

# ANT COLONY DESIGN DOCUMENT

**Project title:** Ant Colony

**Name:** Charles Hanna

**Course #:** CART 353

**Type of Project:** app

**Project Due date:** April 6, 2014

**Instructor:** Sofian Audrey

## DESCRIPTION

Ant Colony is a digital environment simulating the life of ants. Ants go foraging, and they leave behind them trails of chemicals called pheromones. Once an ant finds food they alert other ants by releasing food pheromones and the other ants will follow the food pheromones until they reach to the food source. The bigger number of ants that will walk on the same trails the longer it will last. Once they find food, they carry them back to their home. The user can interact by selecting food from a food menu and feeding them to the ants by clicking inside the window. The user has to make a conscious choice about which kind of food to feed the ants from. One of the food kinds is a toxic one, and when the ants touch it they will soon die. When an ant dies it release an alert pheromones that will alert other ants about the danger. There are random predators that pop up on the screen in random intervals of time. the bigger the number of ants the more likely the predator will flee, otherwise the predator will eat the ants around it. The objective of this simulation is to show how the human interaction with nature is can impact the environment in different ways.

The open source code for this project is distributed under the GPL licence, which can be redistributed and/or modified under the terms of the GNU General Public License as published by the Free Software Foundation.

## INTENTIONS

Besides my great interest in the complexity of the behaviour of ants and their survival habits, I am very excited to put into practice what I am learning in the Nature of Code, into this project, as it perfectly suits its nature. Lot of new things can be learned about the behaviour of natural organisms and phenomena by simulating them on the screen. While doing my prototype, I was further intrigued to study the details of the ants behaviour so I can better represent it in my virtual colony. For instance I used the `arrive()` method I have seen in the Nature of Code when the ant reaches for food, because I learned, by observation, that sometimes ants don't maintain their same speed when they find food, they slow down and examine it for a while before they dive in. The real satisfying outcome I acquired since starting this project, was the new information I have learned so far, about what has been done earlier to simulate the ants intricate behaviour. I hope that by the time I finish my project it will be easier for me to understand the details of complex simulation algorithms such as the Ant Colony Optimization Algorithm. In addition to the technical knowledge I am hoping to acquire, I intend to communicate an environmental message that projects the effects and consequences of human actions in relation to nature. Polluting the earth with toxic materials is greatly impacting the biodiversity of the natural life on earth, and this can be observed through the rapid extinctions of living species in the shadow of our modern times. This call to awareness will be embodied in my project by giving the user the choice to select what kind of food to feed the ants from. The toxic food will start killing the ants and eventually may poison the whole colony.

# INSPIRATION

My idea for my project was not based on any game or simulation, it was just a curious attempts to simulate the life of ants with the knowledge I am acquiring from the Nature of Code.

However after discussing my idea with my colleagues, Olivier proposed to me to take a look at **Sim Ant** for some inspiration. After watching the game in action on YouTube, I was further curious to download the game and take a look at its code. Unfortunately the source code is not an open one and therefore it was not made available online. So I observed the game features through watching the YouTube video and started to deduce how the functionality has been achieved. While immersed in the guesswork, a brilliant simulation found on open processing.org was proposed to me to take a look at by the Mr. Sofian Audrey, my professor. The latter simulation is called **Ant Colony Pheromone Simulation** by Forrest O. Although the code is a bit unorganized, I actually gained lot of insight from studying the code and commenting almost every line of it in order to better understand it. I also contacted the author and asked him about some details, and he only provided general information. My third inspiration came from a project attempt to make Sim Ant an open source game. This project is called **Open Ant** which source code can be downloaded from Github. Open Ant became a cooperative project where different programmers and designers from around the world are contributing their skills to make this attempt possible. Throughout my research, I learned a lot from the Open Ant community in regards of the organizations to their documentations and tutorials, their resources about ants and ant simulation algorithms, and their cooperative bug fixing attempts and programming preferences. The source code was written with Python, a language that I am not familiar with. Nevertheless, it is an OOP language. In preparation for my final project I am going to attempt to understand the logic behind the game, in the hope of becoming a contributor in the One Ant community.

