CS 408
Prof. Alain Crotte

Ariston Bodnarchuk
200285478

Assignment 5

Gas Simulation Program

Video Demo

https://youtu.be/et5sbSEaZ5k

CS 408
Prof. Alain Crotte

Ariston Bodnarchuk
200285478

Github

All of this project can be seen on my github
I have added krupa15 to the private repo.

For any issues please email me at bodnarca@uregina.ca
My Github username is “spacemanstan”

User Instructions

Keyboard Controls

- ‘c’ Key
  - Toggles color mode on or off
- ‘f’ Key
  - Toggles density fade on or off
- ‘s’ Key
  - Toggles stroke on or off when drawing cells
- ‘ ‘ Key (*spacebar*)
  - Toggles sand mode on or off

Mouse Controls

- Left Mouse
  - Click & Drag:
    - Add density to cell mouse is over (represented as dye)
    - Add velocity to cell mouse is over
- Right Mouse
  - Click & Drag:
    - Remove density to cell mouse is over (represented as dye)
  - Click no Drag:
    - Toggles density fade on or off
- Middle Mouse
  - Toggles color mode on or off

Color Mode

When enabled maps density to a hue between 100 and 175 which is from green to purple with a bit of orange thrown in there, it gives a pretty cool effect

Toggle Fade

When enabled the density or dye will slowly be drained out of every cell, until it reaches zero

Toggle Stroke

When enabled each cell then draws a black outline, this breaks the illusion of it being homogenous and allows the user to see each cell individually

Sand Mode

This came about through experimentation, if you increase the viscosity or thickness of the contents of the cells being simulated it will behave like sand, and you can draw using erase!

CS 408
Prof. Alain Crotte

Ariston Bodnarchuk
200285478

How to Use

Just click around and have fun!

The entire simulation is controlled with mouse input with keyboard input support as well. Keyboard input has a secret (not really secret if I just told you) feature I call sand mode that changes the viscosity of the simulation mid simulation allowing the user to speed up or slow down the movement of diffusion and how much friction it experiences.

Creative Features & Description

My work relied heavily on the research and mathematical work done by Jos Stam, and the interpretations of his paper by Mike Ash. I would not have been able to make software as cool as this without their hard work and mathematical prowess. Basically my entire assignment is the creative feature at this point lol.

The math behind this is pretty over my head, but there is a lot of great descriptions of formulas and how they work in Jos Stam's paper, as he is the one who did the mathematics behind it. My biggest constraint with this assignment was time. Initially I had no luck implementing anything that was given to us for the gas simulation. In frustration I came upon the work of Jos Stam in relation to fluid simulation and thus was able to implement his work and translate it into a java solution that interfaces with processing. There is a lot of room for optimization in the code and I desperately want to refactor it, but sadly I do not have enough time to get the code in a state I am happy with. The C code written by Mike Ash and Jos Stam leaves a lot to be desired in terms of readability, especially comments as the mathematics behind it even with formulas and some description alludes me. The simulation still makes use of multiple copies of the same arrays to make calculations based upon the previous state. The core functionality of this fluid system is the same however the math behind it is much more flushed out and robust, and in my case, easier to get implemented. I had no issues getting the code running in java and was able to get some basic things working in processing right away. If I had more time I would like to use PVectors to hold velocity, which would allow me to rewrite all the functions to perform vector math instead of the decomposed parts of one. I would also rewrite the set boundary code which reverses velocities around the edge to direct the simulation back into itself keeping it a close system that doesn't leak all its contents out.

The main concept of this system is that dye is used to visualize the density and velocity of the system as it allows the human eye to observe these forces. The code works in two primary steps from the main draw loop, sometimes three. The simulation takes a step, the density is rendered to the screen through the dye concept, and if enabled, the simulation will fade part of each cell's density. Each step consists of three main operations itself: diffuse, project, and advect; with two supplementing functions. Diffuse simply disperses the density and velocity to the surrounding cells, much like the provided simulation code, however in this simulation all 8 surrounding tiles are used instead of a four tile overlap system. Project keeps the density of each cell in check and advect just processes the movement of everything

CS 408
Prof. Alain Crotte

Ariston Bodnarchuk
200285478

Citations

- Mike Ash
  - https://mikeash.com/pyblog/fluid-simulation-for-dummies.html
- Jos Stam's work
  - https://www.dgp.toronto.edu/public_user/stam/reality/Research/pdf/GDC03.pdf