CSS385: Assignment 1 Question Sheet

Instructions:

- In questions where you are asked to explain, please be concise.
- Show your work when necessary, be neat, precise, and brief!
- To help us grade your assignments and return this to you in a timely fashion please:
  - Put your name and answers in the answer sheet only (separate link provided). Anything you write outside of the answer sheet **will not** be graded.
  - Provide your answers in the order of the problems.
  - Please use only one side of 8.5x11 paper.
  - Please make sure you bring a hardcopy print out **of the answer sheet (!!only!!)** to submit at the beginning of class. Please do not print out the problems.

Your assignment may not be graded if any of the above is violated, you have been warned.

1. **(5pt)** The Development Environment:
   a. **(1pt)** Which is the file you need to modify to change the title on the game window? **AssemblyInfo.cs**

   b. **(1pt)** When you change the icon for your application, which of the file(s) is/are changed in your development environment? **The .csproj file (say .ico is ok)**

   c. **(1pt) True or False**: In all of the provided *ClassExample* projects (the ones under **In Class Examples** off our course website), you can remove the *MonoGameLib* folder after you build the project for the first time and continue to build the project successfully.

False! During each re-build all the dll files are copied from the MonoGameLib folder over to our debug folder.

Question: what is the difference between **debug** and **release** build?

OK to say True and explain future compile will fail.

   d. **(1pt) Ture or False:** In mp1, your game is based on GTCS1Lib and you have no access to the functions provided by MonoGame.

False! It is true that mp1 is based on GTCS1Lib, but it is not true that we do not have access to functions provided by MonoGame (e.g., Keyboard).

   e. **(1pt)** By default, all provided class example projects define the default game class name as ***ClassExample*** in ClassExample.cs, where we can see this line:

```
public class ClassExample : XNACS1Base
```

In MP1, you are told to change the name of this class to something else. Identify at least one other file you must modify after this name change.

Program.cs: we must change the new ClassExample() to the new class name.

2. **(4pt)** GTCS1Lib SDK. Let's evaluate the claim that GTCS1Lib is a SDK:

    a. **(1pt) True or False:** An API documentation is provided for the developers.

True: let me know if you don't know where is the API doc.

    b. **(1pt) True or False:** There are tutorials showing the developers how to work with the API.

True! I hope you have read (at least the first three of) these.

    c. **(1pt) True or False:** An IDE wizard is provided to assist developer creating their own projects from scratch.

False! Do you actually know how to create a project from scratch?

    d. **(1pt) True or False:** The SDK is self-contained where developers do not have to be concerned with installing other API's.

False! I thought so in the beginning, but anyone remember OpenAL?

3. (**2pt**) Vector length and direction:

    a. (**1pt**) What is the length of $\vec{v} = \begin{bmatrix} 8 \\ 15 \end{bmatrix}$?

Length = sqrt(8*8 + 15*15) = sqrt(64+225) = sqrt(289) = 17

    b. **(1pt)** What is the direction of this vector? $\begin{bmatrix} \frac{8}{17} \\ \frac{15}{17} \end{bmatrix}$

4. (**3pt**) $\vec{V}$ is my current Velocity (in Vector2), and user gives me a new Vector2 quantity, $\vec{D}$. Given that both $\vec{V}$ and $\vec{D}$ are non-zero vectors, which of the following defines a new velocity that travels at the **speed defined by $\vec{V}$** in a the **new direction defined by $\vec{D}$**.

    a. Velocity (Displacement/Time) = $\hat{V} * \hat{D}$
    b. Velocity (Displacement/Time) = $\vec{V} * \vec{D}$
    c. Velocity (Displacement/Time) = $\hat{V} * \vec{D}$
    d. Velocity (Displacement/Time) = $\vec{V} * \hat{D}$

e. Velocity (Displacement/Time) = $\vec{V}.Length$ * $\vec{D}.\mathrm{Length}$

f. Velocity (Displacement/Time) = $\boldsymbol{\vec{V}.Length}$ * $\boldsymbol{\hat{\mathrm{D}}}$

g. Velocity (Displacement/Time) = $\hat{\mathrm{V}}$ * $\vec{D}.\mathrm{Length}$

h. All of the above

i. None of the above

5. (**4pt**) We have a game window with resolution 1024x768, and in ***InitializeWorld()*** we call:

World.setWorldCoordinate(new Vector2(9, 6), 60)

a. (**1pt**) What is the Width of the world? **60**

b. (1pt) What is the Height of the world?

(768/1024) * 60 = 45

Tutorial 1: says aspect ratio is fixed at 16:9, so, 33.75 is an acceptable answer

c. (**2pt**) Assume the ***Update()*** function is called exactly 30 times a second, if we want a ball to travel *horizontally* from left of the window to the right of the window in 4 seconds. In your code, what value should you set for the speed of the ball?

Distance to cover = 60

Speed = distance / (second * ticks-per-second) = 60/(4 * 30) = 1/2 = 0.5