**Fall 2024: Programming Fundamentals (CS1002)**

**Self-Evaluation Sheet**

**Student Information**

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_muhammad yahya\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Roll #: \_\_\_23i\_\_\_\_-0729\_\_\_\_\_\_\_\_\_\_\_

Section: \_\_\_B\_\_\_\_

**Evaluation Rules**

Assign yourself full marks if you claim a complete implementation of the given Question. Assign yourself zero marks if you have missed the implementation of the given Question.

**Phase 1: Implementation (160 Marks)**

| # | Self Evaluation Sheet | | Marks | Obtained Marks |
| --- | --- | --- | --- | --- |
| 1 | Implementation of Spray Can mechanics (20 Marks). | |  | -- |
|  |  | Movement: Moves left and right across the entire screen. | 5 | 5 |
|  |  | Sprays: Clearly show the number of sprays left for the spray can. | 5 | 5 |
|  |  | Lives: Visually display the remaining lives to the player. | 5 | 5 |
|  |  | Accurately sprays one shot at a time. | 5 | 5 |
| 2 | Correct implementation of bee types (worker and killer bees) (30 Marks). | |  | -- |
|  |  | Correct Movement of Bees: collision with borders, and alternative left right movement. | 10 | 10 |
|  |  | Worker bees collide with honeycombs, but hunter bees do not. | 5 | 5 |
|  |  | Worker bees occasionally stop randomly for a short duration. | 5 | 5 |
|  |  | Worker bees turn into yellow honeycombs, while hunter bees turn into red honeycombs. | 5 | 5 |
|  |  | Bees properly exit the borders after pollinating the flowers. | 5 | 5 |
| 3 | Implementation of flowers (20 Marks). | |  | -- |
|  |  | The first bee creates two consecutive flowers on the left and right borders. | 10 | 10 |
|  |  | Subsequent bees create only one flowers. | 5 | 5 |
|  |  | A flower is created in the middle if a bee reaches the center. | 5 | 5 |
| 4 | Functionality of the hummingbird and its interactions (35 Marks). | |  | -- |
|  |  | Movement: Randomly decides the direction to move. | 5 | 5 |
|  |  | Movement: Travels several blocks in the chosen direction. | 5 | 5 |
|  |  | Movement: Pauses briefly before changing direction. | 5 | 5 |

|  |  | Movement: Properly navigates and moves across the entire screen. | 5 | 5 |
| --- | --- | --- | --- | --- |
|  |  | Interaction: Eats a honeycomb upon reaching it, awarding the player points. | 5 | 5 |
|  |  | Interaction: Becomes sick and exits the screen when hit (3 times). | 5 | 5 |
|  |  | Return: Reappears after a delay if it exited due to sickness. | 5 | 5 |
| 5 | Honeycombs and hives (20 Marks) | |  | -- |
|  |  | Collision: Honeycombs and hives properly interact with sprays shot by the spray can. | 5 | 5 |
|  |  | Hive Creation: A bee correctly forms a hive when it gets stuck. | 15 | 15 |
| 6 | Accurately implements all 3 levels, ensuring all elements are correctly created as specified in the project PDF (15 Marks). | | 15 | 15 |
| 5 | Accurate scoring mechanism as per gameplay rules (10 Marks). | | 10 | 10 |
| 8 | Complete and navigable game menu (10 Marks). | | 10 | 10 |

**Phase 2: Implementation (140 Marks)**

| # | Self-Evaluation Sheet | | Marks | Obtained Marks |
| --- | --- | --- | --- | --- |
| 1 | High-score tracking system using file handling (40 Marks) | |  |  |
|  |  | Stores both player name and high score in file. | 10 | 10 |
|  |  | High scores are stored in ascending order. | 5 | 5 |
|  |  | File Handling: Proper implementation of file handling to store and retrieve high scores. | 5 | 5 |
|  |  | High scores are accessible from the game menu. | 10 | 10 |
|  |  | Displays updated high scores when the player wins the entire game or loses. | 10 | 10 |
| 2 | The Boss Level is properly created, accessible from the main menu, and includes all elements as specified in the project PDF. (10 Marks) | | 10 | 10 |
| 3 | Power-ups (45 Marks). | |  |  |
|  |  | Power-ups must create noticeable changes to the spray can, ensuring the game remains playable. | 15 | 15 |
|  |  | Each power-up should have a timer bar that visually represents its duration. The timer must gradually decrease, and the effect ends when the timer depletes. | 15 | 15 |
|  |  | If the player picks up the same power-up while it's active, the timer resets, extending the effect. | 7.5 | 7.5 |
|  |  | Opposing power-ups should cancel each other out, and the effect should end instantly. | 7.5 | 7.5 |
| 4 | Infant Bee Mechanic (if not created dynamically then this entire section will be a straight 0) (45 Marks) | |  |  |
|  |  | The Infant Bee must spawn dynamically from the top of the bee hive. It should move upward and avoids | 30 | 30 |

|  |  | obstacles by moving left or right. If trapped, it transforms into a new hive. |  |  |
| --- | --- | --- | --- | --- |
|  |  | Once the Infant Bee reaches the top, it should matures into a Hunter Bee. | 5 | 5 |
|  |  | Killing the Infant Bee in its child form should result in a 500-point penalty. | 5 | 5 |
|  |  | The Infant Bee spawns after a fixed interval. The spawn interval should be balanced—long enough to clear other bees, but short enough to maintain challenge. | 5 | 5 |

**Bonus Section (60 Marks)**

| # | Self Evaluation Sheet | Marks | Obtained Marks |
| --- | --- | --- | --- |
| 1 | View implementation (double grid, dynamic view shifts) | 50 |  |
| 2 | Animation of the Infant Bee using a sprite sheet | 5 |  |
| 3 | Upload of project code on GitHub with a detailed README.md | 5 | 5 |

Eligibility for Bonus: Bonus points will only be awarded to students who have fully implemented the required features in Phase 1 and Phase 2.

Total Marks for Phases 1 & 2: 300

Bonus Marks: 60

Overall Total (with bonus): 360