

## TP1 Project Proposal

1. **Project Description** - The name of my term project is Fruit Ninja. Fruits and bombs are thrown randomly on the screen and the difficulty increases as the game progresses. The main objective of this game is to slice the maximum amount of fruits on the computer screen by dragging the mouse across the entire body of the fruit and avoiding any contact with bombs. The score increases with the slicing of each fruit and slicing multiple fruits at the same time can result in a higher score. The game will start off with a splash screen giving the user options to select either Classic Mode, Arcade Mode, or Zen Mode. The goal in the classic mode is to hit all of the fruits on the screen and the game is over when either a bomb is hit or 3 fruits are unable to be sliced by the user. The zen mode has a timer of 90 seconds and the user is not penalized for not slicing fruits. When a bomb is hit, the score decreases by 10. The arcade mode is similar to the zen mode, but the time limit is 60 seconds and it also includes special fruits that can increase the score even more.
2. **Competitive Analysis** - Some similar projects I have seen online include Fruit Slice, Cut the Rope, and Crossy Road.
  - a. In Fruit Slice, the gameplay is very similar to Fruit Ninja, but the added modes are not as complex and the graphics are unique as well. In addition, the accuracy of the user and statistics are not displayed during the game, and the difficulty is not dependent upon how the user is doing during gameplay. This aspect will be one of the main focuses of my project and will help to create a better user experience.
  - b. In Cut the Rope, a piece of candy is attached to a rope and the objective of the game is to slice the rope in such a way that the candy is dropped into the mouth of an animal on the bottom of the screen. My project will include similar ideas to this game in terms of how the fruit is sliced, but my game will not have a set number of levels to reach the end of the game. Instead, my game will include algorithms that will increase the next level's difficulty based on how well the user is doing. An example of this is to make it harder to hit the fruits by overlapping bombs with the fruits for an extended period of time.
  - c. In Crossy Road, the goal of the game is for the user to get as far as possible to the other side of the road without hitting any cars or objects that are in their way. This game involves tapping/ touching the mouse instead of dragging the mouse and the game can end if the user stays stationary for too long. In my game, this will only

occur during the classic mode, but the zen mode and arcade mode have a time limit.

**3. Structural Plan** - I will break this project down into several components that will make the code much more organized.

- a. I will have a fruit class which will be the superclass and it will initialize all of the necessary components of each fruit such as the cx, cy, radius, speed, and direction. The subclasses of the Fruit class will include all the fruits being used in gameplay such as Apples, Bananas, Lemons, Oranges, Pomegranates, and Watermelons.
- b. A bomb class will also be included which is quite similar to the fruit class.
- c. Next, will be the main class which will be integrated with the CMU 112 graphics. This will run the main core of the game and will include functions such as appStarted, mouseDragged, mousePressed, timerFired, and redrawAll. The main algorithmic functions will also be in the main and it includes: fruit and bomb possibility based on user statistics, different levels of gameplay based on user accuracy, ability to slice the fruit and keeping track of hits and misses, and the selection of a random fruit to be thrown onto the screen.
- d. Another class will be used for the audio part of the game and will be played throughout gameplay.

**4. Algorithmic Plan** - The trickiest parts of the project will include slicing fruit rather than just pressing inside the fruit, having parabolic motion of the fruits when they appear on the screen, tracking user accuracy, creating levels based on how well the user is doing in the game, increasing the difficulty of the game by overlapping the fruits and bombs for an extended period of time, and having an appealing graphical user interface.

- a. In order to slice the fruit, I will use the mouseDragged and mouseReleased functions. I will keep track of the previous and current X and Y values when the mouse is pressed on the screen. These values will be set to None in the mouseReleased function. Next, if these values are not equal to None, I will set the previous values to the current values and current values to event.x and event.y.
- b. To have a parabolic type motion for the fruits, I will keep set variables for the direction and speed of the fruits and vary them based on random values. The fruit will be slow on the way up and will speed up on the way down. I will also have a targetY value such that the fruits will not go up past a certain point.
- c. In order to track user accuracy, I will have to keep track of the hits and misses of the user. The hits will increment by 1 each time the fruit is sliced and popped off the fruit list. The misses will increment by 1 each time the user drags the mouse and releases it without interesting with a fruit. The accuracy will be the hits/(hits + misses) and will be displayed on the top middle of the screen

- d. To create different levels based on how the user doing, I will use the weights parameter for the random.choices method and vary them based on user accuracy. If the user is slicing all of the fruits with very high accuracy, the weightage of a bomb appearing will increase. To overlap the fruits and bombs, I will decrease the range for where a bomb or fruit can start from the bottom of the screen.
- e. For an appealing graphical user interface, the images will need to look very sharp and clear and the splitting of the fruits, when sliced, will need to be very precise. I plan on using images for the fruits and will try to simulate the slicing in 3D.

