1. **PROJECT PROPOSAL(FILE).**

➔ PROJECT DESCRIPTION

This is a project that looks to focus on students in ALU who are having a lot of work that they can't handle and this might lead them to have a mental breakdown. So this project is a 2d game that tends to help the ALU students escape from the pressure of the work to maintain their mental health. Since games are important they help people escape the problems of the real world and just have fun when playing the game. That is why games were created.

➔ PROBLEM STATEMENT

Jacob is a new first-year student at the African Leadership University who has no idea of how the ALU learning curriculum is. After a few weeks, Jacob has been panicking due to a lot of assignments given, so he had to attend therapy for some weeks to keep him mentally safe. As a Computer science student, how can I help the first-year students to keep them mentally ok so that they have any mental issues?

➔ PROCEDURES AND METHODS

As soon as the snake game app opens, a nice background will be displayed and a snake will also appear on the screen which the user will be able to control. The main goal of the game will be for the snake to eat the food and get the snake as long as possible without hitting the body of the snake Every time a user eats the food a score will be added to the user's current scores. It's going to be an endless game since the majority of people enjoy endless games

➔ ANTICIPATED OUTCOMES

To create a Snake game that allows users to control the movement of a snake on a screen, to get points for eating food and avoiding running into the snake itself.

2. MAIN: LIST OF CLASSES(FILE).

‘’’ Our first class is the snake class. This class contains objects of our snake, its properties, and functionalities. For instance; it handles the movement of our snake, the intersection of our game whenever the snake touches its tail, and also draws our rectangle(snake). The object on the surface of the game screen. And whenever this class is called upon, it will not only display our snake but also call our different objects with its functionalities.’’’

```
class Snake:
  def __init__(self):
     self.length = 1
     self.positions = [((screen_width / 2), (screen_height / 2))]
     self.direction = random.choice([up, down, left, right])
     self.color = (17, 24, 47)
     self.score = 0
```

‘’’ Our second class is a food class. This class contains objects that the functionalities of our food. For instance, we need to place our food in different positions whenever our game reset. This class also draw our rectangle(food) object on the surface of our game screen. This class also relates to the other classes as it does only display our food on the screen but also position it randomly’’’

```
class Food:
  def __init__(self):
     self.position = (0, 0)
     self.color = (223, 163, 49)
     self.randomize_position()
```

3. Pseudocode

Pseudocode:

```
Create snake  Class
Define init to declare self,snake_length,snake_position
Declare self.length
Declare self.position
Declare self.direction
Declare self.color
Declare self. Score
```

4. SNAKE FOOD CLASS(FILE).

Pseudocode:

```
Create Food Class
Define init to declare self
Declare self.position
Declare self.color
Declare self.randomize_position
```

Tests

Test number	Description	Test data	Expected result	Actual result	Pass/Fail
1	Snake game reset when it touches its tail	Def reset: Game reset based on action trigger by the user	Game reset when the snake intersect with its tail	The game reset when the snake touches its tail	Pass
2	When a user presses a key on their keyboard, the program should respond to the key pressed	Handle keys: quit, keydown, keyup, key left, key right,	The game should respond to user inputs when triggered	Quit game, snake move based on keys pressed	Pass
3	Def draw: draw a rectangle on the screen surface.	Draw snake: draws a rectangle on the game screen	It should draw the only rectangle and not circle or others	Draw rectangle when the snake eats a piece of food	Pass
4	Def turn, turns snake in a	Def turn:	Should not	Move snake	pass

	different direction	Checks for snake position and length size before the snake move	result in an error when moving the snake in a different position	head based on the snake’s direction and position	
5	Add a rectangle to snake tail whenever snake eats a new food	Def move: self.position. insert	The program should add a new rectangle based on the exact position of the snake.	Adds a new rectangle of the snake’s current position.	Pass