

Faculty of Computing and Informatics (FCI)

Multimedia University

Cyberjaya

**TCP 2201 - Object Oriented Analysis and Design**

**Trimester 2, 2019/2020**

**Assignment : Ants and Bugs**

|  |  |
| --- | --- |
| No. | Name |
| 1. | Amiera Nur Hafidzah binti Muzaffar (Leader) |
| 2. | Perivitta A/P Rajendran |
| 3. | Amirah Arisya binti Mohamad Noor Steven |
| 4. | Sarah Batrisyia binti Ahmad Salman |

***TABLE OF CONTENTS***

|  |  |
| --- | --- |
| ***Title*** | ***Page*** |
| Introduction | 2 |
| UML Diagram | 3 |
| Use Case Diagram | 4 |
| Sequence Diagram | 5 |
| User Documentation | 6-7 |
| Screenshots of the Program | 8-10 |

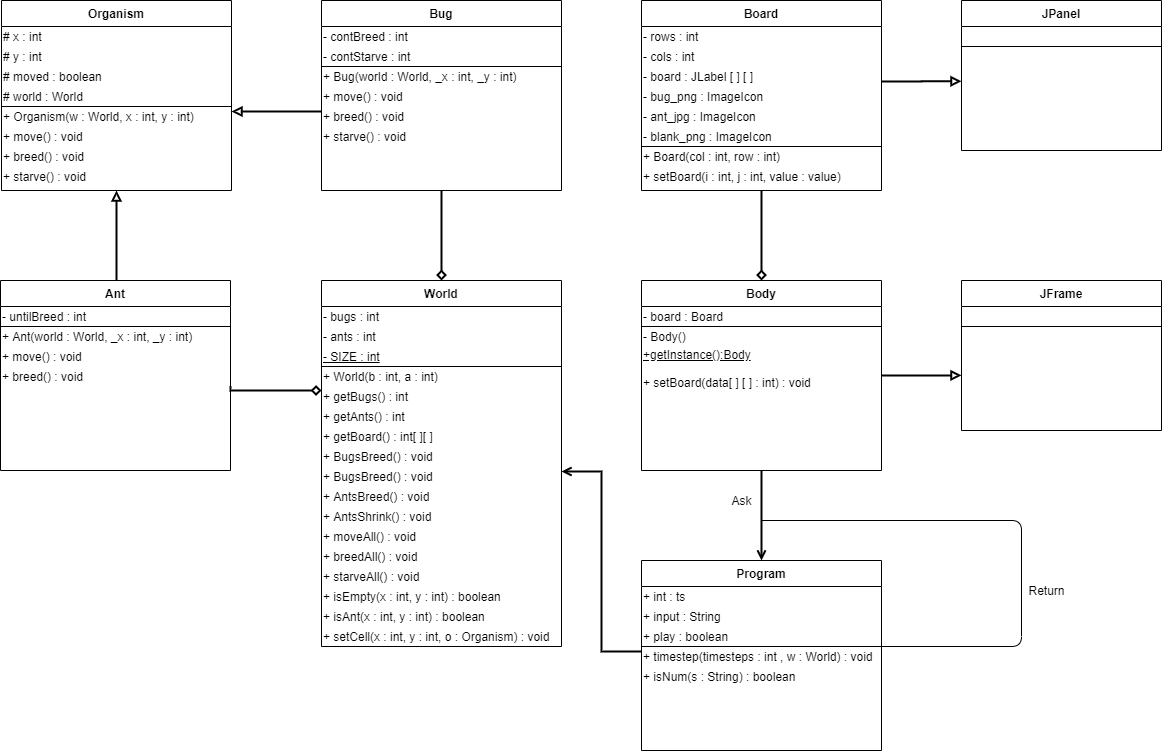
***INTRODUCTION***

As for this project, we are required to simulate a simple 2D simulation of predator(bugs) and prey(ants) behaviours. According to the requirement, ants and bugs live in a grid world of the size 20x20 cells. The grid world is enclosed so the creatures cannot move out of the grid world and only one creature may occupy a cell at a time. The time is simulated in time steps and the creatures perform some action every time step.

