INF2110 - Introduction à l'Informatique Graphique **M1 Informatique**
TP2 **Fall 2019**
Prof. Caroline Larboulette

What you need to turn in at the end of the lab session

At the end of the lab session, you should upload a .zip file containing the source code of exercices 1 to 3 (can be in multiple files) as well as a .pdf report describing what you have done: source code extractions with associated screenshots to illustrate the results you have obtained. Remember the assignment may be graded.

For this lab, first download the file tp2_squelette.cpp from the lecture's webpage on Moodle.

Have a look at the code and the comments. Compile.

1. Drawing Bitmaps and Fonts

The commands `glRasterPos*()` and `glBitmap()` position and draw a single bitmap on the screen. The example file tp2_squelette.cpp creates a bitmap (the letter F) and draws it three times on the screen. Bitmap data is always stored in chunks that are multiples of 8 bits. However the actual bitmap doesn't have to be a multiple of 8 (we don't necessarily draw the entire bitmap).

Have a look at the source code, and, once you have understood what each function does, create your own letters or patterns and write your name (or something else) on the screen. Use `glColor*()` to color the letters. What do you observe ?

```
void glRasterPos{234}{sifd}(TYPE x, TYPE y, TYPE z, TYPE w);
void glRasterPos{234}{sifd} v(TYPE *coords);

void glBitmap(GLsizei width, GLsizei height, GLfloat xbo, GLfloat ybo,
GLfloat xbi, GLfloat ybi, const GLubyte *bitmap);
```

2. Antialiasing Points or Lines

To antialias points or lines, you need to turn on antialiasing with `glEnable()` passing in GL_POINT_SMOOTH or GL_LINE_SMOOTH. You also need to enable GL_BLEND, with a correct blend function and hint.

Try drawing lines with the following settings:

```
glEnable(GL_LINE_SMOOTH);
glEnable(GL_BLEND);
glBlendFunc(GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA);
glHint(GL_LINE_SMOOTH_HINT, GL_DONT_CARE);
```

Make a screen capture before and after to observe the difference. For other options, check the documentation.

Use antialiasing on points.

3. Implement the Cohen-Sutherland line clipping algorithm

Create a few segments and a rectangular window. Using the Cohen-Sutherland algorithm, clip the lines so that only parts that are in the window are drawn.
The details are up to you !