Term Project

CSC 413 SPRING 2018

SAN FRANCISCO STATE UNIVERSITY

Introduction

For the final assignment in CSC 413, everyone will be completing a term project with ***1*** partner. This project will consist of two 2D games written in Java. For the first game, every team ***MUST*** implement the Tank Wars game. This is non-negotiable. For the second game, teams may select from a list of second game options. However, if a team wishes to do something different they may request so by emailing the instructor. More about this process later. Choosing something other than the games listed here has its disadvantages.

The goal of this term project is to practice good OOP. The intended purpose of these two games, is for teams to implement the tank game, and then re-use a good portion of the first game to write the second game. This means the design and implementation of the first game must practice good OOP. REUSUABBILITY will be the focus of this term project. A more reusable game one will make implementing game two easier (relatively speaking). This also means choosing a different idea then listed in the second game options section will break this, requiring your team to implement two entirely different games/code bases.

A chapter from Game Maker has been provided to get you started on implementing the first game. NOTE we are not writing these games in game maker, therefore you must convert the steps into Java code. There will also be resources that you may use for your games. There is a set for the tank game as well as one for each of the second game choices.

The following document will outline all the information needed for completing the term project. Please read this document very carefully. Mistakes made while implementing or handling submissions will cause grade penalties.

Team selection

The term project is a pair project. This means you will need to find a partner to work with. You ***MUST*** work with someone in the same section you are in enrolled in. Section hopping is ***NOT ALLOWED.***

To let the instructor know who you would like to work with, please fill out the google form at the link given below. You have until ***Friday, March 16, 2018 @ midnight*** to fill out this form. Teams will be organized on ***Saturday March 17, 2018***.

- If you do not fill out the form by the deadline given above, you will be randomly assigned a partner
- If you do not care who you work with, ***PLEASE DO NOT FILL OUT THE FORM***
- If you cannot find a partner, this is OK, ***PLEASE DO NOT FILL OUT THE FORM*** and a partner will be assigned to you.

The google form can be found at the link on ilearn under the Term project section.

Requirements

Below is the list of requirements for completing the term project. All steps must be completed.

1. Each student shall have a partner to work with.
2. Each team shall correctly set up the GitHub repo according to the instructions given in the GitHub Repo section.
3. Each team shall complete a class diagram for the Tank game
4. Each team shall implement the Tank Game
5. Each team shall complete a class diagram for the Second game
6. Each team shall implement the Second Game Chosen.
7. Each team shall write documentation for both games (1 document for both games).
8. Each team shall give a 10-15-minute presentation.

Requirements 4,6,7 and 8 will make up your term project score.

However, points will be deducted from mishandling submissions or an incorrectly configured GitHub repository.

GitHub Repository

When your team is formed, an email will be sent to you notifying or confirming who you are partnered with. This email will also contain a link to create the GitHub repository. To correctly configure your repository, you must do the following steps.

1. Select one team member to click the project link in the email to create the repository. ONLY 1 TEAM MEMBER SHOULD DO THIS. Creating more than 1 repo will cause the other 1 to be deleted. I will not inspect these repos, I will simply delete the second one created.
2. The team member that created the repository MUST change the name of the repo to the following format:

   csc413-tankgame-TeamNN for the Tank game.

   csc413-secondgame-TeamNN for the second game.

   where NN is your team number, teams with numbers less than 10 MUST pad their numbers with 0's. For example, for team one it would be Team01 NOT Team1. ***Failure to correctly name your repo will cause a 10-point penalty.***

3. The repository creator then MUST ADD his/her teammate to the repository. This can be done in the settings tab of the repository.

At this point your repository should be setup correctly and your team may begin storing code in your repository.

Milestones

Milestone One – Due Saturday March 24th, 2018 @ midnight

Milestone one requires no coding. For this milestone, you will create a class diagram for the tank game. Some sample code has been given on ilearn. You can use this code but DO NOT COPY it verbatim. The class diagram does not need to be 100% perfect or complete. It is not expected that each team know the structure of their game this early on. The class diagram should be digitally drawn. Hand drawn submissions are OK but not advised. If you spend the time now drawing the diagrams correctly, you save time when you need to fill them in when completing final documentation. You can use sites like https://www.draw.io/ to draw your diagrams.

The class diagram does not need to list any members (variables and methods). Only classes, interfaces and abstract classes need to be drawn.

Please submit your class diagram to ilearn as an image or as a PDF.

