# **INFO6022 – Physics & Simulation 2 - Mid-term Exam – Winter 2024**

Instructor: Michael Feeney

## The exam format:

* You may use any resources you feel are necessary to complete the exam, but you are to answer the questions **on your own**. I will be looking for plagiarism (i.e. copying) very carefully. There is *no possible way* that the specific code to answer these questions, or the output to the screen, would be very similar to the look of another student’s code. Remember, this is a test and there are very clear policies about cheating on tests.   
  + <http://www.fanshawec.ca/admissions/registrars-office/policies/cheating-policy>
  + <http://www.fanshawec.ca/sites/default/files/assets/Ombuds/cheating_flowchart.pdf>
* The questions are **NOT** of equal weight. There are five (5) pages with three (3) questions
* You are to submit a **single** project that answers all the questions:
  + You are ***not***to submit a separate project per question.   
    (if you do this, I will either mark the first one, or just give you a mark of zero.)
  + You are ***not***to have any code and/or reconfigurations to demonstrate different questions.   
    I will ***not***modify your code and/or files and/or file structure *in any way* and will mark your submission *as is* – meaning that if that untouched build + run doesn’t answer certain questions, then you’ll get zero on those questions.

|  |
| --- |
| *(Note: Having separate projects that contain your own DLLs or libraries that your code uses are 100% OK.)* |

* For applications: if it doesn’t build and run, *it’s like you didn’t answer it*. I’ll correct trivial, obvious problems (like you clearly missed a semicolon, etc.), but you need to be sure that it compiles and/or runs.
* This exam should nominally take you three (3) hours to complete (i.e. if it was physically scheduled in a classroom, it would be for three (3) hours). You have until **11:59 AM** on **Wednesday, February 28th** to submit all your files to the appropriate drop box on Fanshawe Online. This is an exam, so don’t miss that deadline.
* Also be **SURE** that you are actually submitting the correct files pointing to the correct folders, etc.
* I’ll be monitoring e-mail, discord, and will be in my office (6th floor) for part of the day, if you have any questions.

## Questions:

You are to implement parts of Warhammer 40,000 cinematic, specifically the from about 1:40 to 1:50, cued up here: <https://youtu.be/gBgXH7eyRo4?t=102>

In the various games, there are many soft cloth physics sort of effects, including flags, banners, and capes.

Rather than using the specific “banner” shape of the “Blood Ravens” chapter, you will use a typical plan rectangular flag shape as well as having the flag mounted on the side, rather than hanging from the top – as it is in the cinematic.

|  |  |
| --- | --- |
| **The Space Marine banner:**   * Mounted on the top * Hangs down * Has little bits hanging at the bottom | **Typical flag:**   * Mounted on the side * “Flies” to the side * Is simple rectangular shape |

You can find one of your favourite Warhammer Astartes banners here: <https://www.deviantart.com/sinistha/art/Warhammer-40K-Astartes-Chapter-Banners-840856008>

A collection of banners with different designs

Description automatically generated

(I googled “space marine chapter banner” and found a number of them)

1. (25 marks) Recreate the flag after the Space Marine has finished raising it, and the camera A banner with a face and wings

   Description automatically generatedis moving away, with the following features and changes:  
   * You will use a regular rectangular flag (like the “I ❤️Beer” flag on page 2), mounted on the *side* **not** from the top like the in the video.
   * For a texture, use any Space Marine chapter you’d like.
   * Assume there’s a strong enough wind to hold the flag more or less straight to the side (like the “I ❤️Beer” flag on page 2)
   * Use a cylinder or some equivalent shape as the flagpole.
   * Use some “ground” mesh as the ground, like the MeshLab terrain, or something you’ve found.
   * Use some kind of evening, daytime, etc. skybox or equivalent to give the effect of the sky and clouds behind the flag.
   * Place a light (or enough light) to easily see the flag, etc.
   * This must be a textured, lit scene, not wireframe or something like that.
2. (200 marks) Using the soft bodied Verlet integration technique from class, show the following effects on the flag from question 1:  
   * The flag must have at least 10 x 10 “nodes” along each axis of the flag.   
     In other words, the smallest distance between nodes is 1/10th the length/width of the flag.
   * Start with the wind being essentially absent, and the flag hangs down.
   * Gradually increase the wind until the flag is flying like it did in question 2.
     + You can either add two controls that “turn off/on” the wind, or increase/decrease the wind blowing over the flag.
     + Note that when the wind “stops blowing” the flag will gradually fall down, and when the wind blows, it will gradually fly. This might not take much time (like if there is a sudden “gust” of wind) but you can’t just place the nodes in place. It needs to be simulated.
   * The “wind” can be controlled by changing the overall forces on the nodes of the flag. Usually “gravity” is pulling the flag down, but the “force” of the wind either overcomes this force – or since this is a game, can gradually replace it (the force can *gradually* go from straight down to straight to the left depending on how “strong” the wind is).
   * The flag should be flexible by not “stretchy”, like cotton or nylon rather than spandex or something.
   * The flag should “flap” somewhat in the wind. You can do this by using random perturbation of forces on the “ends” of the flag or a sin function, or something like that. If it looks “good enough”, then I’m OK with it – but it has to “looks like it’s a flag flapping in the wind” to a typical person.
   * *Like question 1, it needs to be a solid, textured, lit mesh, not wire frame or nodes, etc.*
3. (200 marks) Mimic the holes that appear in the flag:  
   * In the cinematic, I’m assuming it’s the Orcs that are shooting the flag with bullets or something. You can assume it’s something else if you don’t like that. Maybe there’s really high winds blowing rocks through it?
   * (Assume that the winds are very high)
   * Assign a key to do the following:
     + Randomly choose a circular location on the flag. This needs to be large enough to encompass at least a few nodes.
     + Disconnect the connections between these nodes to simulate parts of the flag being knocked/shot out.
   * Assign another key to do the following:
     + Disconnect the flag from the flagpole and have it fly away.
   * Assign yet another key to do the following:
     + Reset the flag back to “full health”: back together on the flagpole, like in question 2.

That’s it.