SCRATCH Competition

1

Problem 1: Maze Solver (up to 5 points)

Given: Full maze designs with solution directions at each junction, solver sprite with skeleton code for following the maze

Task: The participants are tasked to add code to the maze solver sprite which will cause it to solve the given mazes.

The following features are required:

- The solver should be clearly seen following a correct path through the maze (e.g., it should not just teleport to the finish)
- The solver works autonomously - it should not require assistance from the user
- If the solver touches a maze wall, it should reset to the start point of the maze
- The solver should be able to solve other similar mazes that were not provided to the participants. That is, the solver should be coded in such a way that it would be able to follow any other mazes constructed in a similar way (having the same form of solution directions at each junction).

For this problem, participants must submit the given project template with only the solver sprite's code changed. Solutions which change other aspects of the project will not be graded.

To access the project template click **Problem 1**, click 'See Inside', and then click 'Remix'

SCRATCH Competition

2

Problem 2: Guitar Hero (up to 5 points)
Given: Guitar Hero game background, note sprite

Task: The participants are tasked with making a simple form of the classic game Guitar Hero!

The following elements are required:

- Random notes should periodically appear on the right side of the background, along the horizontal lines
- Notes should move along the horizontal lines to the left side of the screen
- A different keyboard key corresponds to each horizontal line
    - Indicate which key corresponds to which line in some way
    - A “hit” occurs when a player presses the correct key while the correct note is touching the target on the right side of the corresponding line
    - A “miss” occurs when a player presses a key while no note is touching the corresponding target, or when a note passes the target without being “hit”
- A different sound should play when the player either scores a hit or takes a miss
- The player loses when five consecutive misses are taken. Here consecutive means five misses occur without a hit occurring anytime between them

Participants are encouraged to use the provided project template, but are permitted to use other sprites, backgrounds or sounds, so long as the required elements are still fulfilled.

To access the project template click **Problem 2**, click ‘See Inside’, and then click ‘Remix’

SCRATCH Competition

Problem 3: Basketball (up to 5 points)

Given: Basketball, basket

Task: The participants are tasked with designing a basketball simulator.

The following elements are required:

- The basketball should start in one corner of the screen, waiting for user input
- The user should be able to launch the basketball by providing user input
    - The user should be able to adjust the initial speed and direction of the basketball by providing user input
- The ball should exhibit realistic (or nearly realistic) changes to direction and speed due to gravity
- The ball should not leave the screen
- The ball should be able to bounce off at least one object on the screen
- The player should be able to score points by landing in the basket or hitting some sort of target, with the score displayed on the screen
- The player should be able to reset the basketball to the corner (but not reset the score!) with either a keyboard press or a button on the screen

Participants are encouraged to use the provided project template, but are permitted to use other sprites, backgrounds or sounds, so long as the required elements are still fulfilled.

To access the project template click **Problem 3**, click ‘See Inside’, and then click ‘Remix’

SCRATCH Competition

Problem 4: Menu (up to 15 points, and up to a combined 20 points with problem 5)
Note that a project template is not provided for this problem.
Task: Take on your dream of owning a restaurant as you create an interactive restaurant menu! A user should be able to select food choices, but how, when, and what happens next is up to you!

An on-screen waiter might prompt the user with vivid descriptions of the most expensive dishes. Other guests might be seen in the background, entering, leaving, and making conversation. Maybe the chef makes the food to order and you can eat it piece-by-piece. Be creative!

Use this project prompt to show off your ability with the visual design side of the scratch platform. Give the user lots of areas to interact smoothly with the application, and make some unique animations!

Design Document: In addition to your Scratch file, submit a short document (a .txt file from Notepad or Notepad++) describing the features of your project. It doesn't have to be too formal or long - just list the primary ways your user can interact with the project, and describe the features you're most proud of. Pretend you're selling your solution - make sure the judges know about all of the features you spent your time on!

SCRATCH Competition

Problem 5: Calculator (up to 15 points, and up to a combined 20 points with problem 4)
Note that a project template is not provided for this problem.
Task: Your cutting-edge software development company has decided that the traditional calculator needs a facelift. Design a complex math machine to help math students across the country!

Your project should serve as a tool for performing math calculations, including the basics (plus, minus, times, divide, square root, etc.), but also some others that might not be seen on the traditional calculator. Pull some ideas from your math classes - you might help the geometry student with calculating the third side of a triangle, or the algebra student with the quadratic equation or graphing. Be creative!

Use this project prompt to show off your ability with the technical side of the scratch platform. Let the user smoothly perform lots of consistent mathematical operations; make them practical, but come up with some complex ideas!

Design Document: In addition to your Scratch file, submit a short document (a .txt file from Notepad or Notepad++) describing the features of your project. It doesn't have to be too formal or long - just list the primary ways your user can interact with the project, and describe the features you're most proud of. Pretend you're selling your solution - make sure the judges know about all of the features you spent your time on!