NO WORD DOC OR ANY OTHER WORD PROCESSOR SUBMISSIONS ARE ALLOWED. FAILURE TO SUBMIT THE CORRECT FILE TYPE FOR THE CLASS DIAGRAM WILL RESULT IN A 10-POINT PENALTY.

Milestone Two – Due Saturday April 21st, 2018 @ midnight

Milestone two will require you and your partner to implement the Tank Wars Game in Java. You may use the sample code posted on ilearn to help you get started. There are resources posted on ilearn for the Tank Wars Game. These contain 2D sprites and sounds. You may also use your own resources as well.

Please follow the instructions in the GitHub Repository section when setting up the repository for your team. ***REMEMBER: Failure to correctly name/setup your repo will cause a 10-point penalty***

The Tank Wars Game has the following requirements:

1. Tank Game shall have 2 Players
2. Tank Game shall have tanks that move forwards and backwards
3. Tank Game Shall have tanks that rotate so they can move in all directions
4. Tank Game shall have a split screen.
5. Tank Game shall have a mini-map
6. Tank Game shall have health bars for each tank
7. Tank Game shall have lives count (how many lives left before game over) for each tank
8. Tank Game shall have power up (these are items that can be picked up to modify your tank. What these power ups are, is up to you as a team).
9. Tank game shall have unbreakable walls
10. Tank game shall have breakable walls
11. Tank Game shall have tanks that can shoot bullets that collide with walls
12. Tank Game shall have tanks that can shoot bullets that collide with other tanks.
13. Tank Game shall come with a brief readme.txt file that contains what IDE the project was made in, how to run the game and the controls for playing the game.

When submitting the tank game, make sure ALL code and readme.txt file is contained in the repo you made. The late deadline for the tank game is determined by the timestamp of the last commit on the ***MASTER*** branch. No other branches will be checked. ***YOU HAVE BEEN WARNED. NO LATE SUBMISSIONS***

Milestone Three – Due Wednesday April 25th, 2018 @ midnight

Milestone two requires no coding. For this milestone, you will create a class diagram for your second game choice. The class diagram does not need to be 100% perfect or complete. It is not expected that each team know the structure of their game this early on. The class diagram should be digitally drawn. Hand drawn submissions are OK but not advised. If you spend the time now drawing the diagrams correctly, you save time when you need to fill them in for the final documentation. You can use sites like https://www.draw.io/ to draw your diagrams.

The class diagrams do not need to list any members (variables and methods). Only classes, interfaces and abstract classes need to be drawn.

Please submit your class diagram to ilearn as an image or as a PDF.

Milestone Four – Presentations

For milestone four your team will be required to give a short 10 to 15-minute presentation on your term project. During this presentation you will present either your tank game or your second-choice game OR both. Either option is fine. The presentation does not require slides, but you may write some up.

A grading rubric will be posted on ilearn so you know what to expect during your presentation.

You will have AT MOST 15 minutes for your presentation. This time also includes setup (connecting your laptop to the projector).

Note that when you present I WILL NOT BE GRADING THE GAMES, but instead grading your presentation skills. Presenting Games not fully completed is OK.

Presentation Dates:

Section 03 – MW 4:10-5:35	Section 02 T/TH 7:00-8:15
May 7th	May 8th
May 9th	May 10th
May 14th	May 15th
May 16th	May 17th

Milestone Five – Due Wednesday May 23rd, 2018 @ midnight

Milestone five is going to require your team to implement your team's second game choice. Given that you have a completed tank game, implementing the second game should take less time. This is only true if you can reuse some of the code from the Tank game provided it was designed correctly.

For the second game you will need a new repository for the code you will write. A new email will be sent out before you finish Milestone Three. This email will be like the first one you received for the tank game, but the repo link will be different. Please follow the same instructions you followed for the tank game repo for this repo. ***Again, failure to correctly setup your repository will cause a 10-point penalty***.

There are no listed requirements for the second choice. But you do need to follow the rules of the game outlined the in Second Game Options section.

When submitting the second game, make sure ALL code and readme.txt file is contained in the repo you made. The late deadline for the second game is determined by the timestamp of the last commit on the ***MASTER*** branch. No other branches will be checked. ***YOU HAVE BEEN WARNED. NO LATE SUBMISSIONS***

Milestone Six – Due Friday May 25th, 2018 @ midnight

Milestone six will require your team to write a term project documentation. This document will cover the entire term project from beginning to end. There only needs to be ***one document for both games.*** Please ***DO NOT*** submit 1 document for each game.