## Sim Ant

[Download link](#) | [Video link](#)

An ant life simulation game, developed in 1991 by Maxis and designed by Will Wright the creator of "Sim City" and "The Sims".

## Ant Colony Pheromone Simulation

[Download link](#) | [Video link](#)

Ant ant life simulation links that shows how ants interact with each others using pheromones. This simulation was designed by Forrest O. and posted on openprocessing.org

## Open Ant

[Download & Documentation link](#) | [Reddit Community link](#)

After trying many UML softwares I landed on [Unimozzer](#), which is a Java powered, open source UML software, that allows the user to check the Java code of the diagram formation, where the code can be changed and/or imported and the effect will take place directly on the diagram. As well as it automatically links the classes that have common relations together.

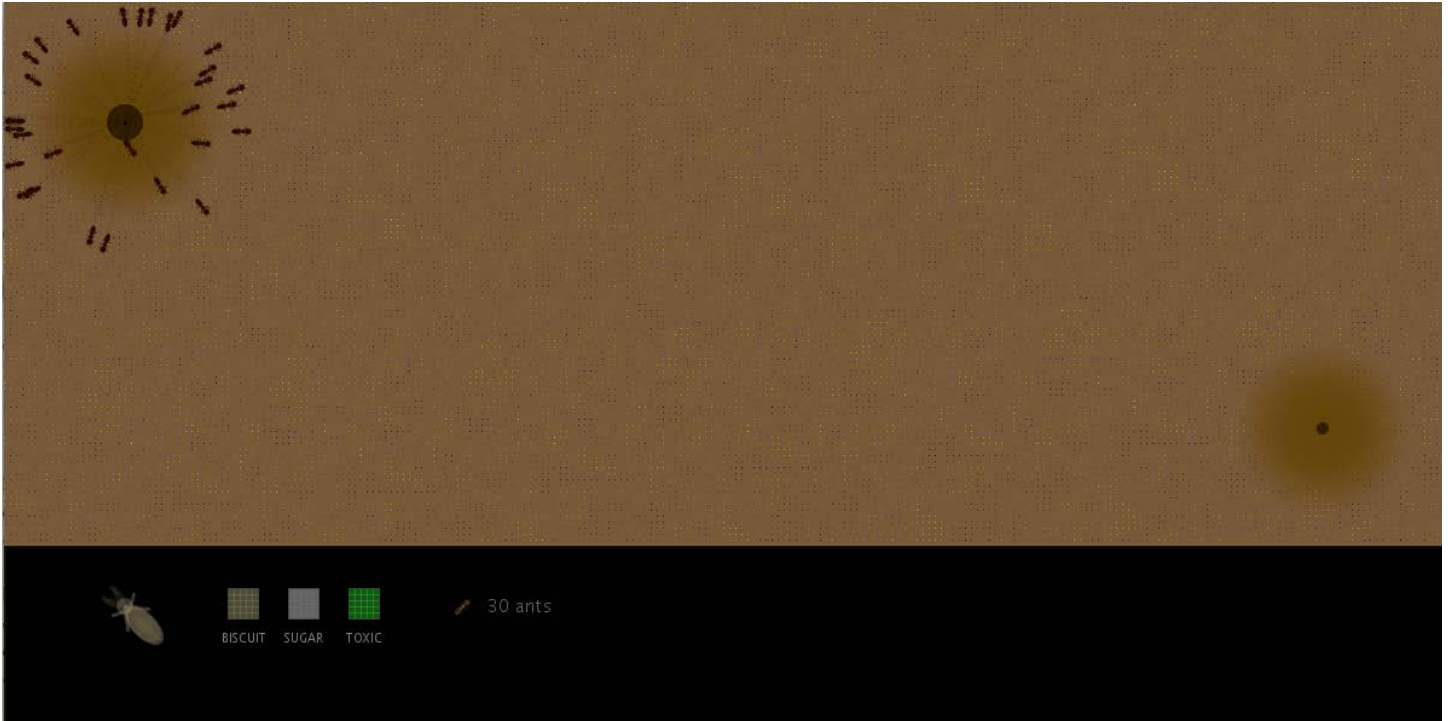
[illegible]

Click on image to open it.

