# ASTRO GLIDER

A logo of a university of engineering and technology

Description automatically generated

# Session 2023 – 2027

# Submitted by:

Muhammad Hassaan Shahid 2023-CS-79

# Supervised by:

Dr. Muhammad Awais Hassan

# Course:

CSC-102 Programming Fundamentals

Department of Computer Science

# University of Engineering and Technology

# Lahore Pakistan

**Story of game:**

Embark on a thrilling space adventure in Astro-Glider! Your mission is to guide a high-tech spacecraft to safely land on a designated target area in various galaxies across the universe. However, beware of the treacherous obstacles and three formidable enemies that seek to thwart your journey.

**Game Characters Description:**

**Player (Spacecraft):**

* Controlled by the player using directional keys.
* Equipped with the ability to shoot projectiles by pressing the space button.
* Limited fuel capacity that depletes with each movement.

**Enemies:**

* Three adversaries with varying speeds across different levels.
* Move horizontally, attempting to intercept the player's spacecraft.

**Game Objects Description:**

**Hurdles:**

* Diverse obstacles scattered throughout levels.
* Complexity and arrangement of hurdles change with each level.

**Rules & Interactions:**

* The player must navigate the spacecraft, avoiding collisions with both the enemies and hurdles.
* Colliding with an enemy sends the player's spacecraft back to the starting position.
* The spacecraft's fuel decreases with every movement, requiring strategic planning and efficient navigation.
* Shooting projectiles can neutralize enemies, providing a brief window of safety.
* Advancing through levels unlocks new challenges, with increasing difficulty in hurdle arrangements and enemy behaviors.

**Goal of the Game:**

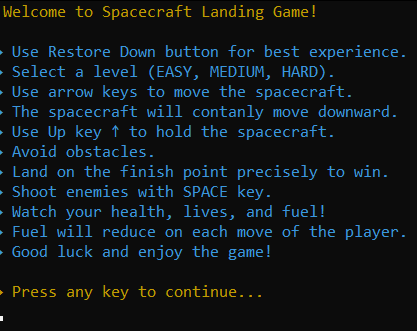
Astro-Glider aims to captivate players by offering an exhilarating and challenging space adventure. The game encourages users to strategize, demonstrate agility, and master precise movements to conquer obstacles, defeat enemies, and successfully land the spacecraft at the designated target area.

The game's dynamic levels and progressive difficulty levels are designed to engage and entertain players, fostering an enjoyable and immersive gaming experience. By combining reflexes, quick decision-making, and skillful maneuvering, players will find themselves immersed in an addictive and rewarding gaming adventure set amidst the vastness of outer space.

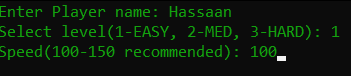
**Wireframes of the Game:**



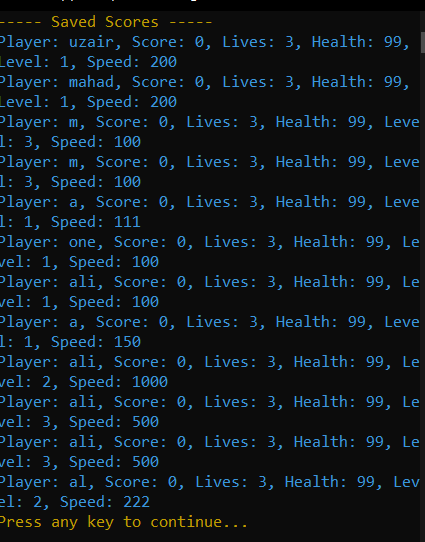
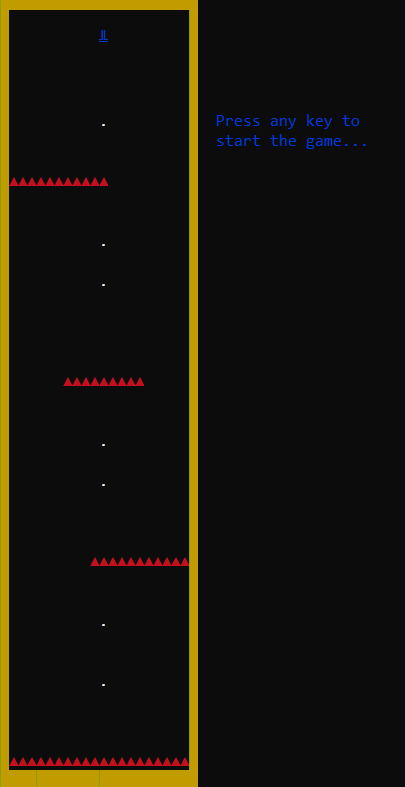
**MAIN PAGE**



**INSTRUCTIONS PAGE**



**OPTIONS**

**HISTORY** **VIEW**

Variables:

const int HEIGHT = 40;

const int WIDTH = 22;

int pX = 10, pY = 1; // Player position

int health = 1; // health count

int lives = 3; // lives count

int score = 0; // score count

bool gameOver = false;

int level;

int enemy1X = WIDTH - 3, enemy1Y = 10; // Enemy 1 position

int enemy2X = WIDTH - 3, enemy2Y = 18; // Enemy 2 position

int enemy3X = WIDTH - 3, enemy3Y = 30; // Enemy 3 position

int enemySpeed = 1; // enemy speed

int endY = HEIGHT -2, endX = WIDTH - 2; // Finish point

int deadY = HEIGHT -1, deadX = WIDTH - 2; // dead point

char maze[HEIGHT][WIDTH]; // Maze array to hold maze layout

int FinishPointX = 16, FinishPointY = 37; // Finish Point positions

int FinishPointSpeed = 1; // Speed at which the FinishPoint moves

int fuel = 100; // The initial fuel level

int fuelX = 5; // The X position of the fuel level display in the maze

int fuelY = 5; // The Y position of the fuel level display in the maze

int projectileX = pX, projectileY = pY+1;

bool projectileActive = false;

string name;

Prototype:

void drawMaze1();

void drawMaze2();

void drawMaze3();

void loseGame();

void livesEnded();

void endGame();

void movePlayer();

void moveFinishPoint();

void drawEnemies();

void moveEnemies(int level);

void shootProjectile();

void moveProjectile() ;

void checkCollision();

void updateGame();

void gravity();

void fuell();

void resizeConsole();

void cursorHide();

void cursorShow();

int starter();

bool containsOnlyIntegers(string integer);

void instruction();

bool containsOnlyAlphabets(string word);

void saveScore();

void viewScore();