This document needs to be written by both team members. However, towards the end of the term project it may be in your team's interest to have one team member start working on documentation while the other finishes all the small things in the second game.

Your documentation SHALL contain the following sections:

1. Title page containing
   a. Team number
   b. Team members
   c. Class, Semester
   d. A Link to ***BOTH*** repositories.
2. Introduction
   a. Project Overview
   b. Introduction of the Tank game (general idea)
   c. Introduction of the Second game (general Idea)
3. Development environment.
   a. Version of Java Used
   b. IDE Used
   c. Any special libraries used or special resources and where you got them from.
4. How to build or import your game in the IDE you used.
   a. Note saying things like hit the play button and/or click import project is not sufficient. You need to explain how to import and/or build the game.
5. How to run your game. As well as the rules and controls of the game.
6. Assumptions Made when designing and implementing Game
7. Tank Game Class Diagram
8. Second Game Class Diagram
9. Class descriptions of all classes shared among both Games
10. Class Descriptions of classes specific to Tank Game
11. Class Descriptions of classes specific to Second Game
12. Team-reflection on Development process during the term project
13. Summary of work done by each team member.
    a. Please be honest here
14. Project Conclusion.

When completing this documentation please make sure you are concise but that each section contains enough information. Point penalties will be added for insufficient information or missing sections.

The final documentation ***SHALL*** be submitted in PDF FORM. Please submit your final documentation in PDF form to ilearn by the deadline at the heading of this section (note the repetition, must be important). ***Submitting documentation that is not a PDF will cause a 30-point penalty.***

THERE WILL BE NO LATE SUBMISSIONS.

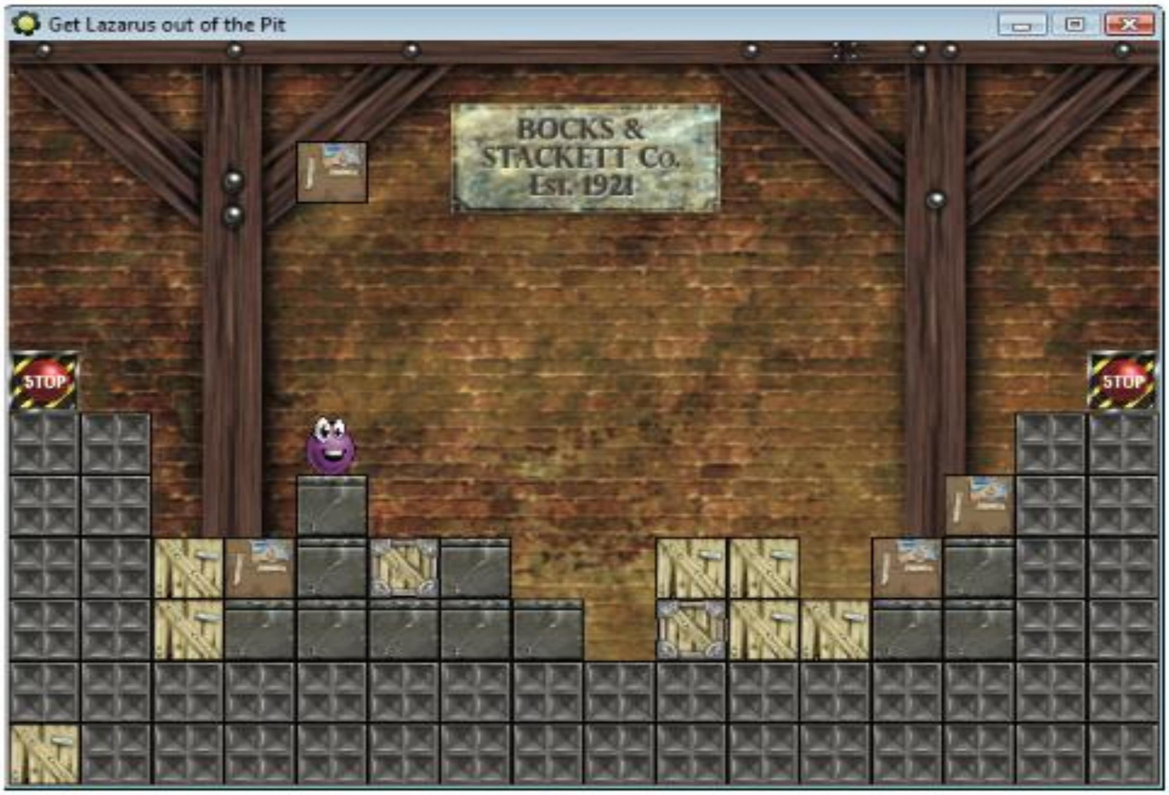
Second Game Options.