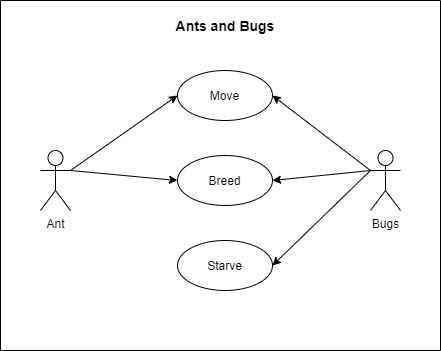
Besides that, the ants should behave according to the following model where every time step it moves to the next cell randomly top, bottom, left, or right direction. If the cell is occupied or out of boundary, then it does not move. In addition, the ants have a function to breed, where a new ant will be created at an adjacent empty cell if it survives at the end of third step. If there is no empty cell available, then no breeding occurs. After that, the ant cannot reproduce until three more time stamps elapsed.

Furthermore, the bugs behave according to the following model where every time step it moves to the next cell randomly top, bottom, left, or right direction. If there is an adjacent cell occupied by an ant, the bug will move to that cell and eat the ant. But a bug is not allowed to eat another bug. The bugs can also breed, and they are allowed to breed eight times. In addition, the bugs has a special function called starve, where if a bug has not eaten an ant in the last three time step moves, it will die at the end of the third time step and it should be removed from the grid.

