Student ID ______________________________

Spring 2017
CAP 4730 Computer Graphics
Exam II

Instructions:

1. Write your name and UF ID on the cover of the blue exam booklet provided. Write your UF ID on each page of your exam in the space provided.
2. The exam is exactly 50 minutes long.
3. **Answer all 3 questions.**
4. Write your answers in the blue exam booklet provided. Additional booklets are available if necessary. Clearly indicate which questions you are answering on the answer sheets in the exam booklet.
5. Do not leave the room until the exam is over.
6. Raise your hand if you have a question. **Do not get out of your seat for any reason** unless you have permission.
7. At the end of the 50 minutes, the proctor will announce that the exam is over. At that time, **stop writing**, otherwise, your exam will not be accepted. Turn back to the front page, make sure all of your answer sheets are labeled and included in your exam.
8. You are permitted one $8\frac{1}{2} \times 11$ sheet of paper with notes on both sides.
9. Calculators are permitted but they are not necessary to complete the exam. No other wireless devices or devices with image-based memory are permitted.

Student ID ________________________________

Image Processing (12 pts)

Sampling and reconstruction methods can be used to represent continuous functions using a finite set of elements. Images for example, are stored using a finite set of bits as the value of a function at fixed points. The *in between* values may be reconstructed as needed.

1. The diagram below depicts a digital audio signal $f(a)$ point sampled at regular intervals h. Each discrete sample (shown as dots on the curve) at each interval (vertical lines intersecting the curve) is stored in a digital file so that they may be retrieved and used to reconstruct the original signal later.

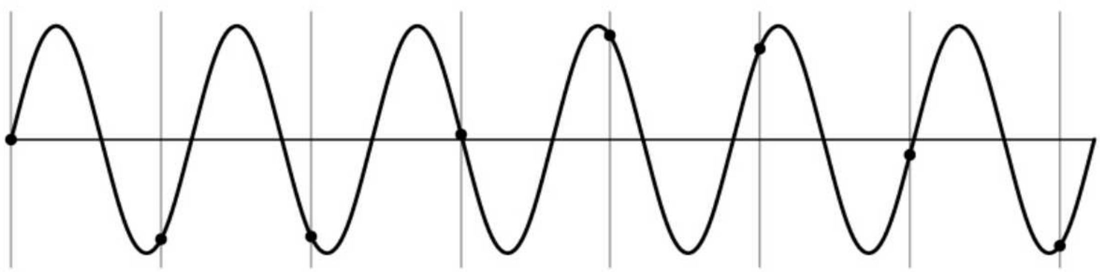


Figure 1: Sampling $f(a)$

   (a) Explain one problem with this sampling choice? Your answer should state the problem and any adverse effect it has.

   (b) What would you do differently to correct the problem?

   (c) How might this same sampling issue cause problems in images?

2. Consider each statement a, b and c below. In each case (a, b and c), name **AND** draw the profile curve of the filter **F** you would use to reconstruct a 1-D function $g(a)$ from a set of discrete samples $s[i]$ so that the statement holds true. Assume that your profile curve is centered at the pt(0,0) in the xy plane and drawn over the interval $-r$ to r where r is the natural radius of the filter.

   (a) $g(a)$ is the interpolation of $s[i]$

   (b) **F** reproduces $g(a)$ using a piecewise linear function with C^0 continuity.

   (c) **F** reproduces $g(a)$ with a function that has C^1 continuity.

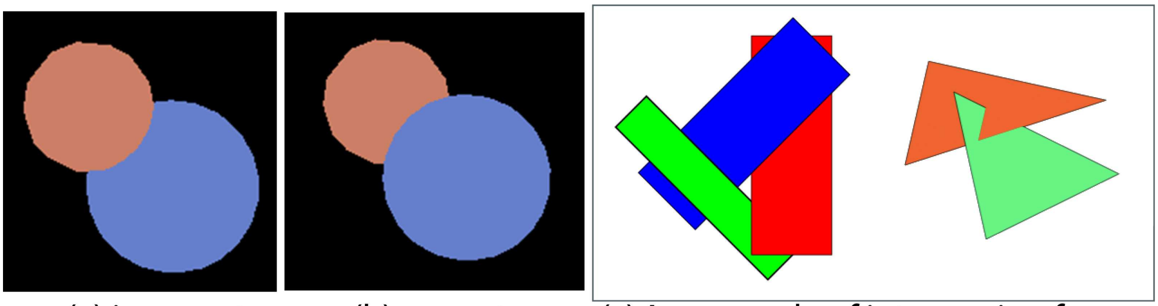
3. You are asked to generate a $500x500$ pixel image from a $5,000x5,000$ pixel image.

