Matrix Program

COSC-230
Assignment

Stephen Marz

Topics

- Assignment
- Requirements
- Plagiarism
- Submission

Task

- You will be writing four assembly functions that control the scaling and translation of a 3D matrix and vector.

```
Matrix scale(const Matrix &orig,
             float sx, float sy, float sz);

Matrix translate(const Matrix &orig,
                 float tx, float ty, float tz);

Vector mul(const Matrix &m, const Vector &v);

Vector normalize(const Vector &v);
```

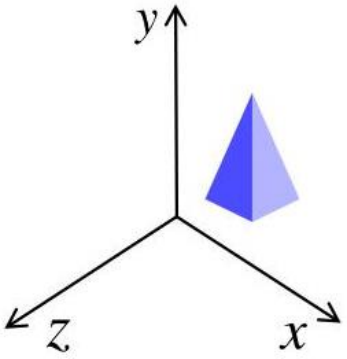
C++ Structures

```
// A 1x4 vector
struct Vector {
    float v[4];
};

// A 4x4 matrix
struct Matrix {
    float m[16];
};
```

Scaling

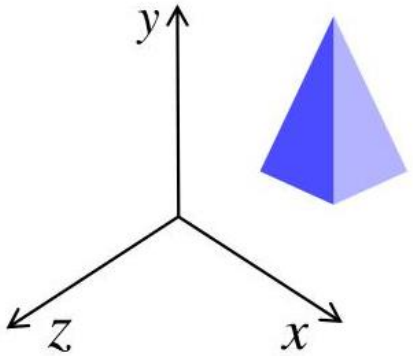
- Scaling modifies the "diagonal" of a 4x4 matrix to scale pixels. The scaling is a coefficient of the vectors passed through it.



$$x' = x \cdot S_x$$

$$y' = y \cdot S_y$$

$$z' = x \cdot S_z$$



Enlarging object also moves it from origin

$$\mathbf{P}' = \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \mathbf{S} \cdot \mathbf{P}$$

C++ Scale

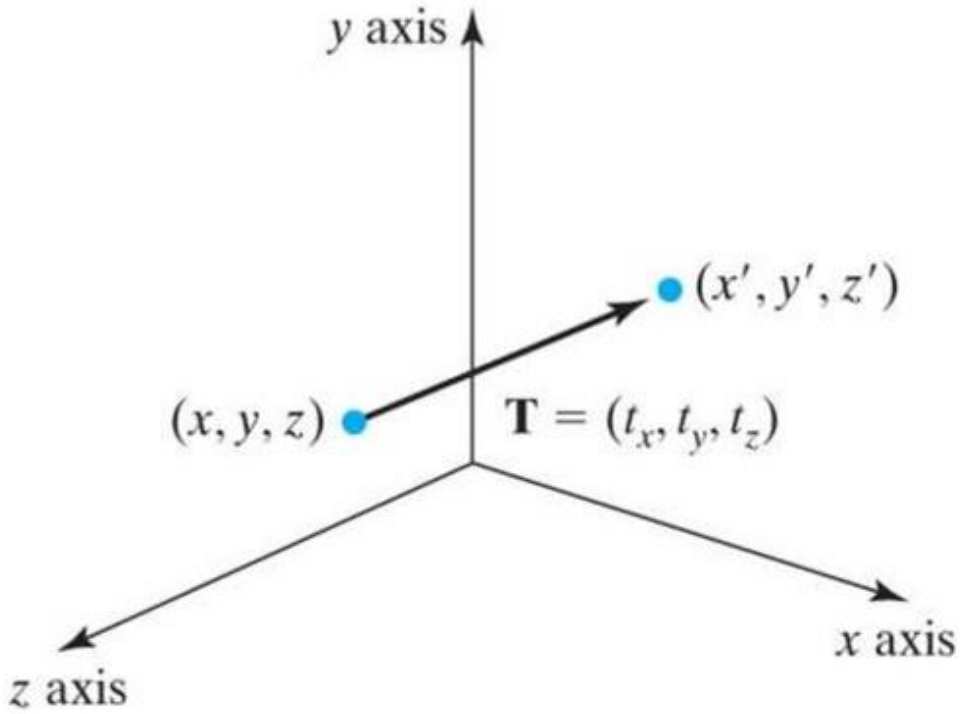
```
Matrix scale(const Matrix &orig,
             float sx, float sy, float sz)
{
    Matrix ret = orig;

    ret.m[0]  *= sx;
    ret.m[5]  *= sy;
    ret.m[10] *= sz;

    return ret;
}
```

Translation

- Translation moves the vectors from <x,y,z> to <x',y',z'>.



