**1.1.1**

**a)**

Matrix multiplication corresponds to the composition of transformations. Each matrix represents a linear transformation of space, and multiplying matrices "chains" these transformations together into a single operation. The resulting matrix represents the combined effect of all the transformations applied in order.

For example:

* **Rotation followed by translation**: First rotates a shape, then shifts it in space.
* **Scaling followed by reflection**: First enlarges or shrinks a shape, then reflects it across an axis.

**Geometric Meaning of Each Transformation:**

* **Rotation**: Rotates a vector around a point (often the origin) in either an anticlockwise or clockwise direction.
* **Scaling**: Multiplies a vector by a scaling factor, changing its size; if negative, it can also reverse the vector's direction.
* **Translation**: Shifts all points by the same amount in a specified direction.
* **Reflection**: Reflects a vector across a specified axis, typically by scaling one dimension by a negative factor.

**Inverse of Each Transformation:**

* **Rotation**: The inverse is a rotation by the same angle in the opposite direction.
* **Scaling**: The inverse of scaling by factors (sx,sy,sz)(s\_x, s\_y, s\_z)(sx​,sy​,sz​) is scaling by (1/sx,1/sy,1/sz)(1/s\_x, 1/s\_y, 1/s\_z)(1/sx​,1/sy​,1/sz​).
* **Translation**: The inverse is a translation in the opposite direction.
* **Reflection**: The inverse of a reflection is the same reflection.

**b)**

In general, matrix multiplication is not commutative, meaning that A⋅B≠B⋅A for most matrices A and B. This property holds for transformation matrices as well. However, in some specific cases, certain transformations do commute. Let’s examine each product:

1. **R1⋅R2 ​: Rotation followed by rotation**  
   In general, rotations do **not commute** because the order of rotations around different axes affects the final orientation. For example, rotating around the x-axis followed by the y-axis usually yields a different result than rotating around the y-axis followed by the x-axis. However, **if both rotations are around the same axis**, they do commute (e.g., rotating 30° and then 45° around the z-axis is the same as doing it in the reverse order).
2. **T1⋅T2​: Translation followed by translation**  
   **Translations do commute.** When you apply two translations in succession, the order does not matter, as both translations simply add vectors. For example, translating by (1, 0) followed by (0, 1) has the same result as translating by (0, 1) followed by (1, 0). Therefore, T1⋅T2=T2⋅T1 ​.
3. **S1⋅S2​: Scaling followed by scaling**  
   **Scalings do commute.** When applying two scaling transformations, the order doesn’t matter because scaling is a scalar multiplication applied independently to each axis. For example, scaling by factors of 2 and then 3 is the same as scaling by 3 and then 2. Therefore, S1⋅S2=S2⋅S1.
4. **R⋅T: Rotation followed by translation**  
   **Rotations and translations do not commute.** The effect of rotating an object and then translating it is different from translating the object and then rotating it. Rotation changes the direction of translation, so R⋅T≠T⋅R.
5. **R⋅S: Rotation followed by scaling**  
   **Rotations and scalings do not commute.** Scaling affects the dimensions of the object relative to the coordinate system, so the effect of scaling after a rotation is different from scaling before rotating. For example, if you rotate a shape and then scale it, it scales along the rotated axes, whereas scaling first would apply the scaling along the original axes. Therefore, R⋅S≠S⋅R.
6. **S⋅T: Scaling followed by translation**  
   **Scalings and translations do not commute.** Scaling changes the distances in the coordinate space, so translating after scaling moves an object a different amount than scaling after translating. For example, translating by (1, 1) and then scaling by 2 will not give the same result as scaling first and then translating by (1, 1). Therefore, S⋅T≠T⋅S.

**Explanation for Commuting vs. Non-Commuting**

* **Commuting transformations**: Translation and scaling matrices commute with others of the same type because they do not depend on orientation; they simply add distances or multiply scaling factors, which are independent of order.
* **Non-commuting transformations**: Rotations, translations, and scalings do not commute when combined with different types of transformations because they change the coordinate space in a way that affects how subsequent transformations are applied.

**1.1.2**

**a)**

We can multiply the matrix P from the left by a matrix 3x3 which looks like this:

S =

**b)**

We can multiply P by a column vector of ones, size nx1 to sum each row:

C =

**c)**

The compact expression for the projected distance of each point is:

projected\_distances =

Computation using **e** and **d**:

A black and white math equation

Description automatically generated

We first normalize d to get d\_hat:

Then we will use the equation to find the projected distances:

Znear = approx. 8.16

Zfar = approx. 11.43

**Bonus**

**Given P = T, and if =then:**

**A group of black letters

Description automatically generated**

The reflection matrix N become,

A group of black letters

Description automatically generated

Thus, the matrix N that reflects a vector across the plane with normal n is:

A number of mathematical equations

Description automatically generated with medium confidence