## 24780 - Engineering Computation | Problem Set 05 | ypatawar

This Program intends to be a game where you start as a **Vampire** (represented as red), and your objective is to bite all other people (represented as green) in the window.

1. The program starts with the Question on the console stating, "**`How many people blood you want to drink?`**" The user must input the number of people to be displayed on the graphics window.
2. The next question asks the user for the type of game mode:
   a. Fast (3) – vampire moves at high speed (75 ft/s)
   b. Normal (2 - default) - vampire moves at normal speed (50 ft/s)
   c. Slow (1) - vampire moves at slow speed (25 ft/s)
3. Once these questions are answered, the graphic window opens, and people randomly start coming into the window.
4. The user can use Up, Down, Right and Left arrow keys to move across the window but there is one condition where the user loses the game:
   a. The user goes out of bounds, a dotted red quadrilateral is displayed for the boundary and if the user crosses it, they lose (User can touch the line but not cross it!). If they cross the line, it means they went into sunlight and they die!
5. One place that the people cannot be bitten is shown by a black dotted quadrilateral – which is supposed to be the safe zone – People will only remain healthy if they are within this safe zone. The vampire cannot affect healthy person within this area but if the person leaves the safe zone, they can be bitten!
6. The objective is to bite all people in least amount of time. If the vampire goes within 10 feet of a person, the person turns into a zombie (shown by blue).

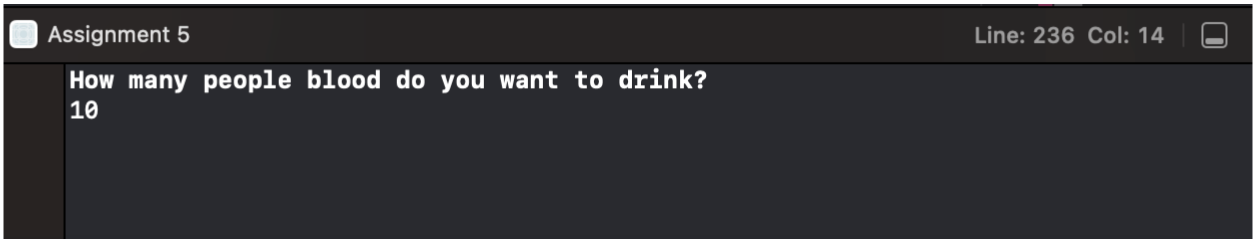
### Features for satisfying requirements:

1. Double Buffering animation with FsSwapBuffers() function is used
2. The game is over if:
   a. User presses 'Esc'
   b. User goes out of bounds
   c. User bites all people (Win)
3. It is an interactive program since the user takes control of the vampire and tries to move as close as possible to the person(s)
4. The program uses Line Stipple feature of OpenGL for rendering the safe zone and bounds of where the user can move
5. The OpenGL primitives that are being used are:
   a. GL_TRIANGLE_FAN – For rendering people as circles
   b. GL_LINE_STRIP – To render the safe zone and the bound quadrilateral
6. The program uses the following functions –
   a. Math library function – Uses 'cos' and 'sin' to render circle, uses 'sqrt' function to calculate distance between two people
   b. State transition – To indicate if a person is human or zombie

c. Numerical integration using Euler's method – To ensure that people stay in the same boundary

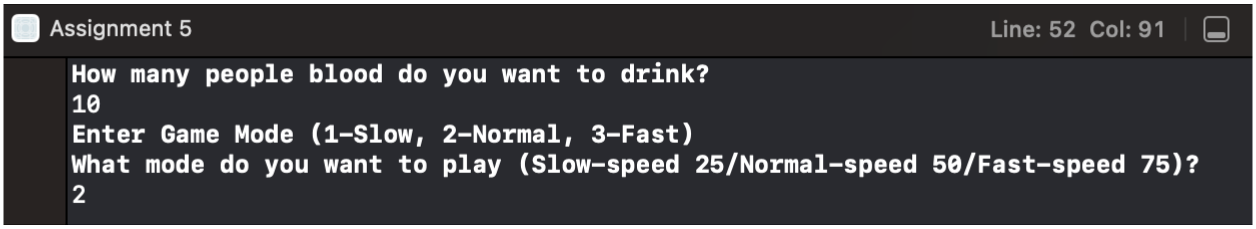
Screenshots of multiple scenarios in game (Not all inputs are same for the screenshots)

1.



