Assignment 3 :: Heat GL 3D

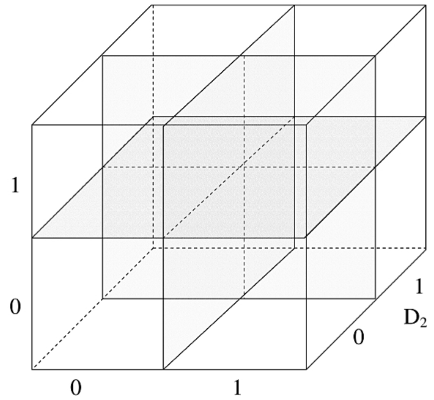
Task 1: Viewing Heat Dynamics in OpenGL

Visualize the behaviour of heat distribution in an object in OpenGL. The governing behaviour of heat spread is given by the equation

$$\frac{dU}{dt}=\alpha\frac{d^2U}{dx^2}$$

where U is the energy at a point in an object U(i, j, k) and it changes over time, α is the thermal diffusivity given as α = k / (c ρ), k is the thermal conductivity, c is the heat capacity, and ρ is the density of a material. The values for these constants can be obtained from the following table:

Material	k ($W\ m^{-1} K^{-1}$)	c ($J\ g^{-1} K^{-1}$)	ρ ($K\ gm^{-3}$)
Air	0.026	1.0035	1.184
Water	0.6089	4.1813	997.0479
Concrete	0.92	0.880	2400
Copper	384.1	0.385	8940
Diamond	895	0.5091	3500



If we consider that the object is a cube, then the cells belonging to the region (0,0,0) would be calculate U(i,j,k) in the following finite difference form:

$$U(i,j,k)=U(i,j,k)+\frac{\Delta t\,\alpha}{\Delta x^2\,\Delta^2\,\Delta z^2}\begin{pmatrix} U(i+2,j,k)-2*U(i+1,j,k)+U(i,j,k)\\ +U(i,j+2,k)-2*U(i,j+1,k)+U(i,j,k)\\ +U(i,j,k+2)-2*U(i,j,k+1)+U(i,j,k)\end{pmatrix}$$

Likewise, cells belonging to the region (1,0,0) would calculate U(i,j,k) as:

$$U(i,j,k)=U(i,j,k)+\frac{\Delta t\,\alpha}{\Delta x^2\,\Delta^2\,\Delta z^2}\begin{pmatrix} U(i,j,k)-2*U(i-1,j,k)+U(i-2,j,k)\\ +U(i,j+2,k)-2*U(i,j+1,k)+U(i,j,k)\\ +U(i,j,k+2)-2*U(i,j,k+1)+U(i,j,k)\end{pmatrix}$$

cells belonging to region (1,1,1) would calculate U(i,j,k) as:

$$U(i,j,k)=U(i,j,k)+\frac{\Delta t\,\alpha}{\Delta x^2\,\Delta^2\,\Delta z^2}\begin{pmatrix} U(i,j,k)-2*U(i-1,j,k)+U(i-2,j,k)\\ +U(i,j,k)-2*U(i,j-1,k)+U(i,j-2,k)\\ +U(i,j,k)-2*U(i,j,k-1)+U(i,j,k-2)\end{pmatrix}$$

and so on ….

To accomplish the task, do the following in sequence:

STEP 1: Prepare Geometry and Object

Create an Object which is broken down into multiple discrete objects. E.g., a cube can be broken down into smaller cubes. Visualize it for correctness. You should have variables which store the number of blocks along X, Y, and Z direction, along with the size of the smallest block, and the size of the larger block. You should also construct a 3D array called U to store the energy of each cell.

STEP 2: U → Color Mapping

Devise a strategy to convert the U values (floating point numbers) to R, G, B color values. If you want to normalize the values between 0 and 1, do so on the R, G, B, not on the U array directly. Cooler temperatures can appear in blue whereas hotter temperatures in red.

STEP 3: Heat Simulation

Perform the finite difference equations in the timer function. Before the equation executes, apply heat at any point U[your choice][your choice][your choice] (or set of points of your choice). You can this heat for a longer period (maybe indefinitely), or for a smaller period. Your choice.

<u>Include some rotations of the camera so you can see the heating effect from all sides.</u>

Deliverables

I need exactly 1 source-code named as follows:

12P-1234_and_12P-4321-task1.c

Work in groups of TWO persons.

Remember that I use Linux. I should be able to compile your code.

Note: Do the code yourselves. Use only concepts studied in class. If your code has more than 70% similarity with other students, there will be penalty marks applied.