CS251 Project Report by Group 04.

Mridul Sayana
140050001
140050001@iitb.ac.in
Surender Singh Lamba
140050075
140050075@iitb.ac.in
Thanuj Raju
140050077
140050077@iitb.ac.in

October 20, 2015

Overview

This report is made to describe the concepts behind our CS251 project Box2d simulation .

We also would like to mention the references used to complete our project through this report.

Motivation

This project is a demonstration of the power of the software systems like BOX2d .

The basic concepts and machinery used in this can be utilized to generate even more complex and relevant games.

So we wish to build a machine which explores the whole power of Box2d and demonstrates it visually!
This project involves several concepts discussed in this course like makefile, doxygen, bash, beamer, libraries,html etc...

This serves as a good and brief outlook of this course.

Base Code Problems

The crux of our simulation is the water and representing it with a bunch of balls is not doing justice to simulation and also effects RAM very badly.

So, while searching for an effective way for this, we stumbled upon LiquidFun which is an extension of Box2D and has a particle module which efficiently does the task we want.

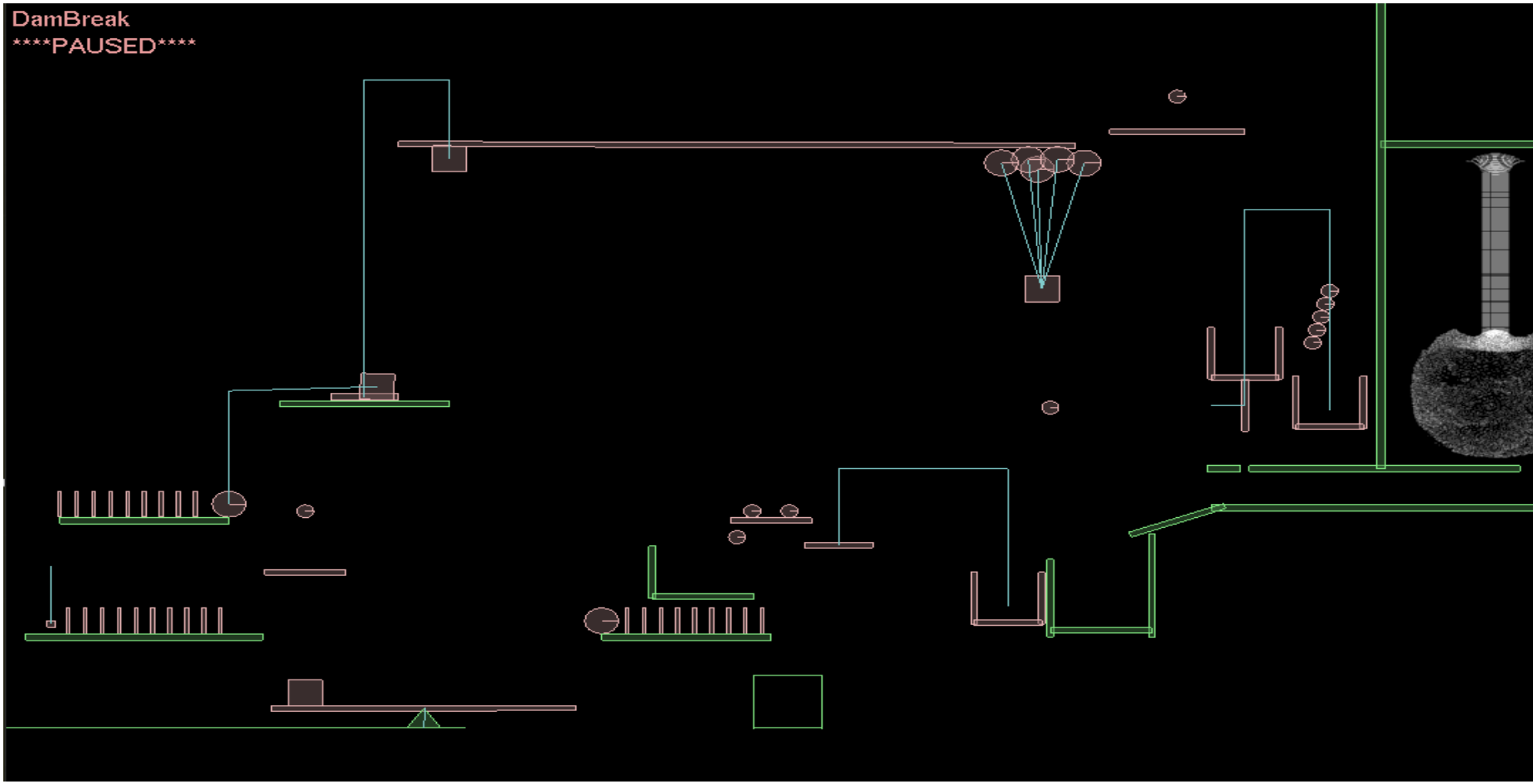
For the documentation of LiquidFun, you can visit their github page at http://google.github.io/liquidfun/

