VERSION [0.0]

FEBRUARY 16, 2018

CS 4053: COMPUTER GRAPHICS

# TEAM 05 PROJECT PLAN

## COLLISION OF COLORED PARTICLES USING JAVA WITH OPENGL

16/20

- Report is more bullets than prose, making it seem disjointed.
- Milestone tasks are well detailed, but the whole thing seems a little unambitious.
- Pick an aspect or two to experiment with, and allocate time for that. Dynamics of exploding white particles would be a good choice for this.

BETSIE KOSHY
SANDY HUYNH
MISTI ANDERSON

# Table of Contents

## Project Plan

### Project Proposal document

The subject of the Team 05 animation will be the collisions of colored particles. For this project, a particle will be defined as a small circle. The particles will be scattered, at random, across a black background, and will be different colors. If two particles of differing colors collide, they will merge to form a larger particle. Particles of the same color will "bounce" off of or "repel" each other. In addition, the newly formed particle will combine the colors of its "parent" particles with respect to the additive color model (ex. If a red and green particle collide with one another, it will turn into one bigger yellow particle). Once enough particles collide together to form a large, white particle, it will then explode or "pop" and disperse the smaller colored particles it was originally formed with. This will give us the opportunity to experiment with lighting, motion, shaders, collision, and blending.

So... 2-D only?

### Overall Process

First we will research on our subject to get a better understanding of the mechanics that will be put into play while developing this program. Although it may seem like a simple particles program, we will have to understand subject matters such as: physics - velocity, lighting, amalgam of colors, shaders, collision and eruptions.

Figuring out resources that we will use for this program will also help ease the production as well. We will be implementing our project in Java using OpenGL. We were also informed that there are libraries specializing in creating particles, so using the libraries as reference will also help in starting with this project.

Once we have enough understanding, we will ~~set~~ Analyze? a schedule to complete each of our goals. By following the timeline, we hope to be organized, and be able to implement our program smoothly.

## Team Structure

### Team goals

Using the knowledge that we developed through research, we have set these goals:

- Create a simple free flowing particle simulator
- Implementing different RGB colors to each particles
- Collision detection between particles — missing a verb. Designing? Implementing? Testing?
- Blending of colors on collision
- Eruption of particles

These are "feature" goals, and fairly general ones. What's your overall goal? What are your specific goals for each of the above features?

## TEAM ROLES

| Role Type | Description | Assigned To |
|---|---|---|
| Team Lead | *Helps assign responsibilities/tasks to team members, and monitors the progress and individual participation to keep the team on track.* | Sandy Huynh |
| Project Analyst | *Performing, analyzing and providing support to the entire project team. Assist in creating the application, and identify and solving problems the team may encounter. Makes recommendations for improvement.* | Betsie Koshy |
| Software Developer | *Responsible for building the deliverables and communicating the status of the software project to the team.* | Misti Anderson |

**Although Roles are set, members will help one another regardless of what role is assigned to whom.*

## TASKS

| Tasks | Description | Assigned to |
|---|---|---|
| Main Frame | *Create the main frame of the program.* | Betsie Koshy |
| Creating Particles | *Implement a simple particle generator.* | Betsie Koshy |
| Color Implementation | *Implement different RGB colors to each particles.* | Misti Anderson |
| Collision Detection | *Create a function that will be able to detect if particles collide with one another.* | Misti Anderson<br>Sandy Huynh |
| Amalgam of colors | *Create a function that will give the newly formed particle a mixture of the parent particle's colors.* | Misti Anderson<br>Betsie Koshy<br>Sandy Huynh |
| Repelling particles | *Same color particles will repel off each other.* | Sandy Huynh |
| Eruption of particles | *Create a function that will detect if a particle has been fully turned white, or is ~~in a large scale~~. At this point, it will explode into smaller particles.* | Sandy Huynh<br>Betsie Koshy<br>Misti Anderson |

**Tasks may be broken up depending on the difficulty of creating the methods.*

What are the dynamics of particle interaction, specifically?

are you sure this can happen? how will you make sure for your color algebra?

by what definition?

## Milestones

The Milestones for this project are as follows:

- Create a prototype of what the program will look like and how it will function
- Create a framework for the program (frame/canvas/space)
- Create a particle object (class)
- Define the behavior of particles (patterns of motion, differing velocities)
- Add a color values to the particle objects
- Define a method that will recognize the collision of two particles
- Define a method (called when a collision is detected) that will merge the colliding particles into a larger particle
- Define a method (called when a collision is detected) that will merge (blend) the colors of the colliding particles
- Define a method that will cause particles of the same color to repel each other
- Define a method that will cause a fully merged particle (i.e. white) to explode (decompose) into many smaller particles with differing colors