Lazarus

Lazarus has been abducted by the Blob Mob, who are intent on bringing this harmless creature to a sticky end. They've imprisoned him at the Blobfather's (sorry) factory, where they are trying to squish him under a pile of heavy boxes. However, they've not accounted for Lazarus's quick thinking, as the boxes can be used to build a stairway up to the power button that halts the machinery. Do you have the reactions needed to help Lazarus build a way up, or will the evil mob claim one more innocent victim?

Each level traps Lazarus in a pit of boxes stacked up on either side of the screen to contain him within the level. The arrow keys will move Lazarus left and right, and he will automatically jump onto boxes that are in his way. However, he can only jump the height of a single box, and stacks two or more boxes high will block his path. New boxes will periodically appear directly above Lazarus's current position and fall vertically down from the top of the screen until they come to rest. This means that the player will be able to use Lazarus's position to control where boxes fall and build a stairway up to the power button.

There will be four different types of boxes, increasing in weight and strength: cardboard, wood, metal, and stone. Falling boxes will come to rest on boxes that are stronger than them, but will crush boxes that are lighter. The type of each box is chosen at random, but the next box will be shown in the bottom-left corner of the window just before it appears. There will be a number of increasingly difficult levels, with higher stairways to build, and boxes that fall faster. When Lazarus gets squished, the level will restart to give the player another try. See Figure 4-1 for an example of how a level will look.



