Operation Fallout – User Guide

# How to Play

You are a member of the Russian government with unlimited access to nukes and animals. Your government want you to explore the potential mutations of animals should a nuclear apocalypse happen.

The goal of the game is to create as many mutations as possible, the possibilities are endless!

# Features

## Add

You are able to add creatures such as ants, cows and goats to your simulation. You are also able to use nuclear bombs to add radiation to the area. The animals you add have magical blast resistance, and therefore cannot die to a direct hit.

## Generate

As an actual nuclear apocalypse is very random, you are able to generate a new map at any time.

# Limitations

There are a few limitations to our program. First of all, not all mutations are mapped out, reducing possible creativity. Secondly, the graphics may not accurately represent real world situations. Lastly, as the game is resource intensive, the minimum requirements may be beyond what a typical school computer can handle. (One thread on an i7-860 overclocked to 4.0 Ghz and 1.5GB of RAM)

Operation Fallout – Developer’s Guide

# Overview

Operation Fallout depends on three packages, map, object, and tile. Object contains the implementations of all the animals used in the simulation, it has a separate implementation for each animal. The tile package contains the available information of each tile and any effects acting upon it (such as radiation). The map package ties everything together by performing the actual simulation and displaying the results.

## Object

The Object package consists of definitions for all objects and their behaviours. Room was left for future expansion by making all classes highly modular. Each object has three methods that must be invoked each tick. Survive, determining whether it survives. Mutate, whether it mutates. And an action, usually reproduction. It stores an objects properties, such as the amount of radiation it has, its combat power, and so forth. It also has containers for any effects that may affect the object, however, it is not currently implemented.

## Tile

The tile package contains the definition for tiles. It has containers for objects on a tile, the tile’s attributes and any effects acting upon the tile. Specifically, it calculates the environment factor, which partially determines if an animal will survive a tick on tick.

## Map

The map package ties everything together by providing implementations for the actual simulation including all mutations. It also implements ActionListener and MouseListener for additional user interactions. With each tick, all effects are first processed, then all objects. It also provides for selection and adding objects to the simulation.

# Suggestions for Improvement

There are many possible improvements for our game. The addition of artificial intelligence could mean that animals may move with a predictable fashion. Similarly, an accurately generated or pre-set map may ensure greater accuracy. Limiting what animals can exist on what type of terrain may also improve the project. The UI can also be improved to enhance user-friendliness. Lastly, the program may be optimized to reduce load times.