Base Code Problems

We have used Box2D version 2.3.0, a different base code from the one provided to us. Well, Desperate times, Desperate measures.

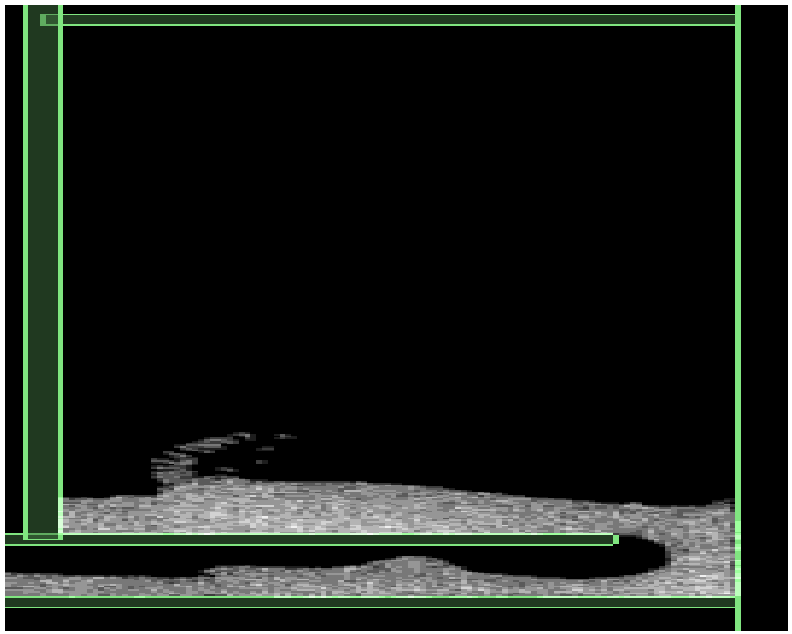
LiquidFun is a 2D rigid-body and fluid simulation C++ library for games based upon Box2D. It provides support for procedural animation of physical bodies to make objects move and interact in realistic ways.

This version has direct support with ios, android, cocos, javascript and vbx



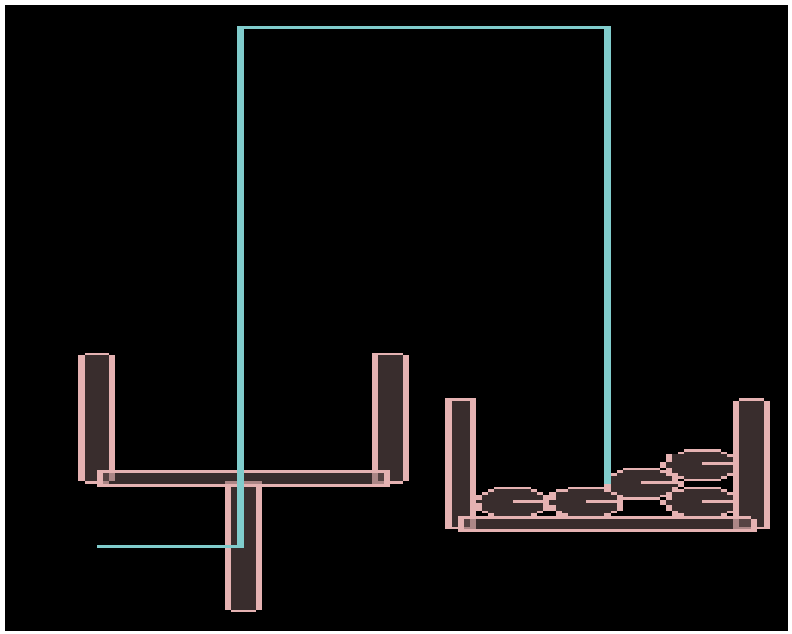
Outlook our our simulation which involves different physics concepts

FLUID



We found difficult to create Liquid using our Box2D so we used extended version of our Box2D includes LiquidFun which can create liquid.