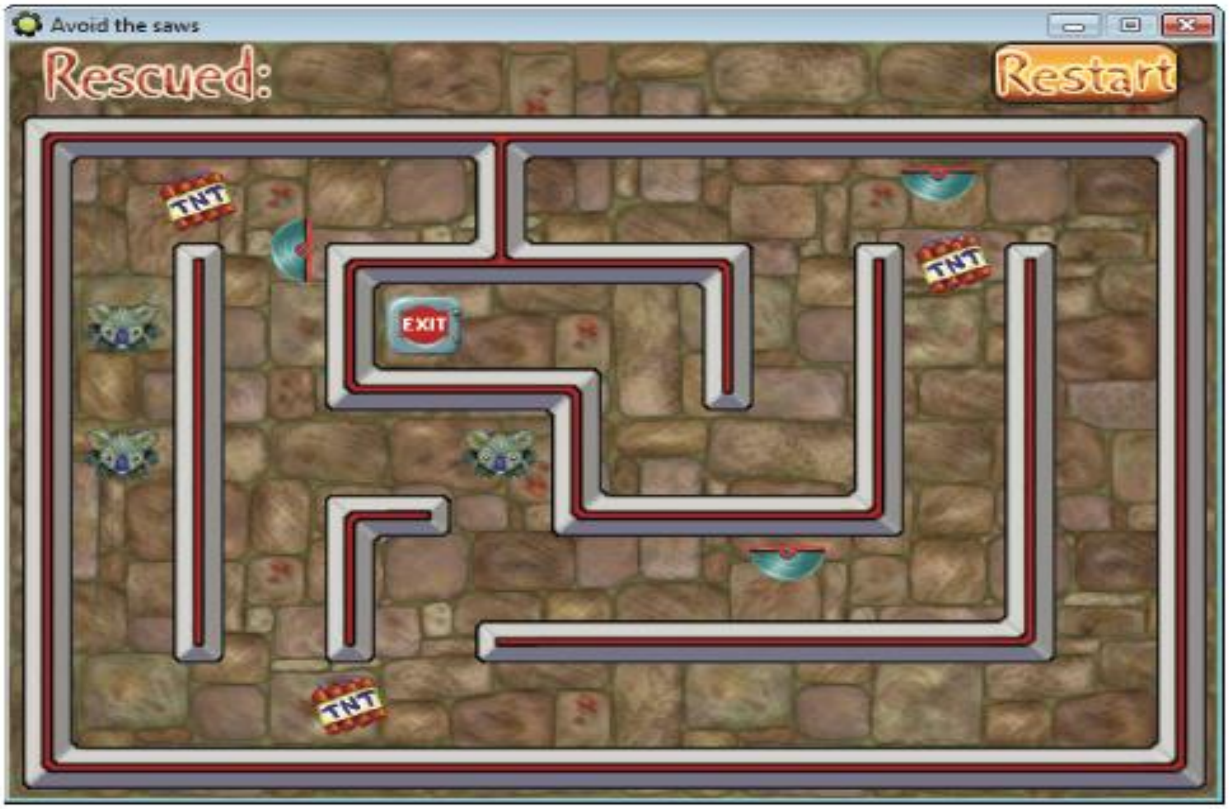
Koalabr8

A colony of koala bears have been captured by the evil Dr. Bruce for use in his abominable experiments. The koalas manage to escape from their cages only to find that the doctor has implanted some kind of mind control device in their brains. The only way they can overpower the controlling effect is to combine their thoughts and all perform the same actions at once. The koalas must work together to find their way past the many dangers in the doctor's laboratory and escape to freedom.

The arrow keys will simultaneously move all of the bears on a level, except bears whose paths are blocked by a wall or another bear. Each level will be a hazardous maze that is completed by getting all of the koalas to an exit. However, if a koala touches a dangerous hazard on the way, then he dies and the level must be replayed. The game will contain a number of fatal and nonfatal hazards shown in the following feature list:

- Fatal hazards
  - ExplosiveTNT
  - Moving circular saws

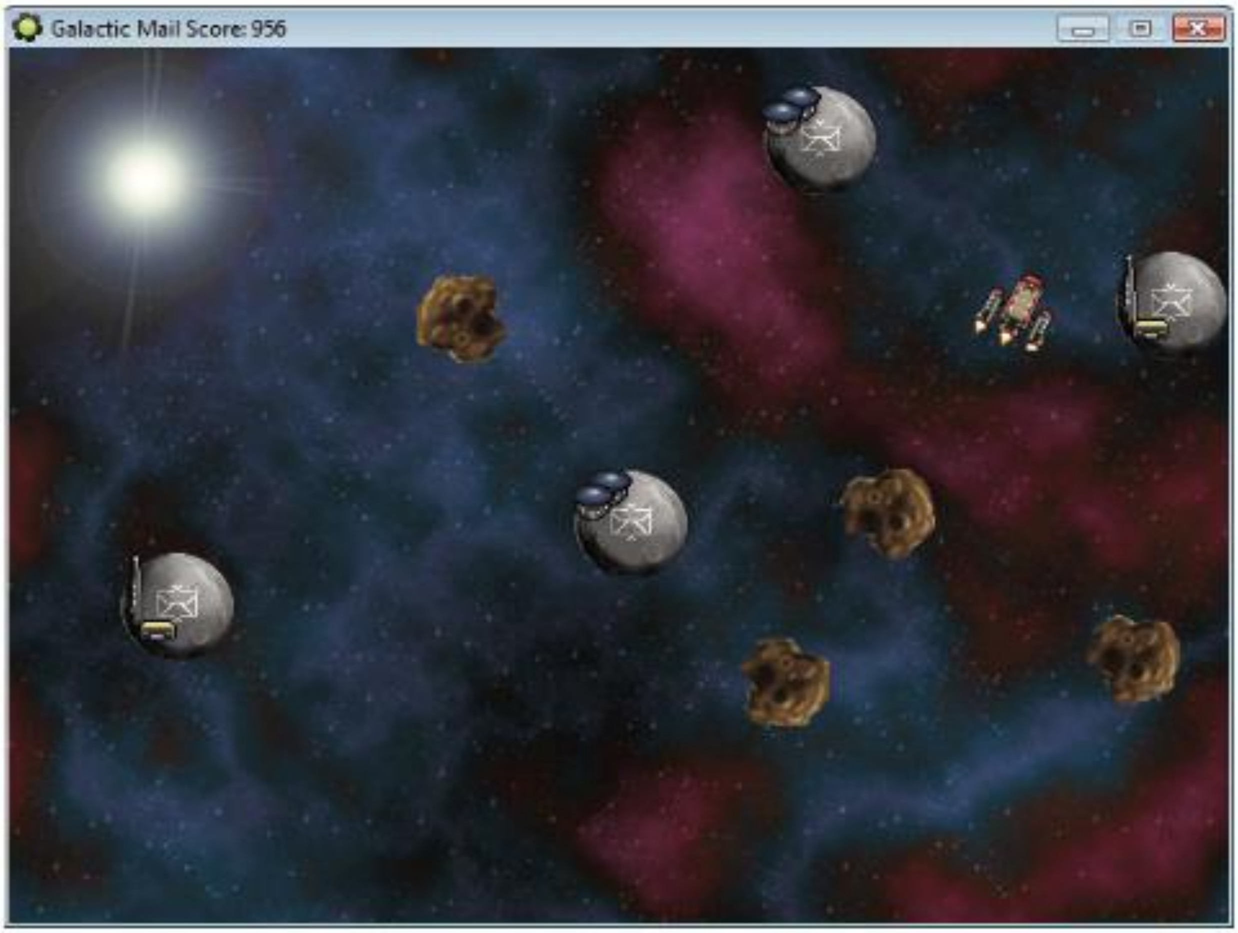
- Non fatal hazards
  - Red exits—Allow any number of koalas to exit the level
  - Blue exits—Allow a single koala to exit the level
  - Locks—Block the path of koalas(red, blue, and green)
  - Switches—Open locked passageways (normal, timed, and pressure)
  - Boulders—Can be pushed by koalas and destroy other hazards