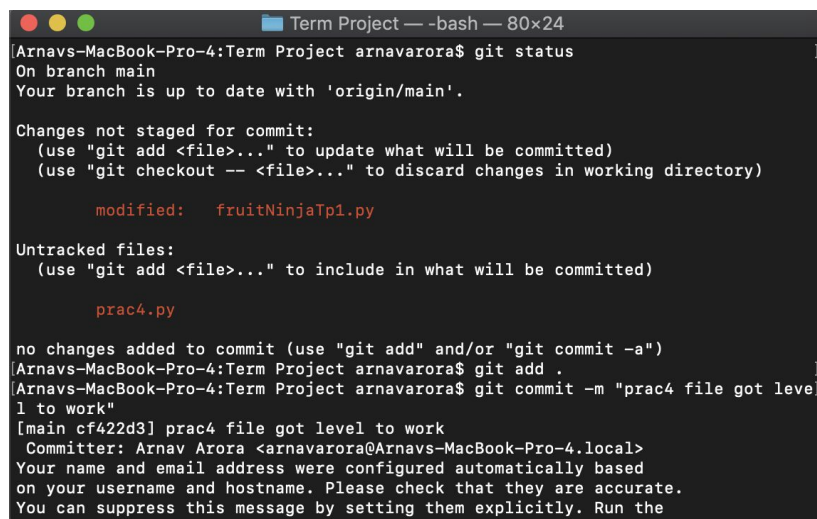
## 5. Timeline Plan -

- a. TP0(11/23/20) :
  - i. Create classes for the fruits and bomb and work on gathering images for the home screen and gameplay screen.
  - ii. Also, create splash screens and menu screen.
  - iii. Learn about pygame (strictly for audio) and show how to use it in the tech demo.
- b. TP1(11/30/20):
  - i. Finalize classes for the fruit and bomb.
  - ii. Create a main class which will be the main core of the game and integrate it with CMU 112 Graphics.
  - iii. Make the algorithm for the parabolic motion of the fruits
  - iv. Be able to slice the fruits by dragging the mouse
  - v. Track user accuracy and display on the top of the screen
  - vi. Display lives left, score, and current mode/level on the screen
  - vii. Add multiple levels and game modes and give the user an option of which one to play on the home screen
  - viii. Randomize fruits and bombs by assigning a certain probability to each
- c. TP2(12/5/20):
  - i. Tune how the game throws fruit at the current level based on user data
  - ii. Determine difficulty of next level based on user data
  - iii. Make the fruit and bombs overlap for more time in a given area to increase level difficulty.
  - iv. Improve graphics and make fruits look more realistic with the ability to rotate and slice in half
  - v. Make all game modes and levels inside each game mode
  - vi. Add audio to game
  - vii. Have a close to 100% working game
- d. TP3(12/9/20):

- i. If MVP is reached, research and implement external modules that can be used to make gameplay smoother
- ii. Fix any bugs and improve upon functionality
- iii. Make parts of the game such as slicing and accuracy more precise
- iv. Have a fully working game

## 6. Version Control Plan:

- a. I made an account on GitHub and constantly upload my work. This enables me to look at any changes I made to my code and what I specifically changed. I can also go back to previous copies of my code. Every time I make a major change or plan to make a major change in my code I use the following commands on my terminal to upload my work:
  - i. Git status
  - ii. Git add .
  - iii. Git commit -m "Add a comment on what was changed in the project"
  - iv. Git push
- b. After each day, I also upload all the files in my Term Project folder to my CMU google drive account.
- c. Images of how I am doing this:
  - i. Terminal commands for uploading work to GitHub, a screenshot of how files are uploaded on GitHub, and uploading files on google drive



```

Term Project — -bash — 80x24
[Arnavs-MacBook-Pro-4:Term Project arnavarora$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)




















        modified:   fruitNinjaTp1.py

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        prac4.py

no changes added to commit (use "git add" and/or "git commit -a")
[Arnavs-MacBook-Pro-4:Term Project arnavarora$ git add .
[Arnavs-MacBook-Pro-4:Term Project arnavarora$ git commit -m "prac4 file got level
1 to work"
[main cf422d3] prac4 file got level to work
Committer: Arnav Arora <arnavarora@Arnavs-MacBook-Pro-4.local>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the

```

 <b>Arnav Arora and Arnav Arora</b> <b>prac4 file got level to work</b> <span>cf422d3 13 hours ago</span> <span>10 commits</span>			provided.
	__pycache__	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	.DS_Store	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	Fruit Ninja Combo Sound.mp3	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	Fruit Ninja Home Screen.jpg	added accuracy to TP1	22 hours ago
	Fruit Ninja Level Up.mp3	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	cmu_112_graphics.py	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	demo.py	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	events.py	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	fruitNinja.py	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	fruitNinjaBackup 11:26.py	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	fruitNinjaBackup1.py	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	fruitNinjaBackup11:26 #2.py	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	fruitNinjaTp1.py	prac4 file got level to work	13 hours ago
	homescreen.py	fixed movement of fruits	2 days ago
	mouse.py	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	movingDot.py	mouse dragged, fruit classes init, fruits going up and down, home scr...	3 days ago
	prac1.py	homescreen	2 days ago
	prac2.py	homescreen	2 days ago

**Releases**


No releases published  
[Create a new release](#)





**Packages**


No packages published  
[Publish your first package](#)

**Languages**


- Python 100.0%


 Drive








 New


☒ Priority


 My Drive

 Shared with me

 Recent


 Starred

 Trash


 Storage


1 GB used


My Drive



**Trash has changed.** Items will be automatically deleted forever after they've been in your trash for 30 days. [Learn more](#)

**Quick Access**


 **Project Proposal**  
You edited today


 **TP Progress**  
You often open around this time


 **Final Draft Research Proposal**  
You edited in the past month


 **Final Research Proposal**  
You edited in the past month


**Folders**


 Arnav

 CMU\_Summer\_2019

 College Applications

 Interpretation and Argument ...

 Sat Subject Tests

 school

**Files**

**Name** ↑

## 7. External Module List:

- Pygame (strictly for audio) (Already approved by tech demo)