Mipmaps

1. Mipmap Levels

Finding the Mipmap Level
To compute which mipmap level to use at run-time, calculate

p = max(sqrt((∂s/∂x)^2+(∂t/∂x)^2), sqrt((∂s/∂y)^2+(∂t/∂y)^2))

where the derivatives are just approximated by one-sided differences such as
$\frac{\partial s}{\partial x} \approx \frac{s_2 - s_1}{x_2 - x_1}$ and $\frac{\partial s}{\partial y} \approx \frac{s_2 - s_1}{y_2 - y_1}$ and $\frac{\partial t}{\partial x} \approx \frac{t_2 - t_1}{x_2 - x_1}$ and $\frac{\partial t}{\partial y} \approx \frac{t_2 - t_1}{y_2 - y_1}$

The mipmap level is then found by computing:
λ=min(maxLOD, max($\log_2$(p)),minLOD))

a. Imagine we are using mipmaps and we have the following (s,t) coordinates for two fragments:
x=0,y=0: s=0,t=2
x=1,y=1 s=3, t=6

If we are using the NEAREST_MIPMAP_NEAREST mode, what mipmap level(s) will be used to produce the fragment color?

b. Suppose λ=1.25 and the fragment being colored has texture coordinates (s,t)=(5.75, 10.25) with the following greyscale color values in texel neighborhoods:

Texture Level:1	Texture Level: 2
T(5,10)=0.25 T(6,10)=0.75 T(6,11)=0.25 T(5,11)=0,75	T(5,10)=1.0 T(6,10)=1.0 T(6,11)=1.0 T(5,11)=0.5

What is the result of using the NEAREST_MIPMAP_LINEAR mode?

2. Minification

Create a mipmap for the greyscale texture shown here.

0.5	0.5	0.0	1.0
0.0	0.0	1.0	1.0
1.0	0.5	0.25	0.25
1.0	0.5	0.25	0.25

3. Size

If the original texture requires A bytes of storage, find an upper bound on how much space the mipmaps will require in terms of A. You may want to recall that for a number r where 0 < r < 1 we have

$$1 + r + r^2 + r^3 + \cdots = \frac{1}{1-r},$$