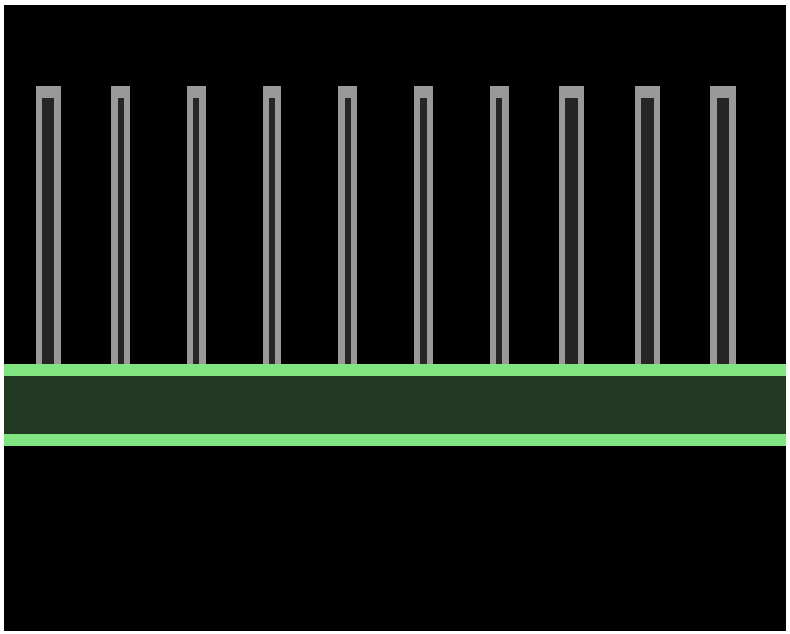
COMMON BALANCE

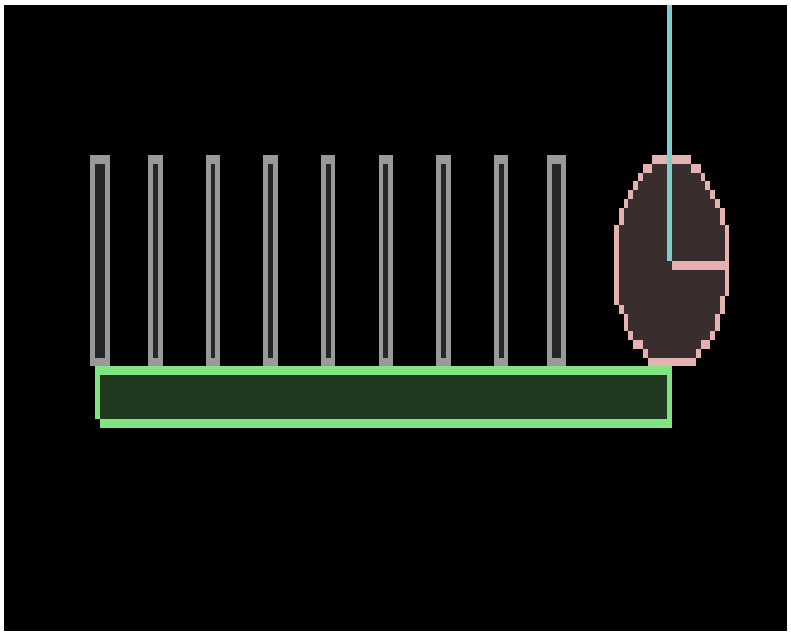


We used pulley joints to make this balance.

It takes two anchors and two objects and connects the two objects with the thread through two points giving the effect of a common balance.