## Timetable

| Date Start/End | Phases |
|---|---|
| Feb. 16 - Feb. 21 | Phase 1: Research and Prototype<br>1. Create a prototype of what the program will look like and how it will function |
| Feb. 21 - April 17<br>1. Feb 21 - Feb 28<br>2. Feb 28 - March 7<br>3. March 7 - March 14<br>4. March 14 - March 27<br>5. March 27 - April 3<br>6. April 3 - April 10<br>7. April 10 - April 17 | Phase 2: Building Project Structure<br>1. Create a framework for the program<br>a. Create a particle object (class)<br>b. Define the behavior of individual particle objects<br>2. Add a color values to the particle objects<br>3. Define a method that will recognize the collision of two particles<br>4. Define a method (called when a collision is detected) that will merge the colliding particles into a larger particle<br>5. Define a method (called when a collision is detected) that will merge (blend) the colors of the colliding particles<br>6. Define a method that will cause particles of the same color to repel each other<br>7. Define a method that will cause a fully merged particle (i.e. white) to explode (decompose) into many smaller particles with differing colors |
| April 17 - April 24 | Phase 3: Review Finished Project |
| April 24- April 30 | Phase 4: Project Delivery<br>1. Create PowerPoint for presentation |

** Note: some of these phases/tasks will be performed in parallel (ex: reviewing each other's work, creating final report/presentation, etc.) Timeline may change throughout the process.*

### Phase 1: Research and Prototype

Each team member will research materials and documentation that will help implement this project. Materials/documentation will include what are the best resources and tools to use while using Java with OpenGL. This will also be the phase in which we will sketch out a prototype to see how we want the application to look like and what kind of actions it will do.

### Phase 2: Building Project Structure

This phase will be the most tedious. Once we have enough understanding on how our project will function, we will begin building the structure and setting deadlines for each module. The software developer will begin creating the frame of the project. Each member will then contribute knowledge they have gained from phase 1 and incorporate it into the project.

### Phase 3: Review Finished Project

After the project has been completed. We will start to review the project to see if there are any issues that may arise. If we identify a problem, we will take this time to find a solution for the issue and fix it. This will be a rinse and repeat phase until we see that our project is in good standing. In terms of what criteria? What constitutes success for your project? Define it from the start, to be objective when it comes to evaluation/assessment.

### Phase 4: Project Delivery

Phase 4 is for the final delivery of our project. At this time, there should be no major issues, and the project should function as planned via the prototype. We will create a PowerPoint explaining the methods we used to create our program and display a demo of our working project.

### Implementations

We will be using Java with OpenGL to implement this project. We will be using GitHub as a source code manager because it can help keep track of changes as the team develops the program, and will also help coordinate work among members of the group. We do not know what kind of packages we will be using yet, but we were informed of different resources - specialization libraries, that would be good reference points for particle generating. — Yes, but what you're doing is straightforward enough to roll your own.

### Deliverables

A Gradle file will be cleanly packaged with a detailed instruction on how to use the program. It will give details on how the program works, and how users will be able to run the file. All of this files will be in a .zip file and turned in together.

### Analyze

When analyzing our product, we will be looking for behavior between particles that is inconsistent with our desired effects. All members will help to recognize these defects and (informally) report any findings to the group for correction. In addition, we will get feedback from outside, non-technical sources in order to gauge the effectiveness of the visual outcomes.

Who? How?

How defined? What aspects?

## Risk Assessment and Mitigation Strategy

| Risks | Mitigation Strategy |
|---|---|
| Defects could be found in the later stages of our project | The project analyst or team leader will make sure each phase of the project is successful |
| Scope is not completely defined | The team will define the scope early on |

What if this is determined later on?

What if someone leaves the project (for whatever reason)?

## Potential Difficulties

While we work on specific methods, we could run into potential difficulties of merging the colliding particles (velocities) into a larger particle, merging (blend) the colors of the colliding particles, particles of the same color to repel each other. The specific tasks that all three of us will be working together on are amalgam of colors and eruption of particles.

## Consider Safety

For this specific program, there could be potential dizziness from observing the particles moving around and colliding with other particles, and of course, carpal tunnel and strained eyesight from staring at a computer screen. Other than the usual concerns for software developers, there will not be any other safety concerns.

## Training and Maintenance

Our application will be very simple to run. Since this is the case no training and maintenance will be provided after the finalization of this project.

## Future Directions

Additional functionality would be utilizing "cookie cutter"-like objects to attract and repel certain particles (ex. a heart shaped outline-- attracts red particles, repels all others). This additional functionality will be implemented only in the event that we complete our set forth goals ahead of schedule due to ease of implementation. We would like to create a product that exhibits all of the essential components of the course without overreaching our limitations. However, we do love a challenge and if time permits, we will certainly add this functionality.

I think it's really doable. I encourage you to do this one and identify "even harder" things as potential future work directions.