate Shapes at speed 90 deg/second Add action

- La forma cade dall'alto, raggiungendo una certa forza verticale.
- Rendiamo il gioco leggermente più complesso: lascia cadere anche alcune bombe che possono togliere la vita al mostro nel caso in cui venisse colpito. A tale fine, utilizza l'oggetto **Ostacolo**.

OBSTACLE	
@ The timer "ObstacleCreation" is greater than 5 seconds Add condition	Create object Obstacle at position RandomInRange(80, 640-80);-100 (layer:) Start (or reset) the timer "ObstacleCreation" Add action
Move obstacle according to the game speed Add condition	Add to Obstacle an instant force, angle: 90 degrees and length: 1.5*Variable(GameSpeed) pixels Change the z-order of Obstacle : set to 4 Add action

- Il codice dell'oggetto **Ostacolo** è simile a quello delle forme. C'è un timer che lo controlla. Come per le forme, l'oggetto ostacolo viene fatto cadere.

Repeat for each instance of Obstacle:	
@ Obstacle is in collision with Monster Add condition	Delete Obstacle Damage Monster , removing 1 from its health Play the sound killed.wav, vol.: , loop: no Add action

- Se si verifica una collisione con il mostro, quest'ultimo viene danneggiato e perde 1 punto salute.

PROPERTIES
BEHAVIORS
VARIABLES
EFFECTS

Flash

Half period (time during which object is invisible), in seconds

0,1

Health

Damage cooldown (in seconds) (0 for no cooldown)

0,8

Health

3

Maximum health (0 for no maximum)

3

StayOnScreen

Bottom margin, in pixels

0

Left margin, in pixels

0

Right margin, in pixels

133

Top margin, in pixels

0

?

HELP

▶

RUN A PREVIEW

+

ADD A BEHAVIOR

CANCEL
APPLY

- La salute appartiene alla proprietà standard del **comportamento** associata agli oggetti. Il punteggio massimo della salute equivale a 3.

The screenshot shows the 'PROPERTIES' tab of the Silvercoders animation editor. The object is named 'Life'. It contains four animations, each with a duration of 0.08 seconds and a loop checkbox:

- Animation #0 Life3:** Shows three yellow hearts. The preview image shows three hearts on a checkerboard background.
- Animation #1 Life2:** Shows two yellow hearts. The preview image shows two hearts on a checkerboard background.
- Animation #2 Life1:** Shows one yellow heart. The preview image shows one heart on a checkerboard background.
- Animation #3 Life0:** Shows no hearts. The preview image shows a checkerboard background.

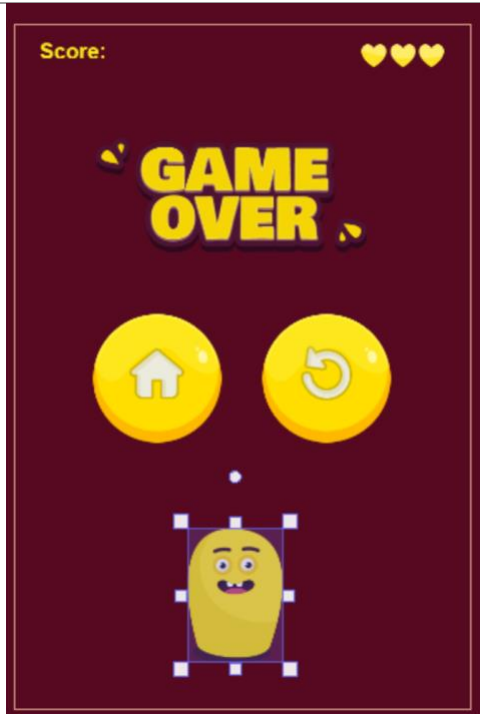
At the bottom of the animation list, there are buttons for 'EDIT COLLISION MASKS', 'EDIT POINTS', 'ADVANCED OPTIONS', and a blue '+ ADD AN ANIMATION' button. Below these are 'HELP' and 'RUN A PREVIEW' buttons, and 'CANCEL' and 'APPLY' buttons.

- Per rappresentare visivamente la salute del mostro si utilizza l'oggetto **Life**. Questo oggetto si compone di 4 fotogrammi, ognuno dei quali rappresenta uno stato di salute.

The screenshot shows the 'HEALTH' event in the Silvercoders event editor. The event is triggered by the condition 'Monster has just been damaged'. The actions are:

- Set animation of **Life** to "Life" + ToString(Monster.Health::Health())
- Make **Monster** blink for 1.5 seconds

- Quando il mostro viene danneggiato, lo Sprite passa al fotogramma successivo.
- Quando il mostro è senza vita, muore. In questo caso, è necessario proiettare sullo schermo il pulsante "GameOver". Crea altri due pulsanti, uno per riavviare il gioco e l'altro per abbandonarlo. Ecco come appare la scena iniziale.



- All'inizio del gioco non si vedono questi oggetti perché sono nascosti.

GAME OVER	
<ul style="list-style-type: none"> At the beginning of the scene 	<ul style="list-style-type: none"> Hide GameOver Hide ButtonTryAgain Hide ButtonMainMenu

- Questi oggetti vengono mostrati quando il mostro muore.

<ul style="list-style-type: none"> Monster is dead 	<ul style="list-style-type: none"> Set animation of Life to "Life0" Set animation of Monster to "MonsterDead" Delete Shapes Delete Obstacle Show GameOver Show ButtonTryAgain Show ButtonMainMenu
--	---

- Controlla il pulsante che ha premuto il giocatore.



RISORSE

Challenge 22 (Base)