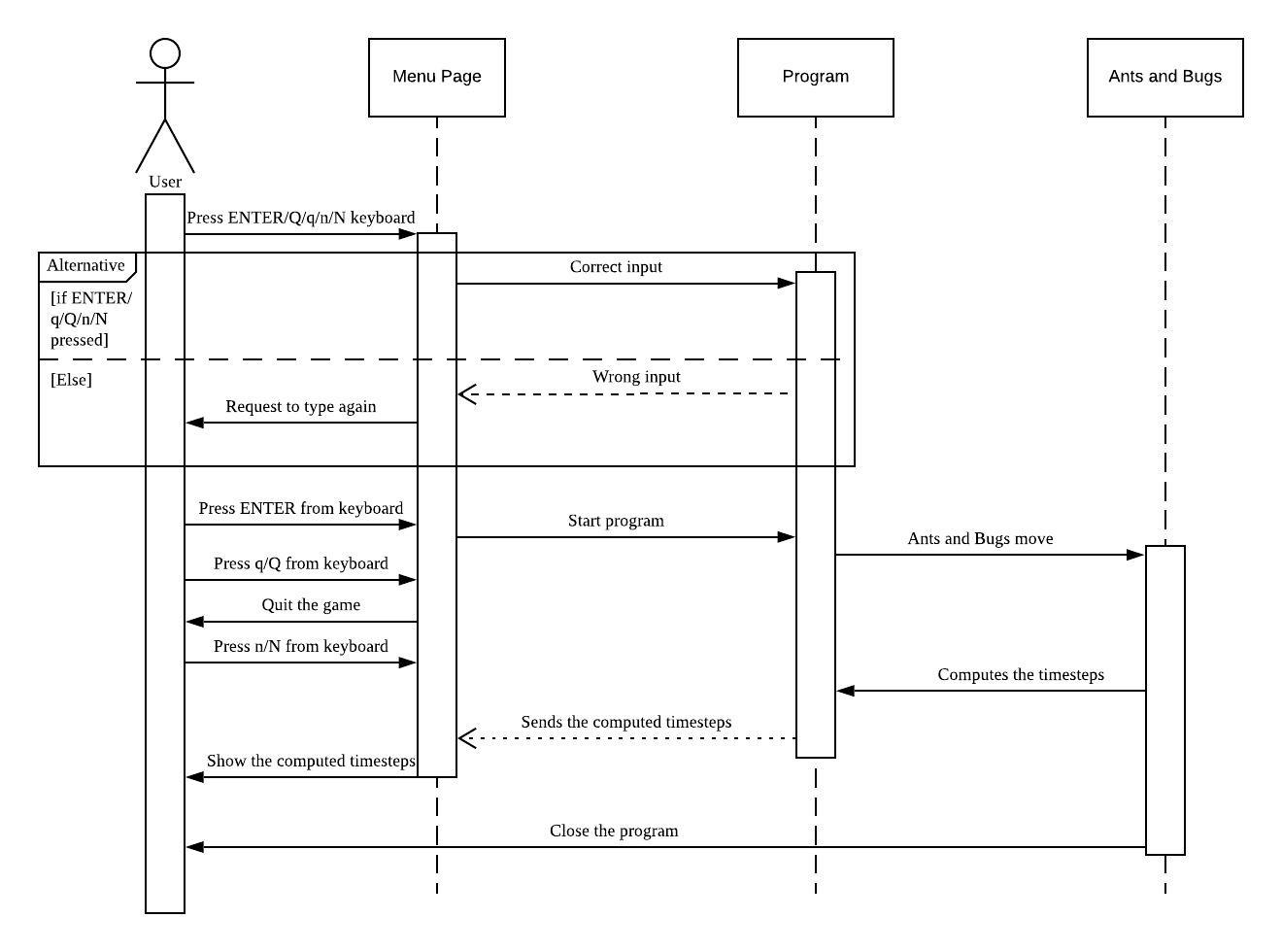
***UML CLASS DIAGRAM***



***USE CASE DIAGRAM***



***SEQUENCE DIAGRAM***

******

***USER DOCUMENTATION***

Inside the program there will be two objects, Ants and Bugs. Here are the representation of Ants and Bugs:

1. **Ants**



*Figure 1.0 - Ant icon used in the simulation.*

1. **Bugs**

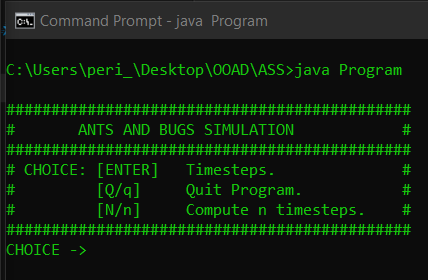


*Figure 1.1 - Bug icon used in the simulation.*

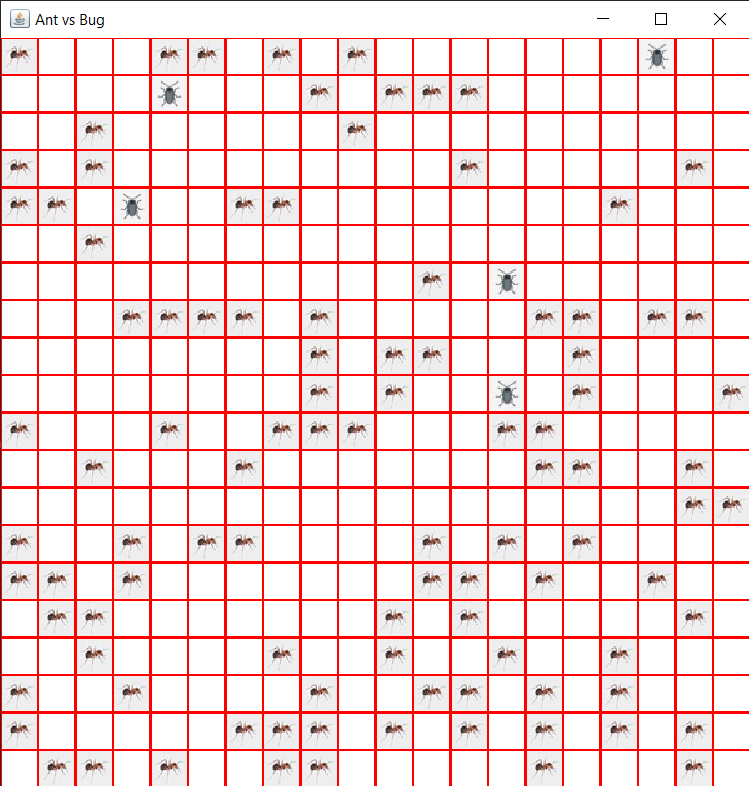
* To play the 2D simulation, the user will have to press **Enter** for the ants and bugs to move. The ants and bugs will move simultaneously.
* For each move, the step time will be recorded in the program.
* In this 2D simulation, inside a 20x20 grid will contain 100 Ants and 5 Bugs.
* Ants will be able to move and breed while Bugs will be able to move, breed and starve.
* When the steps equal to 3, the Ants will breed and for Bugs will starve if it does not eat at that time.
* When the steps equal to 8, Bugs that are still surviving will breed in a new cell adjacent to the survived Bug.
* If the user wants to know the current steps of the Ants and Bugs, the user can press **‘n’** or **‘N’** and it will show the steps in the command prompt.
* If the user wants to quit from the 2D simulation, the user has to press **‘q’** or **‘Q’**. The program will show the Overall Elapsed Time and the Total Computed n Timesteps and the will exit the program.