Galactic Mail

You play an intergalactic mail carrier who must deliver mail to a number of inhabited moons. He must safely steer a course from moon to moon while avoiding dangerous asteroids. The mail carrier is paid for each delivery he makes, but pay is deducted for time spent hanging around on moons. This adds pressure to the difficult task of orienting his rickety, old rocket, which he cannot steer very well in space.

When the rocket is on a moon, the arrow keys will rotate it to allow the launch direction to be set. The spacebar will launch the rocket, and the moon will be removed from the screen to show that its mail has been delivered. In flight, the rocket will keep moving in the direction it is pointing in, with only a limited amount of control over its steering using the arrow keys. When things move outside the playing area, they reappear on the other side to give the impression of a continuous world. The player will gain points for delivering mail, but points will be deducted while waiting on a moon. This will encourage the player to move as quickly as possible from moon to moon. There will be different levels, with more asteroids to avoid. The game is over if the rocket is hit by an asteroid, and a high-score table will be displayed. Figure 3-1 shows an impression of what the final game will look like.

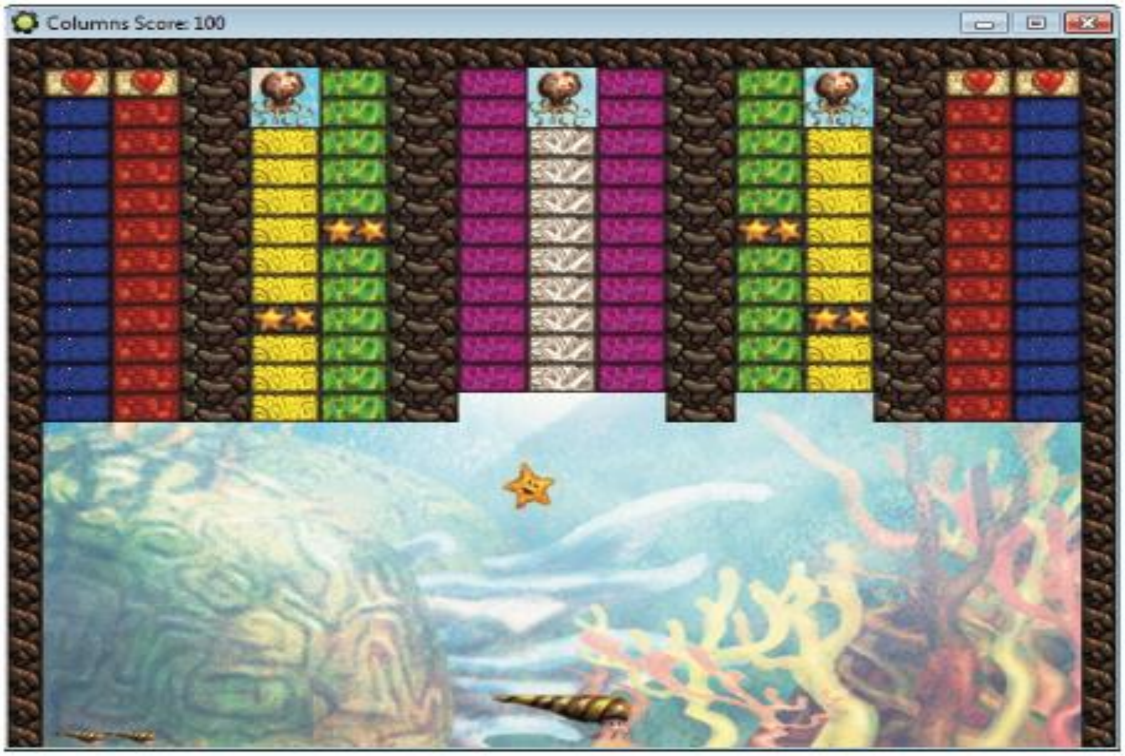


Super Rainbow Reef

The monstrous Biglegs have driven the peace-loving creatures of Rainbow Reef from their ancestral homes. Despite their inexperience in the ways of war, Pop and Katch have invented a way of combining their skills to fight back against the Biglegs. For this incredible feat, Pop must bounce from Katch’s shell to attack the evil invaders. Katch must then move quickly to save Pop from plummeting into the deep waters below. The cowardly Biglegs often retreat behind coral defenses, so our heroes must be prepared to smash their way through if they are to finally drive the Biglegs from Rainbow Reef!

