

Name: WEI Wuhao

The extensions:

1 Add interface to change all configure about the game.

1) You can **change** the mushroom number, game speed, and wormholes. By changing the above, the **score** you get every step the worm moves and the **difficulty** will adjust **accordingly**. (The EveryStepScore means the score you get every move the worm makes.)

2) And you can change the default **user name**, it will concerns the name in the scoreboard if you enter **the top ten**.

Usr:	<input type="text" value="Alex"/>	Usr:	<input type="text" value="Emma"/>
Mushroom	<input type="range" value="56"/>	Mushroom	<input type="range" value="131"/>
speed	<input type="range" value="2"/>	speed	<input type="range" value="8"/>
wormhole	<input type="range" value="5"/>	wormhole	<input type="range" value="5"/>
Difficulty	<i>Novice</i>	Difficulty	<i>Professional</i>
EveryStepScore:	10	EveryStepScore:	115

2 Add a computer player.

The AI of the computer player are based on **greedy algorithm** and a tricky part **aliasing**.
two algorithms:

1) greedy algorithm

First, it detects the empty spaces (grass land) in the four directions of the worm head in order to find the **maximum** empty spaces which will not make it dead and longer.

Second, if no empty spaces around it, the **local optimum** will be eat the mushroom, or it will die.

2) aliasing

Since we have not covered the error handling, I use the aliasing to make the index in bound without too much detecting the bound

3 a high score table

Back	ScoreBoard
Rank 1:	Emma 4485
Rank 2:	Alex 1610

The scoreboard only contains topten list.

Note when testing:

1 The **computer player** may has alert that it has entered the top ten, but I think there is none sense to put it in top ten. So only human players will enter the top ten.

2 Do remember input **return or enter key** in the keyboard after you enter the name in the settings. Or your the name you input will not change in scoreboard.

4 countdown 3-2-1 before the worm start moving.

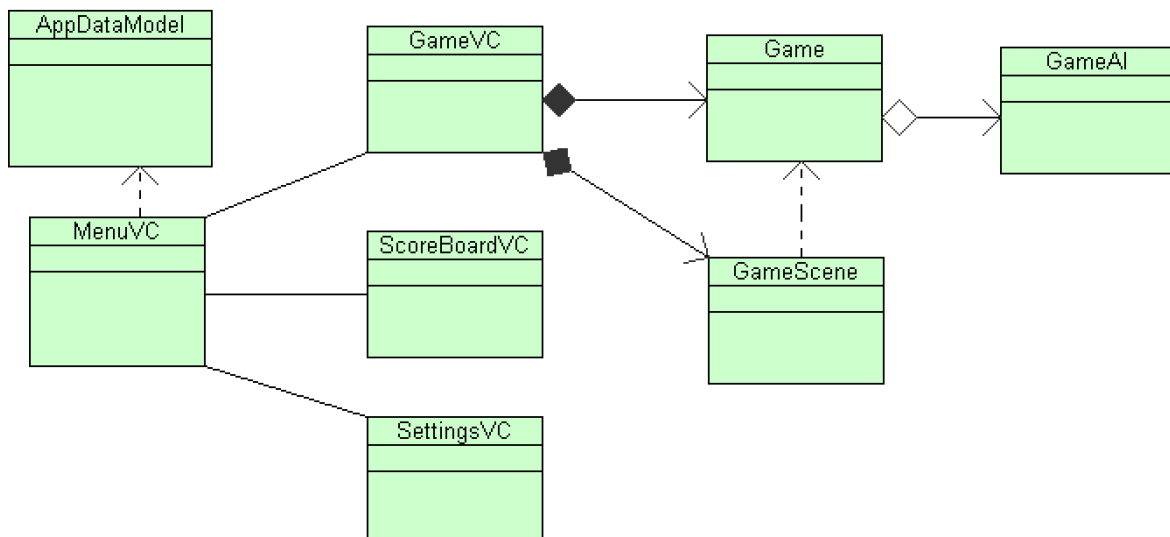
5 smiley face on the head of the worm

6 other slight improvements:

- 1) more ***friendly*** human interface:
 - 1> use swipe to control the worm move.
 - 2> game launching interface to choose
- 2) use ***persistent*** storage

The class diagram:

Class Diagram

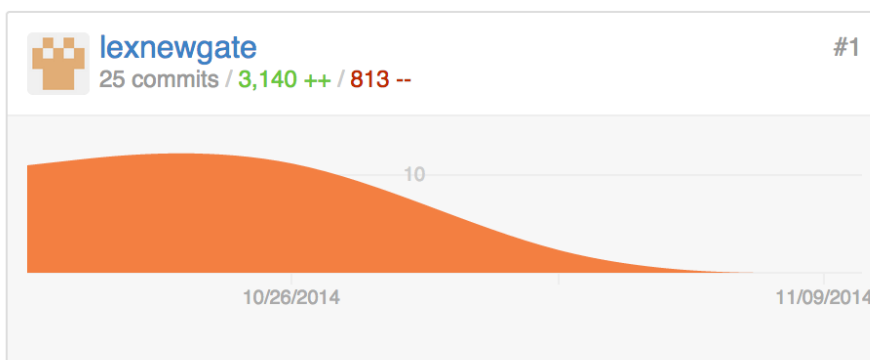


Critical appraisal:

strengths:

1 develop process:

1) I don't finish it in ***one day***, which sort of demonstrates the project doesn't cover lots of obvious mistakes. The following graph is the ***code frequency*** from github.



2) use **prototype** and **test-driven** development

In every period of development, I always do the **code interface** first, then test the **data flow**.
In some complex implementation, I have done some unit testing.

2 Try to use **every bit knowledge** in slides.

1) code structure:

Separate the data model from the view from the **slider6 mvc**.

2) data transfer:

Use the **prepareForSegue** other than **singleton** from the **slider10 storyboards**.

3) data structure:

Use the **Foundation framework classes** such as NSMutableArray other than **c array** from the **slider5 objectivec**.

4) Even the design:

In the settings interface, I combine the **slider2** and **slider6**. I implement the **persistent storage** and **swipe** based on the **slider11 property lists** and **slider8 touch**.

And so on.

weakness:

1 about the **game AI**

1)AI **logic**

Obviously, **only** greedy strategy for it is **not enough**. About half chance, it will move in a **perfect circle**. Here perfect circle means the head chase its tail in a circle, which obviously run into a **infinite loop**.

2)AI **code structure**

Actually, I have asked you about the design which the answer is implement a **parent** class **Player** and two **son** classes: **HumanPlayer** and **ComputerPlayer**. I was not able to think how to figure it out. So the result is I **embedded** the **AI code** into the **GameVC** which is certain a bad design.

2 code **structure** in some implementation

Some coding is done in **causal** time. So some parts of implementation have some **experimental code**. And due to time management, I have not got time to **refactory** that part.