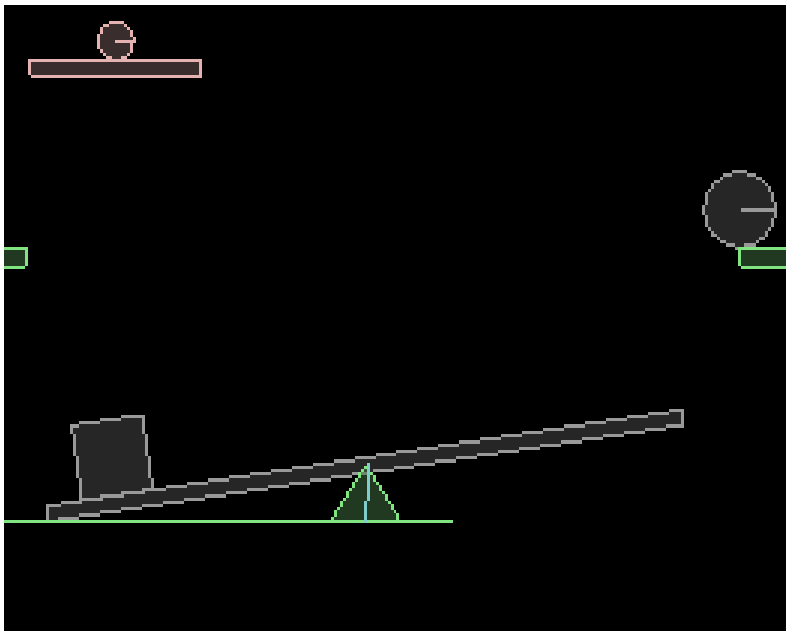
DOMINOS

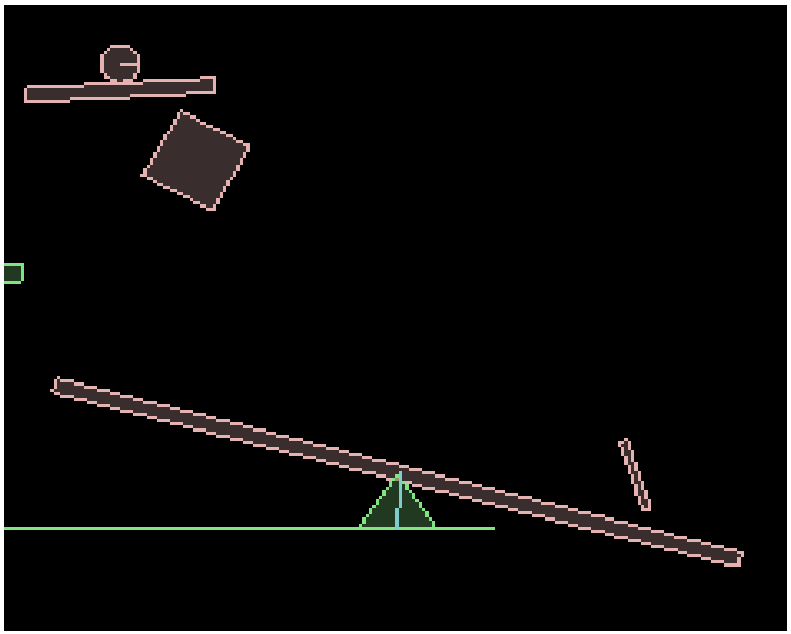




We made it by placing a static platform and placing a small dynamic rectangular blocks at regular intervals in such a way that if one of then falls remaining dominos will fall

SEE-SAW

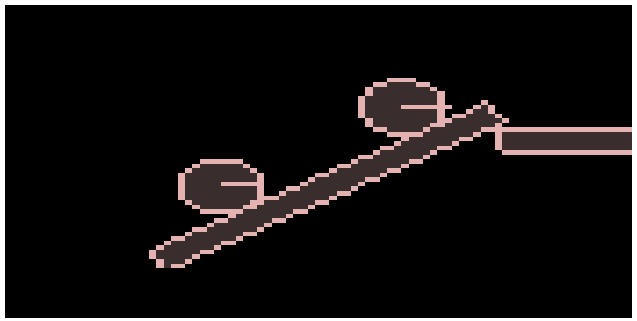




We made the simulation by creating a revolute joint between a static triangle and a dynamic rod,and placing a rectangular block on the plank

We arranged the system such that after ball hitting the other side of see-saw that block hits the hinged platform above it.