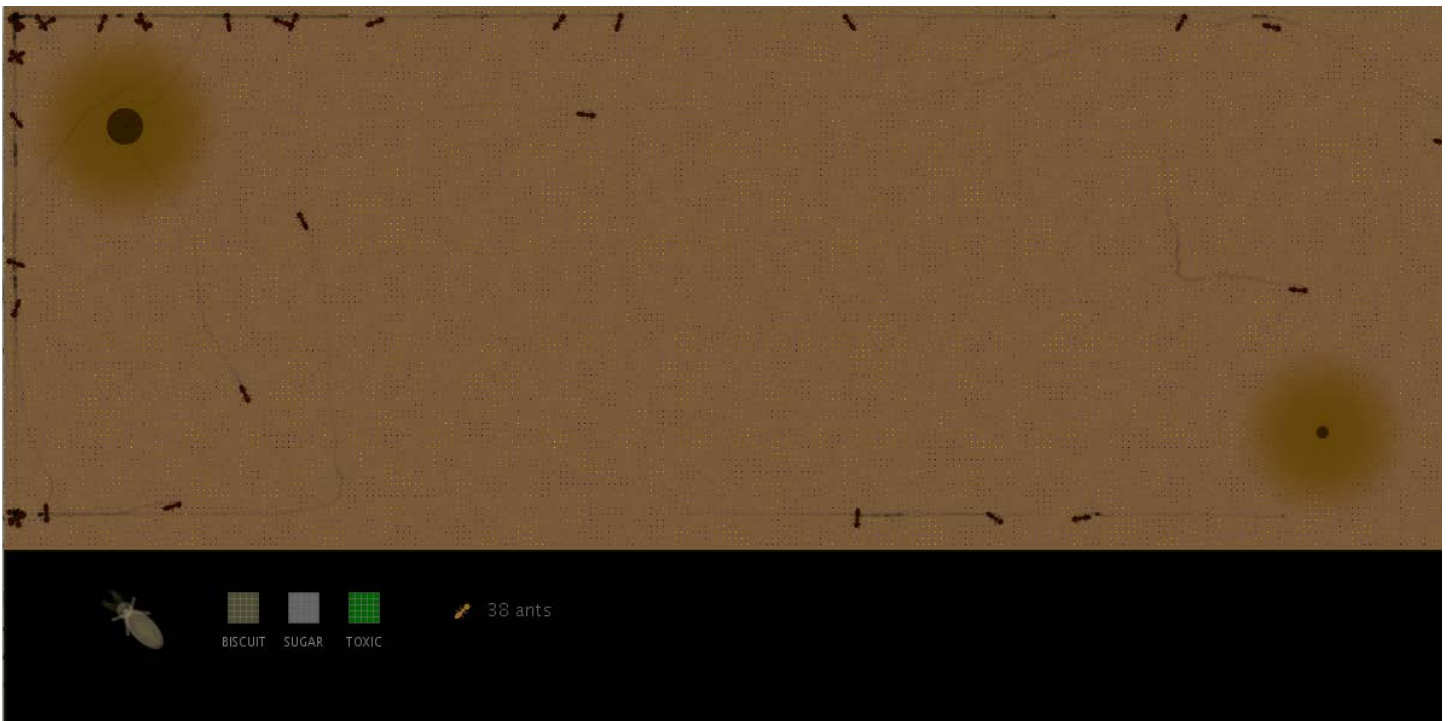
## FEATURES / ROADMAP

Final feature / task		Current status	Completion (%)	Time (h)	
				Planned	Spent
Ant class		done	100%	8	12
Colony class		done	100%	2	1
FoodSource		done	100%	4	4
Food class + 3 subclasses		done	100%	4	6
Food class	Biscuits subclass	done	100%	1	1
	Sugar subclass	done	100%	1	1
	Toxic subcalss	done	100%	1	1
Pheromone class	PheromoneHome subclass	done	100%	4	4
	PheromoneFood subclass	done	100%	4	4
Home class	PredatorHome subclass	done	100%	2	1
Land class		done	100%	1	1
Predator class		done	100%	8	8
Ant modeling		done	100%	8	8
Soil modeling		done	100%	2	2
Predator modeling		done	100%	8	8
Clickable food menu		done	100%	12	15
Add food on mouse click		done	100%	4	5
Commenting		done	100%	8	8
Total				80	80
Total of hours remaining for completion				0	

## SNAPSHOTS



Ants are automatically released from their nest when the program starts. The user can add more ants to the scene from the menu.