***SCREENSHOTS OF THE PROGRAM***

**Step 1 :** Firstly**,** launch Program.java, then a menu will pop up together with the Simulation GUI.

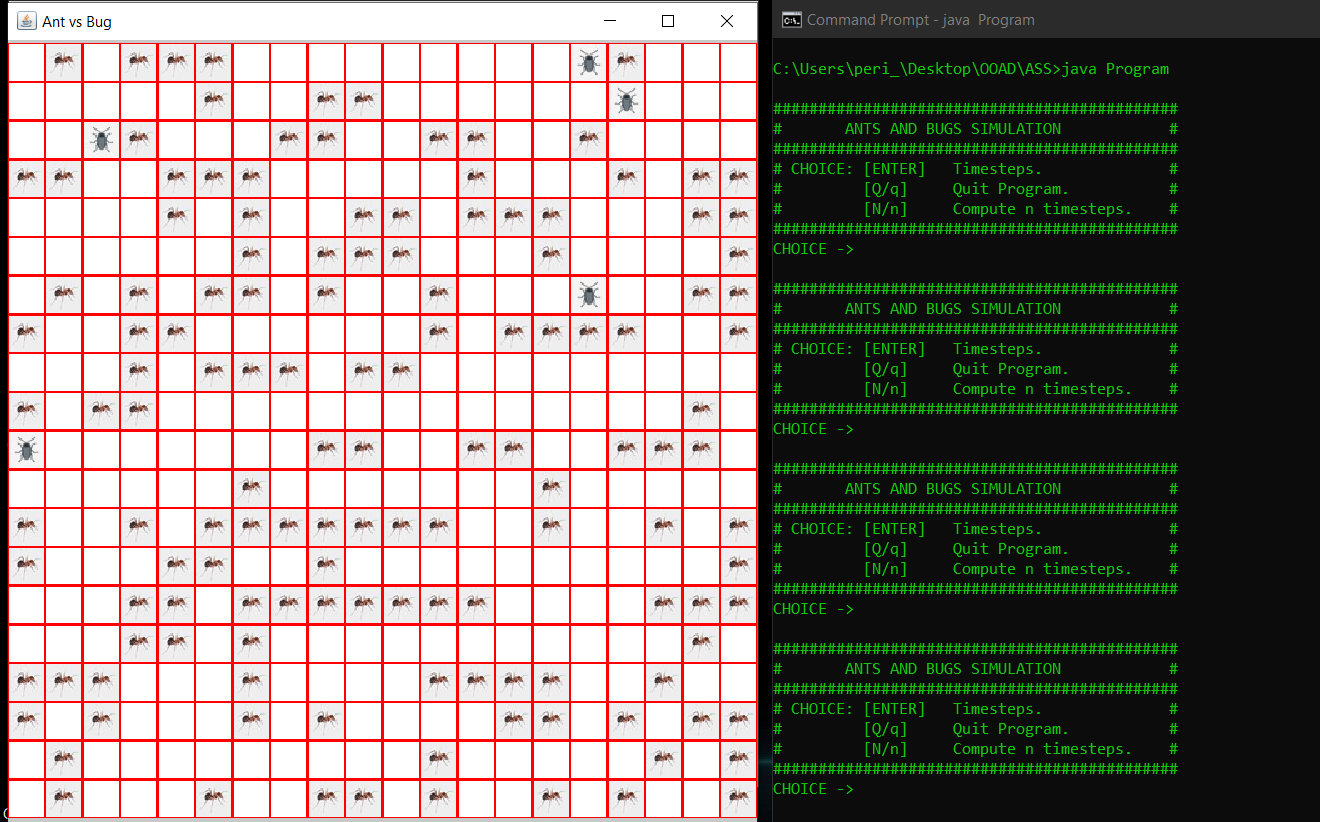
******

*Figure 2.0 - Menu Page*

**

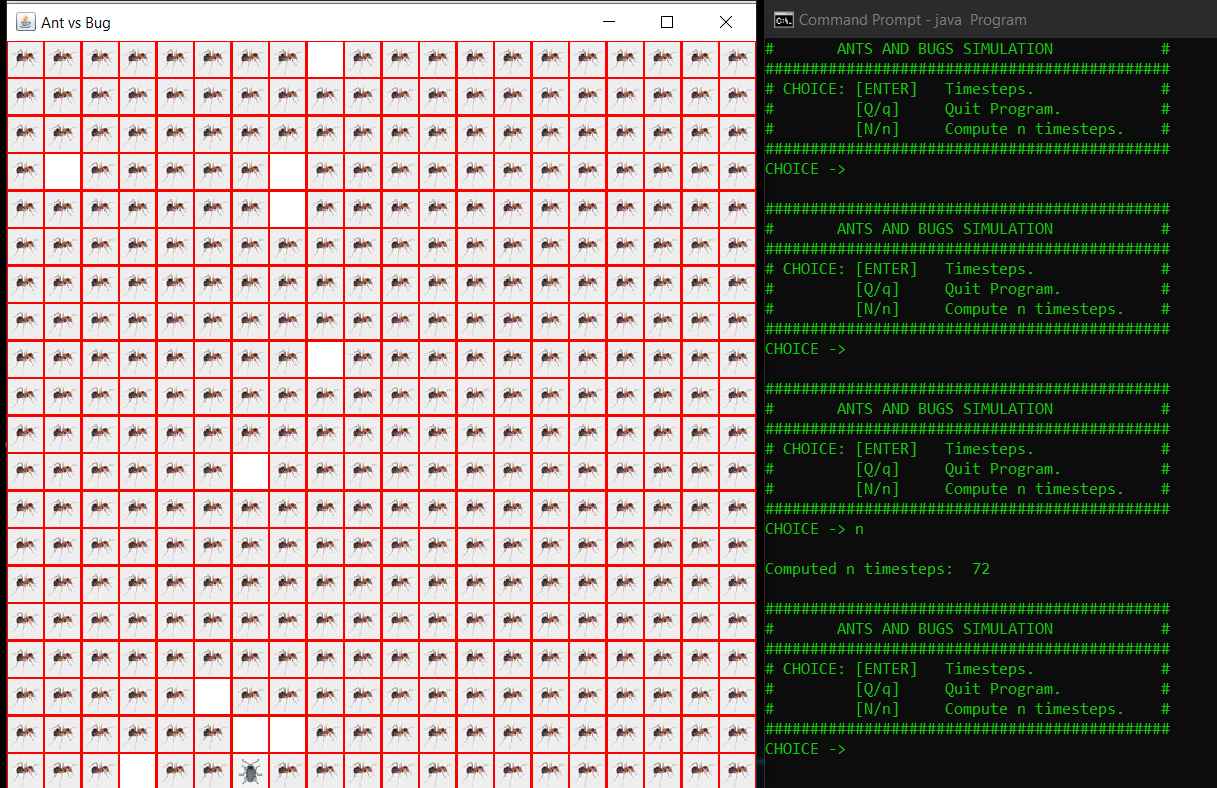
*Figure 2.1 - Output of the Ant VS Bugs Simulation*

**Step 2 :**Secondly, the user needs to click [ENTER] on the command prompt menu, under the choice option.If the user continuously clicks [ENTER], there will be changes in the actual simulation, where the ants and bugs will start moving.

**

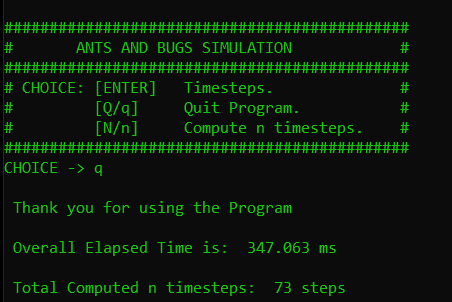
*Figure 2.2 - Sample output after clicking [ENTER], 3 times.*

**Step 3 :** As for the Compute n timesteps,the user needs to choose option [n/N] to get the total steps.



*Figure 2.3 - The N/nth value of clicking [ENTER]*

**Step 4 :** If the user wants to exit the program, then the user needs to choose input [Q/q].The program will calculate the total amount of [N/n] steps and time taken for the program to run.

**

*Figure 2.4 - Computed values for the calculations and time.*