There will be no direct control over Pop’s movement, and he’ll bounce freely around a play- ing area enclosed by walls on all sides except the base. The left and right arrow keys will move Katch horizontally along the base in order to bounce Pop from Katch’s shell and stop him from falling out of the level. The collision point along Katch’s shell will determine the direction of Pop’s bounce, and so allow the player to control his movement. Bounces toward the left will send Pop left and bounces toward the right will send him right. Pop’s movement is also affected by gravity, and each time he collides with Katch, he gets slightly faster so that the game becomes increasingly difficult.

The game will have several levels, each containing a number of Biglegs that Pop must col- lide with in order to complete the level. Most levels will also contain coral block defenses, which must be knocked out of the way in order to reach the Biglegs. Breaking blocks will score extra points and special blocks give the player extra rewards, but they don’t have to be destroyed to finish a level. If Pop leaves the screen, the player loses a life and Pop is brought back into play. Once three lives have been lost, the game ends and a high-score table is displayed.



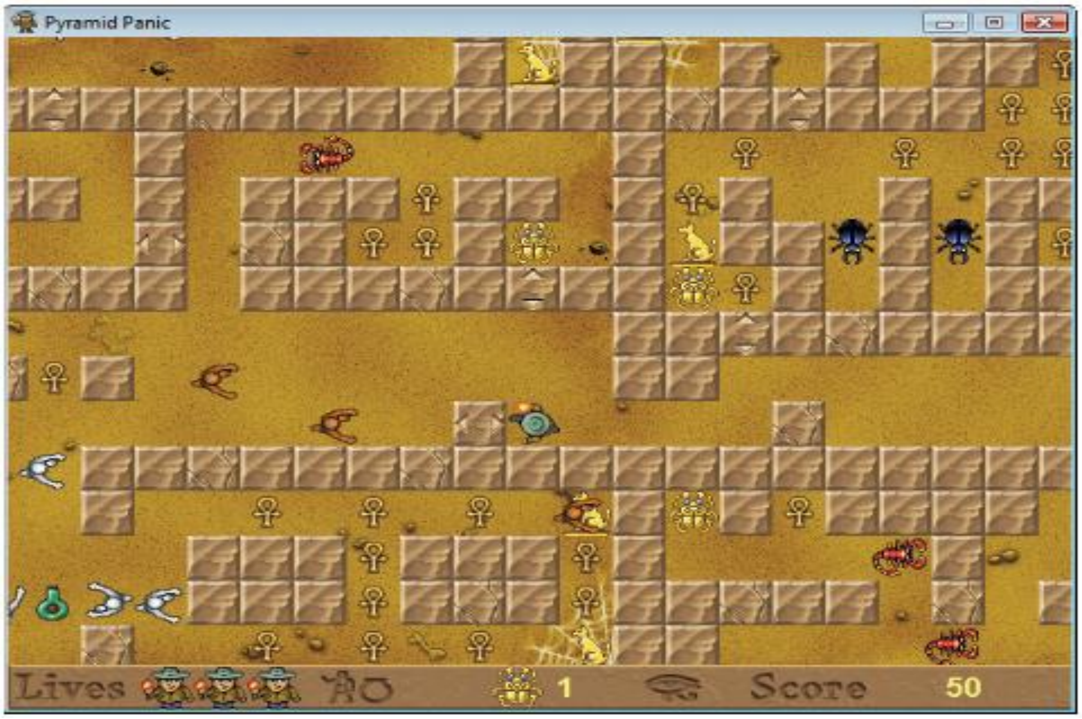
Pyramid Panic

You play an explorer (see Figure 14-1) who has become trapped while investigating a large pyramid complex. All around lie the treasures of an ancient pharaoh, but pyramids are hazardous places and danger lurks around every corner. Deadly scorpions and beetles will block your progress and mummies will hunt you down. Only by keeping your wits about you can you hope to unravel the secrets of the great pyramid and escape as a rich man.