They start looking for food leaving behind home Pheromones, once they find food they leave behind the food pheromones letting other ants know about the food source.



## SNAPSHOTS

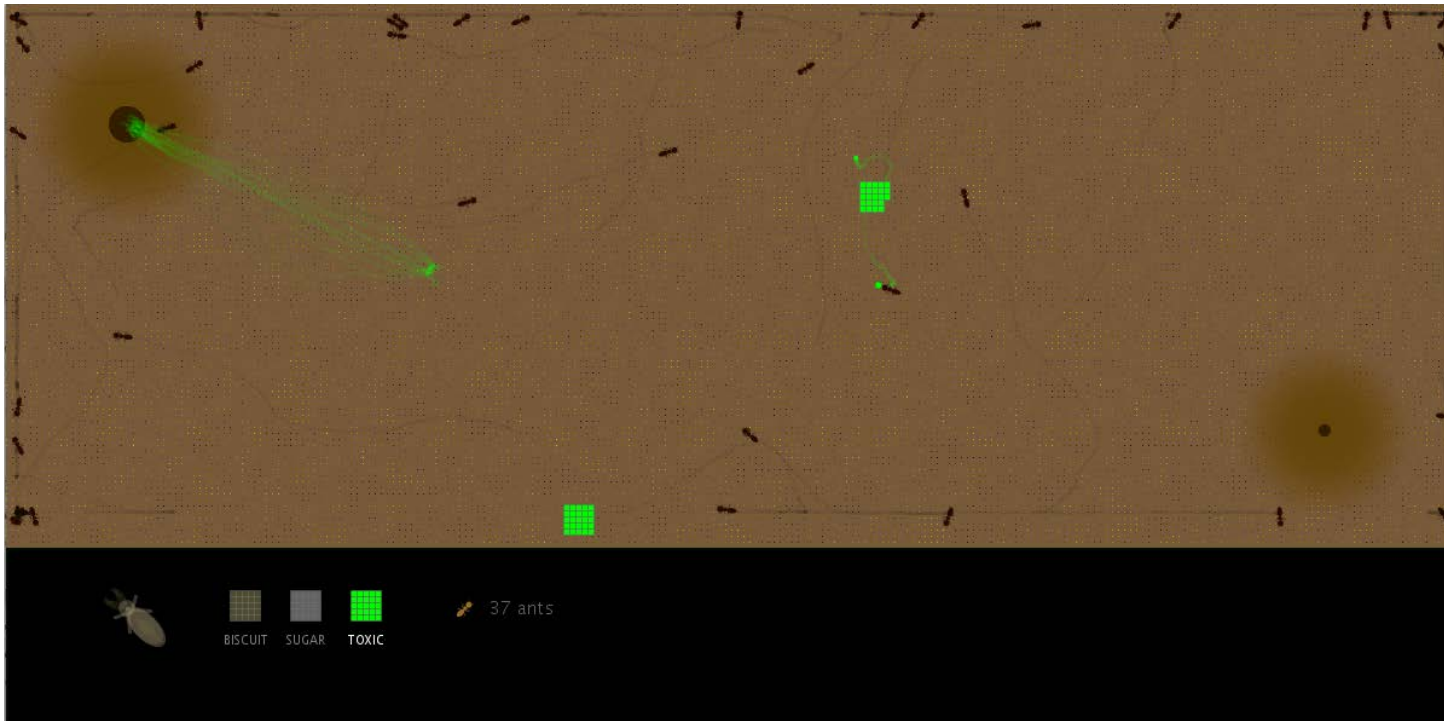


They are attracted to sugar more than they are attracted to any other kind of food.

