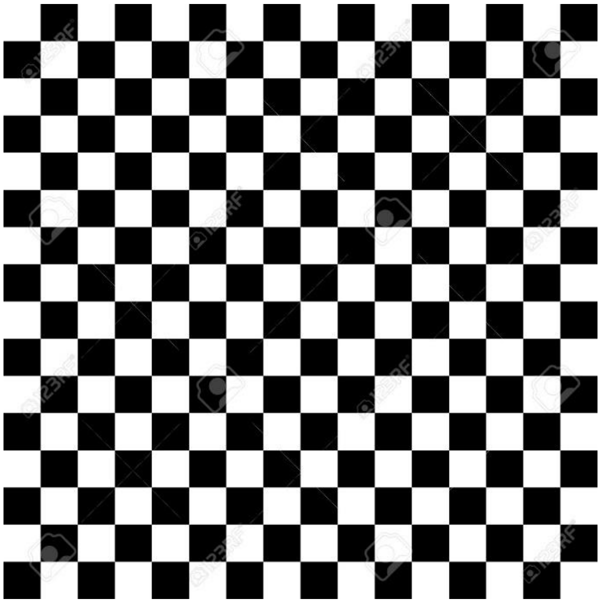
Instruction:

1. Ideally you should program on github so that each edit is self-explanatory. Please also include a word file briefly explaining how you program. You could use any third party libraries or software as long as you document it.
2. The two questions are open ended, and there is no absolutely correct answers to them. You have the freedom of using C++ or Java in programming. Your answer to question 1 is evaluated by how your program performs on our hardware platform. The key is to solve the flickering problem.

Question 1

Build a time-sensitive user interface/video for a monitor that has the resolution of 1020(width)*760(height) pixel and 60hz refresh rate. The user interface is divided into squares that are 20 pixels' wide. Each square will need to blink at a different frequency (alternating between black and white). Take note that modern monitors usually have a refresh rate of 60hz, meaning that it is only able to display a new frame every 1/60 seconds. Hence the set of frequencies you should use are {60/2, 60/3, 60/4, 60/5, …. , 60/(1938)} hertz.



Challenges: while writing such a program is not difficult, to ensure that each square really blinks at the frequency on the screen is not easy. You may observe that there are "flickers" or "defect", i.e., a square that is supposed to blink at 30hz are interrupted sometimes. This may be due to the fact that your graphic card is not powerful enough or because of the garbage collector in Java. 3 possible ways to improve this in the descending order of our preference:

1)Use C++ with OPENGL instead of Java since C++ is supposed to be more efficient than Java

2) Figure out ways to improve on the performance of your java program until it is satisfacotry

3 Export each frame into a lossless image and make a lossless video from the set of images. You may softwares such as FFMPEG. Take note that you need to ensure pixel level perfection in generating video files

(Optional)Question 2

Write a 3D user interface such as in https://www.youtube.com/watch?v=7wK-zo66GdY

Model the human body posture as a skeleton defined by the position of the several key joints. Your program will be able to accept the position parameters of human joints in real time, and show the human posture in 3D animation. You could store the position parameters in a text file and read the data in real time.