   (a) Describe the process you would use to *downsample* your image. In your answer, be sure to explain the key steps in the process including how you would compute the position and value of each new pixel from your original source image. Name any relevant functions required to complete the task.

Student ID ________________________________

The Graphics Pipeline (16 pts)

The graphics pipeline is a series of operations that convert 3-D geometry to 2-D images.



(a) incorrect (b) correct (c) An example of intersecting faces

Figure 2:

1. You are implementing a graphics pipeline to display a blue sphere and a reddish sphere. From your current view, the blue sphere is positioned in front of the reddish sphere. The first image you produce, Figure 2(a), is incorrect (Figure 2(b)is correct).

   (a) What information about your spheres is missing and needs to be processed in the rasterization and fragment processing stages? Assume all matrix transformations (viewing, modeling and others) in your system are correct.

   (b) You implement an algorithm to solve the problem and the spheres are displayed correctly (Figure 2(b)). However, your algorithm is not able to correctly generate scenes with intersecting faces like the ones in Figure 2(c). What classic algorithm did you implement? Why is it unable to generate intersecting faces?

   (c) Name and describe an alternative algorithm that fixes the problem in Figure 2 (a) and solves the intersecting faces problem. Your answer should include the key steps of the algorithm and explain why it is expected to work.

2. Alpha compositing is added. The pipeline can generate a foreground image **f**, and a background image **b**, and produce a composite image **c** where **c** = (**f** over **b**). The foreground and background are opaque, and each image stores 4 values ranging from $[0, 1]$ (over 256 possible values) at each pixel. Assume pre-multiplied alpha.

   (a) If 40% of pixel i in **c** is occupied by a 100% blue object from **f** and the rest of the pixel is occupied by 100% red object from **b**, compute the values that should be stored at each of the 4 channels at pixel i.

   (b) At what stage in the graphics pipeline would you implement the blending operation that determines the final color at pixel i ?

   (c) You save your image to disk using a file format that stores transparency values as an alpha channel. What is the minimum number of bits per pixel you will need to store your color image **c**? Please state what information will be stored and how the bits are allocated.

Gamma Correction (8 pts)

All modern computer monitors are given a γ (gamma) value which is used to convert the digital information stored at a pixel into an intensity for display.

1. You have a dual monitor set-up at your desk. Let **A** and **B** represent your monitors. **A** has $\gamma = x$ and **B** has $\gamma = 2x$. Assume both **A** and **B** have the same maximum intensity and produce no intensity when pixels are switched off. Also assume that both monitors expect each pixel channel of an image to store values from 0 to 1 over 256 possible values. You display the same image simultaneously on **A** and **B**. The only difference is that on **A** you display the image with a $\gamma = 1$ and on **B** you use $\gamma > 2x$. In each case below (a and b), state which image is darker. For each case give an explanation for your answer.
   (a) The image on **A** or the image on **B**
   (b) An image displayed on **B** with $\gamma = 2x$ or a the image currently displayed on **B**.
2. What differences would you expect to see if the image is displayed at the correct gamma setting on each monitor? Explain your answer.
3. You decide to add a third monitor **C** to your set up. The only problem is you do not know its γ value. Assume you have photo editing software installed on **A**. Your photo editing software is weird. It only has a color palette of black, white and shades of grey. It only allows you to create two types of images; images of a single color using a flood fill operation. Or images of two colors in any pattern you like. Can you devise an experiment to roughly estimate γ for **C** using **A**, **C** and your very limited photo editor.