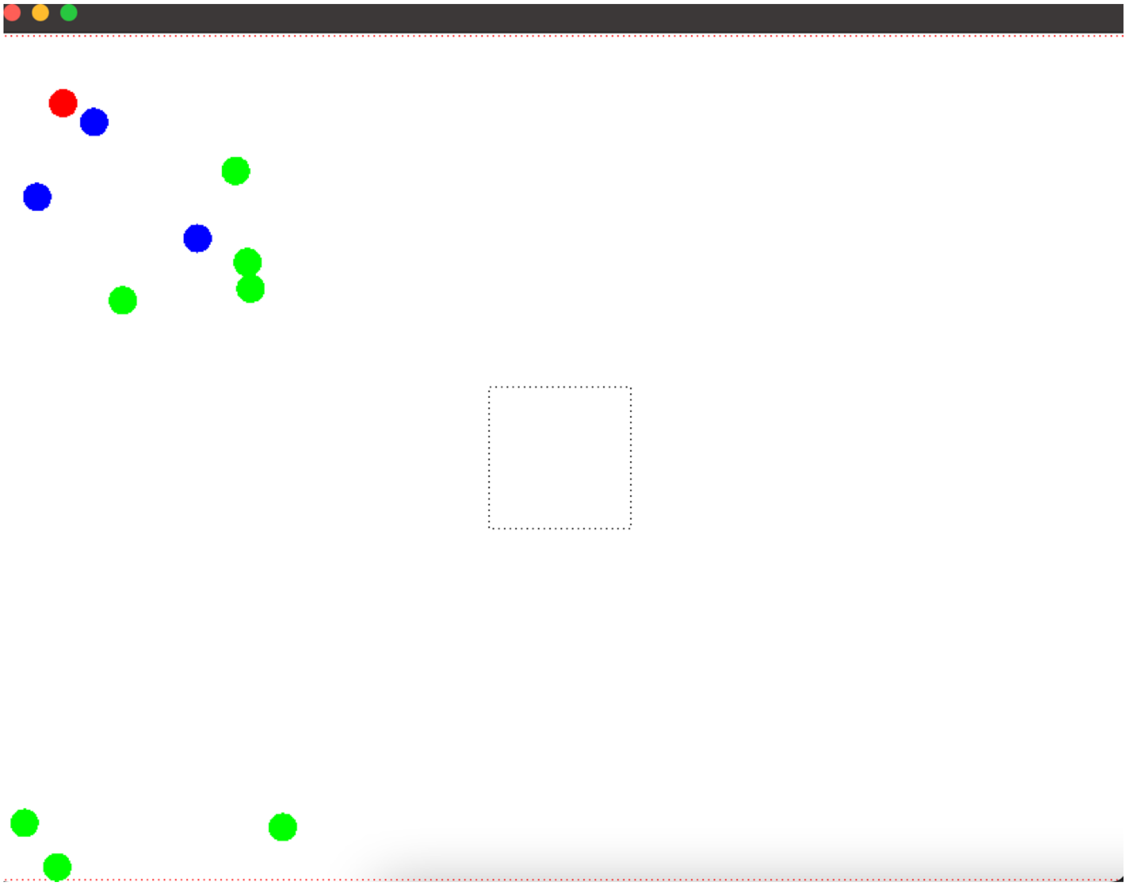
```
How many people blood do you want to drink?
10
```

2.

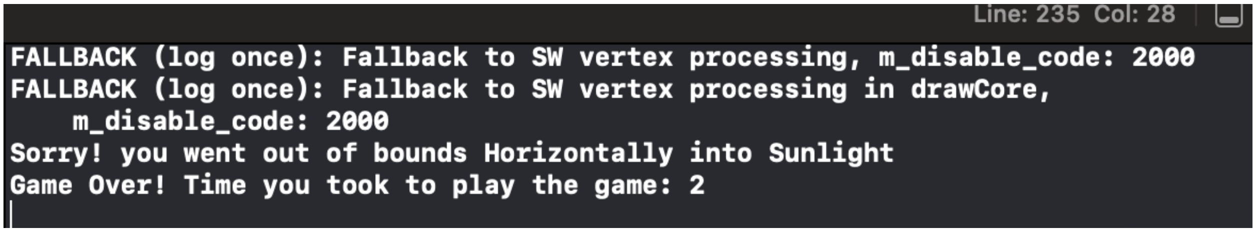


```
How many people blood do you want to drink?
10
Enter Game Mode (1-Slow, 2-Normal, 3-Fast)
What mode do you want to play (Slow-speed 25/Normal-speed 50/Fast-speed 75)?
2
```

3.

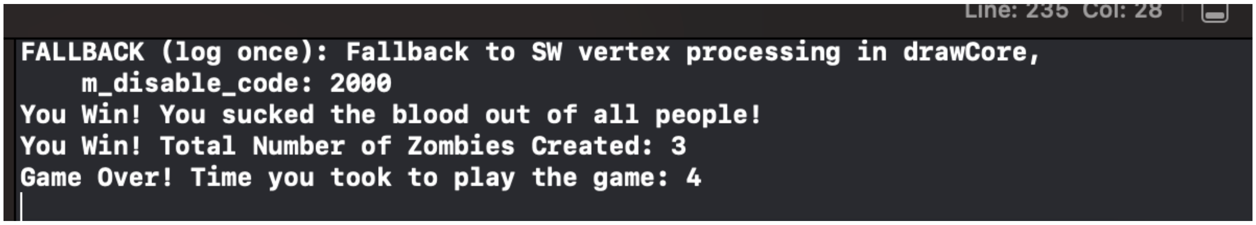


4.



```
FALLBACK (log once): Fallback to SW vertex processing, m_disable_code: 2000
FALLBACK (log once): Fallback to SW vertex processing in drawCore,
    m_disable_code: 2000
Sorry! you went out of bounds Horizontally into Sunlight
Game Over! Time you took to play the game: 2
```

5.



```
FALLBACK (log once): Fallback to SW vertex processing in drawCore,
    m_disable_code: 2000
You Win! You sucked the blood out of all people!
You Win! Total Number of Zombies Created: 3
Game Over! Time you took to play the game: 4
```