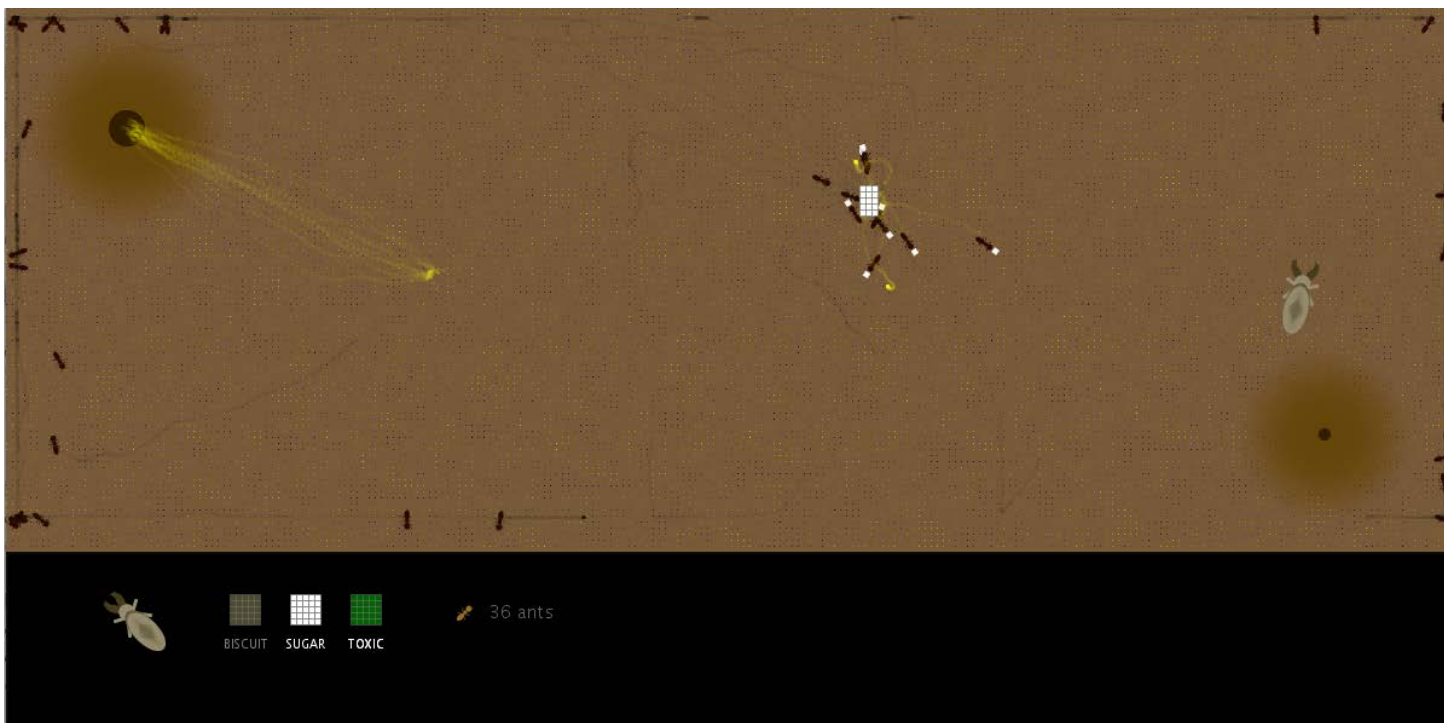


Once they pick up food they wonder around, if they approach their nest they go directly to it.

## SNAPSHOTS



When the toxic food is fed to the ants they start doing a weird behaviour and they die.



The user can release the Antlion, which will follow the ants to eat them. The ants will flee, but if the number of ants around the antlion is greater than 5 they will attack it and it will flee.

## SNAPSHOTS



The antlion sets a trap for the ants at its nest. If an ant enters its nest it will be stuck, and the ant lion will eat it.



## Sketches and Screen Captures

