

CISC 110 Lab 8: Spear Game

In this lab, you will create a simple game that scores the number of times a player can hit a target with a spear within 25 seconds. The target will move randomly around on the stage. The player uses the arrow keys to move the spear to hit the target as many times as possible.

Complete the following during your lab time.

1. Download and open the Flash file called Lab8Start.fla. This file contains:
 - Two MovieClip symbols with instance names **target** and **spear**
 - Two dynamic text fields with instance names **msgBox** and **scoreBox**
 - One input text field with instance name **nameBox**
2. In the Actions layer, define an int variable called **score** and set it to 0.
3. Write a function called **reverse** that returns the reverse of the string that is given as its parameter. For instance, the function call **reverse("glop")** will return **"polg"**. Test your function by calling it with sample values and using a trace statement to display its result in the Output Panel. Here is the header for your function:

```
// Returns the String s reversed
// For instance, if s == "jello", returns "ollej"
function reverse ( s: String ): String
```

HINT: Define a String variable called **temp** and set it to the empty string: **""**. Then concatenate letters from **s** onto **temp** using **s.charAt(i)** within a for loop to create a reversed string.

4. Add a **FocusEvent.FOCUS_OUT** event listener to the nameBox text field to detect when the player has typed in their name and then deselected the text field. Specify that an **initializeGame** function will be called when that event occurs.
5. Write the **initializeGame** function to do the following:
 - Create a String variable, name, and set it to be the value of the string in the **nameBox** text field
 - Reverse that name using function **reverse** that you wrote (only because we needed a reason for you to write a String function...)
 - In the **nameBox** text field, display "Player: " plus the reversed name
 - In the **scoreBox** text field, display "Score: 0"
 - In the **msgBox** text field, display "Go!"
6. Type in the following event listener to allow the player to move the spear with the arrow keys and cause the spear to point in the direction it's moving:

```
stage.addEventListener( KeyboardEvent.KEY_DOWN, moveSpear );
```

```
function moveSpear( evt: KeyboardEvent ): void
{
    if( evt.keyCode == Keyboard.RIGHT )
    {
        spear.x += 20;
        spear.rotation = 90; }
    else if ( evt.keyCode == Keyboard.LEFT )
    {
        spear.x -= 20;
        spear.rotation = -90; }
    else if ( evt.keyCode == Keyboard.UP )
```

```

    {
        spear.y -= 20;
        spear.rotation = 0; }
    else if ( evt.keyCode == Keyboard.DOWN )
    {
        spear.y += 20;
        spear.rotation = 180; }
}

```

7. Type in the following functions, which you will use in the next steps. You wrote the **randomNumber** function during the last lab.

The **moveOnStage** function moves the target on the stage **xAmt** in the x direction and **yAmt** in the y direction. It keeps the target on the stage at least 50 pixels away from all borders.

Note: if **xAmt** is negative, target moves up; if **xAmt** is positive, it moves down
 if **yAmt** is negative, target moves left; if **yAmt** is positive, it moves right
 if **xAmt** or **yAmt** is zero, it doesn't move in that direction

// Moves target on the stage xAmt in the x direction and yAmt in the y direction
 function moveOnStage(xAmt: int, yAmt: int)

```

{
    if( target.x < 50 || target.x > stage.stageWidth - 50 )
        target.x = 100;
    else
        target.x = target.x + xAmt;
    if( target.y < 50 || target.y > stage.stageHeight - 50 )
        target.y = 100;
    else
        target.y = target.y + yAmt;
}

```

// Returns a random integer between 0 and n

```

function randomNumber( n: int ): int
{
    return Math.round( n * Math.random( ) );
}

```

8. Create a **Timer** object to fire an event every 1/4 of a second (every 250 ms) for 25 seconds (so 100 events) and call a **moveTarget** function each time it fires an event. Add a line to the **initializeGame** function to start the timer after the player types in their name.
9. Write the **moveTarget** function to do the following:
- Call the **randomNumber** function twice to get a random direction between 0 and 3 (to represent up, down, left, right) and a random distance between 0 and 50.
 - With an if-else statement to check the direction and call the **moveOnStage** function to move the right amount in the right direction.
10. Add the following code within the **moveTarget** function before you move the target to add one to the score if the player has hit the target (using the ActionScript method 'hitTestObject'):

```

// Check if player has hit target
if ( spear.hitTestObject( target ) )
{
    score = score + 1;
    scoreBox.text = "Score: " + score;
}

```

11. Play the game!