You control the explorer using the arrow keys. Many obstacles will block your path, keeping you from taking the treasures and eventually escaping to freedom. Beetles will only move vertically while scorpions only move horizontally. Mummies move in all directions. These enemies are clever and will react when they see you by trying to catch you and end your explorations. Some wall segments can be pushed, allowing you to reach other areas or hide from enemies. The pyramid also contains scarabs that you can use to make the mummies temporarily vulnerable—allowing you to hunt them for extra points.

Deep within the center of the pyramid lies its greatest treasure, the fabled sword of the sun god Ra. It is this great treasure that casts the unnatural light which reaches throughout the pyramid and allows you to see your way so clearly. It is precious beyond measure, but in taking it you will upset that delicate system and the pyramid will be plunged into eerie darkness.

Only the small glow remaining in the sword will light your way now, and formerly simple puzzles will seem new and challenging. All is not lost, however, for the sword has a second function. When wielding the sword you will be able to press and hold the spacebar to temporarily reactivate its glow. The sword transmutes gold into pure light, lighting your way but reducing your score. When the sword is active, the mummies will flee as they do when a scarab is active, making your journey easier, but draining your wealth.



Grading

Grading will happen throughout the team Project. Each piece is worth a certain of amount of points. At the end of the project the total points earned will be calculated and divided by the max. The resulting percentage will be the percentages that is multiplied by 40%. This result will be your Term Project score.

Item	Points
Tank Game	150
Second Game	150
Documentation	75
Presentation	18
Total:	393

For the tank game your grade will depend on the number of successfully completed requirements.

For the second game your grade will depend on the completeness of the game based on the description given in this document and the rules of the game.

If you chose a different choice for the second game. A list of requirements needs to be drawn BY YOU IN THE REQUEST EMAIL in order to grade your game.

Both games will also be graded for:

- Smooth Performance
- pleasant user experience (score, background music, etc)
- Code Cleanliness
- Project Structure

In its simplest terms, I will simply play your games while grading.

Email Request to Change Second Game

If your team would like to do something different than what is presented in the Second Game Options Section. Then you must email a request to the instructor. The request must follow the following email format:

Send email to instructor with BOTH team members CC'd

Subject: CSC413.NN Spring 2018 TeamMM Second Game Request

Where NN is your section and MM is your team number.

Then in the body of the email you will explain the high-level detail of your game.

Then list out a set of requirements like how I did for the tank game. Don't be afraid to list a lot. Let the instructor decide which ones to remove. It is the instructors job to reduce the scope of your game.

Then state that both you and your team member have agreed on this game choice.

Any emails that break format (most importantly the subject line) will be ignored or returned. Your team is not allowed to work on the project unless it is approved by the instructor.

Team Members and Team Issues

Since this is a team term project it will require you to work with one other person. Normally this is a situation that doesn't cause any issues, but there are times when things do not work out. I would like each team to keep the following in mind. Especially for teams who have randomly assigned partners.

- Be respectful and civil with your team members. There may be times where you will disagree. Try to have discussions about disagreements instead of a 1 up contest.
- Be responsive. This relates to answering emails, texts, slack messages, etc. All of these should be answered within a reasonable time. Communication is Key in Team projects.
- ***Communication is Key. Communication is Key. Communication is Key***. Most Projects fail not because of technically skills but rather because of miscommunication.
- Avoid the "I can do that no problem" mentality. What I mean by this is don't say you can do something when you think you can but really haven't done it before. There is nothing wrong with not knowing how to do something. But there is something wrong with being dishonest.
- DO NOT share code via email, slack, etc. then have the other team member commit the code. One way the instructor tracks team progress is through git history. And if one person has the commits then it seems only one person is doing all the work.
- If there are any team issues for whatever reason, no matter how small, you need to email the instructor right away. ajsouza@mail.sfsu.edu

Any team issues that arise will be handled privately. The instructor will try to do what's best for the team first, then in certain situations the instructor will do what's best for each individual student.

The instructor reserves the right to split a team if issues cannot be resolved.

The instructor reserves the right to split a team if both members are not doing roughly the same amount of work. This team split can happen without the team asking for it. In some cases, the team member who is doing minimal work may find their term project grade severely affected.