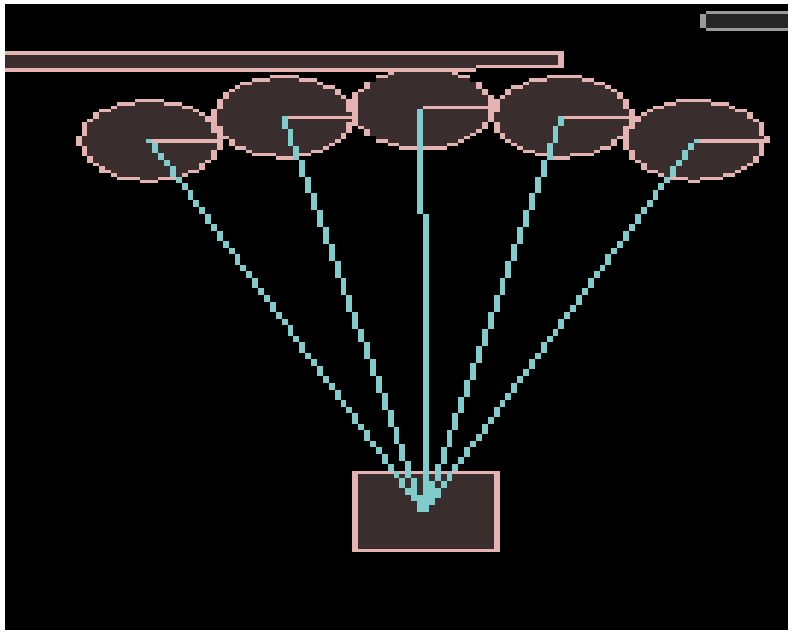
REVOLVING PLATFORMS



We used Revolute Joint to make the effect of hinged rod. It takes one anchors and two objects and connects the two objects with the thread through the point.

We defined one of the objects as a dynamic rod and other as a small static object with the anchor point on itself which resulted in the effect of this hinged rod

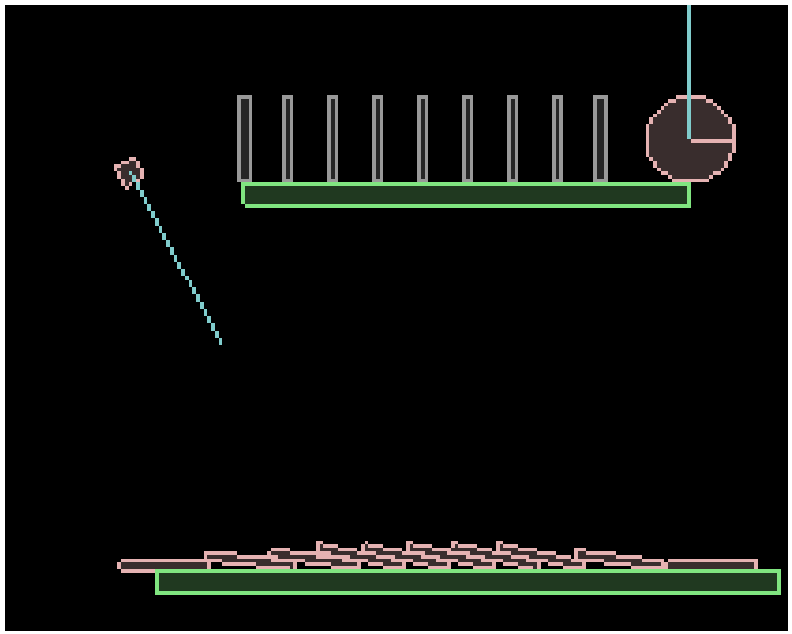
BALLOONS



This is the Balloon system.

We made the simulation by changing the gravity of balloon to negative.