$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\mathbf{P}' = \mathbf{T} \cdot \mathbf{P}$$

C++ Translate

```
Matrix translate(const Matrix &orig,
                 float tx, float ty, float tz)
{
    Matrix ret = orig;

    ret.m[3] += tx;
    ret.m[7] += ty;
    ret.m[11] += tz;

    return ret;
}
```

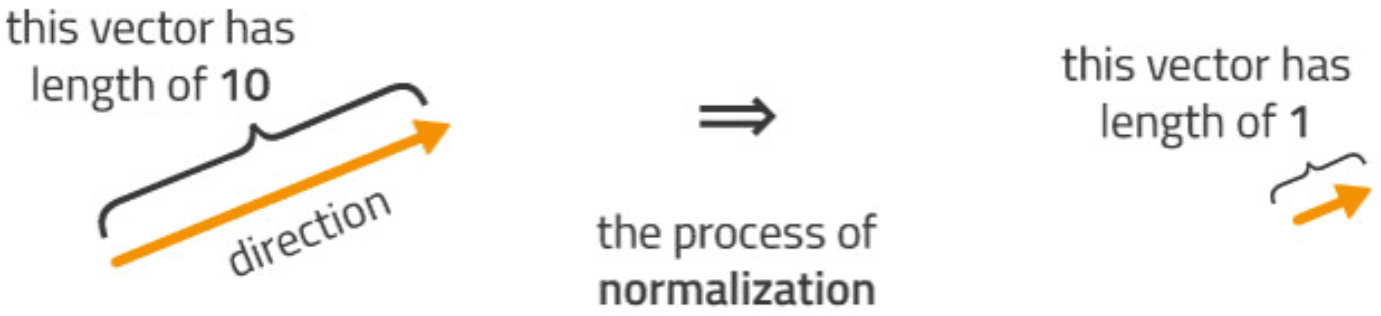
C++ Matrix Multiply

```
Vector mul(const Matrix &m, const Vector &v)
{
    Vector ret;
    for (int row = 0;row < 4;row++) {
        ret.v[row] = 0.0F;
        for (int col = 0;col < 4;col++) {
            ret.v[row] += v.v[row] *
                          m.m[4 * row + col];
        }
    }
    return ret;
}
```

Normalization

- Normalization scales all coordinates of a vector between [-1.0..1.0] while maintaining relative scale between the coordinates.

- Normalization uses the Pythagorean theorem $c^2 = a^2 + b^2$ to calculate the normalization unit.



C++ Normalization

```
Vector norm(const Vector &v)
{
    Vector ret;
    float unit = 0;
    for (int i = 0;i < 4;i++) {
        unit += v.v[i] * v.v[i];
    }
    unit = sqrt(unit);
    for (int i = 0;i < 4;i++) {
        ret.v[i] = v.v[i] / unit;
    }
    return ret;
}
```

Requirements

- You must properly use the stack and ABI calling conventions.
- You must use two nested for loops for the mul function.
- Use shifting when scaling by powers of two.
- Use numeric labels for your loops.
- Use the ABI names for registers
  - ABI names: t0, a0, s0, etc.
  - Index names: x10, x15, x20, etc.

Plagiarism Policy

- This is an **individual assignment**.
- You must NOT be able to see anyone else's code.
- Do NOT send your code and do not accept someone sending you code.
- Do NOT use any online source, such as Chegg, Stackoverflow, etc.
- You MAY use the online notes that I have created for you.
- You MUST cite anyone with whom you worked with, including classmates, students in another class, professors, and TAs.
  - Please note that even if you cite another student, professor, or TA, it does NOT mean you may share code.
- If you cannot attest to the truthfulness of not cheating using the bullets above. DO NOT submit your code. It is better just to get a 0 here and let it be done. If you proceed with copied code, the office of Student Conduct and Community Standards (SCCS) will become involved.

Submission

- Make sure your code compiles and assembles with the following command.

```
~> riscv64-unknown-linux-gnu-g++ -o lab lab.cpp lab.S
~> ./lab
```

- Replace **lab** with the name of your lab.
  - Make sure you have comments in your code, including a header and inline comments.
  - Submit only your .S file.

Topics

- Assignment
- Requirements
- Plagiarism
- Submission

Matrix Program

Stephen Marz

COSC-230

Assignment

THE UNIVERSITY OF
TENNESSEE
KNOXVILLE