PENDULUM



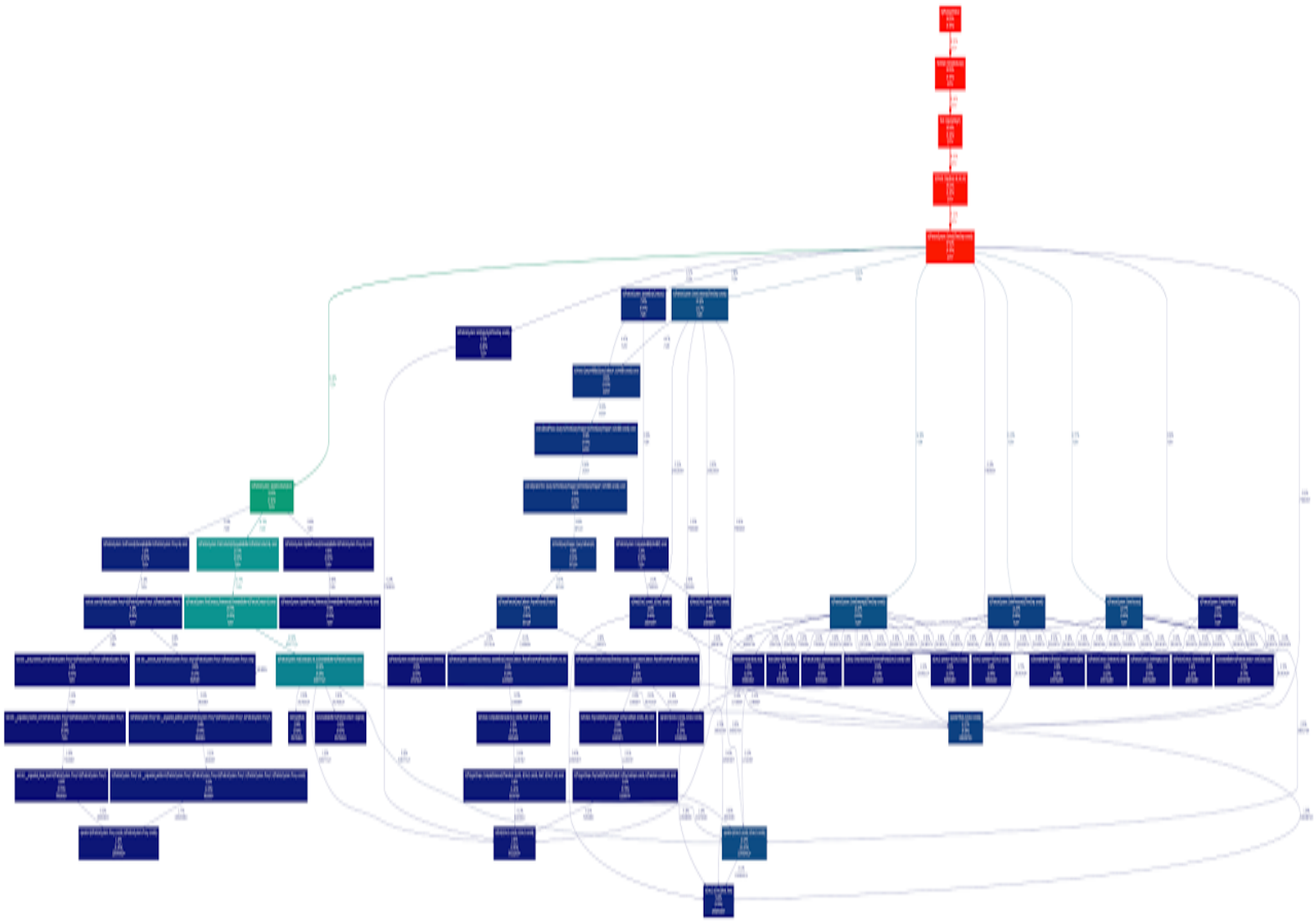
It is created using anchoring the thread such that the pendulum hits the dominos above it.

About Profiling

Profing is very useful to write optimised programs.It gives the description of time consumed by different functions in our code so that we can understand which function to be optimised.

The picture in the next slide is very big, so to see clearly, you can see the picture outside in the report folder.

Profiled output of Project



Citations

For base code we have used the LiquidFun under Creative Commons license and Copyright (c) 2006-2013 is held by Erin Catto http://www.gphysics.com. Link to github https://github.com/google/liquidfun

For the html we have used a template from which is also opensource

Credits to Jose Fonseca for gprof2dot.py to make the profiling easy. Link to github https://github.com/jrfonseca/gprof2dot

Stack Overflow has been of much help for the project.

Thankyous

This project uses many of the things we learned this semester.

Some of the mainly helpful things were cmake, Makefiles, gprof and tex files

And as goes with Occams Razor principle, here we officially thank people.

Thanks for a informative and exciting semester to Prof.Sharat Chandran and always supportive TA Akash Chaudhary.