

Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias de la Computación

GRAFICACIÓN

TAREA 4: ROTACIÓN PARALELA Y Y Z EN 3D



Docente:
Prof. Iván Olmos Pineda

Alumno:
Jesús Huerta Aguilar

Matricula:
202041509

NRC: 10592
Sección: 001

CUARTO SEMESTRE

Puebla, Pue.

Fecha de entrega: 04/10/2022

INTRODUCCIÓN

Mover un vértice en el espacio no es una tarea demasiado exigente. Si conocemos sus componentes horizontales (x), vertical (y) y de profundidad (z), para moverlo necesitamos alterar cualquiera de estas componentes según nuestras necesidades. Lo habitual será sumar y restar la cantidad deseada de movimiento en cada componente. Este mecanismo es conocido como translación.

CONCEPTOS DESARROLLADOS

Para rotar un vértice en tres dimensiones, antes hemos de ser capaces de rotar un punto en dos dimensiones.

Conviene destacar que una rotación siempre es una operación referida a un punto, es decir, cuando rotamos un objeto, lo hacemos respecto de algún otro punto en el espacio, en nuestro caso rotaremos respecto al origen de coordenadas.

Las rotaciones se indican en grados o en radianes. Generalmente será más fácil pensar en grados para imaginar las rotaciones, pero las operaciones matemáticas necesitan radianes para obtener resultados correctos. Por tanto, será de gran utilidad un mecanismo para convertir las rotaciones de una escala a otra.

ANÁLISIS EMPIRICO

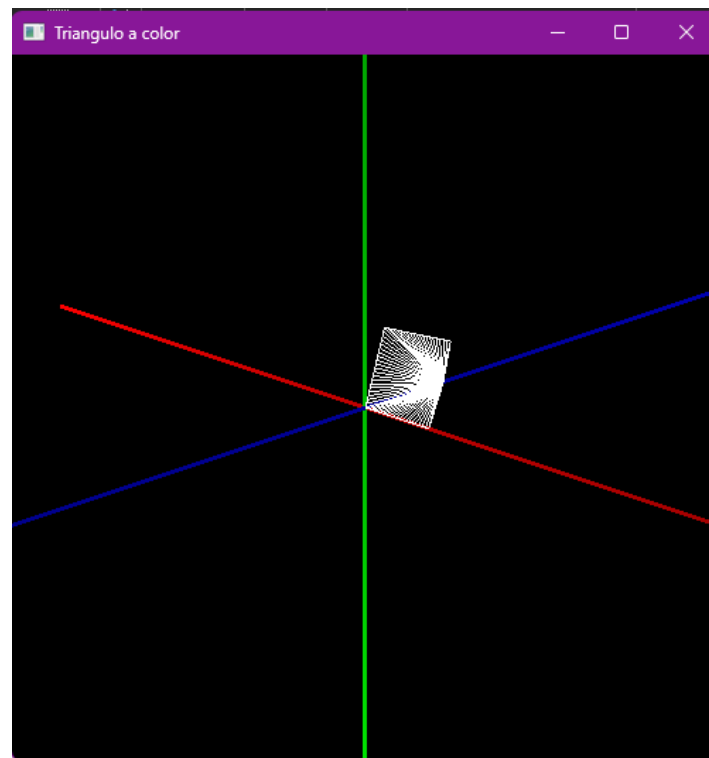
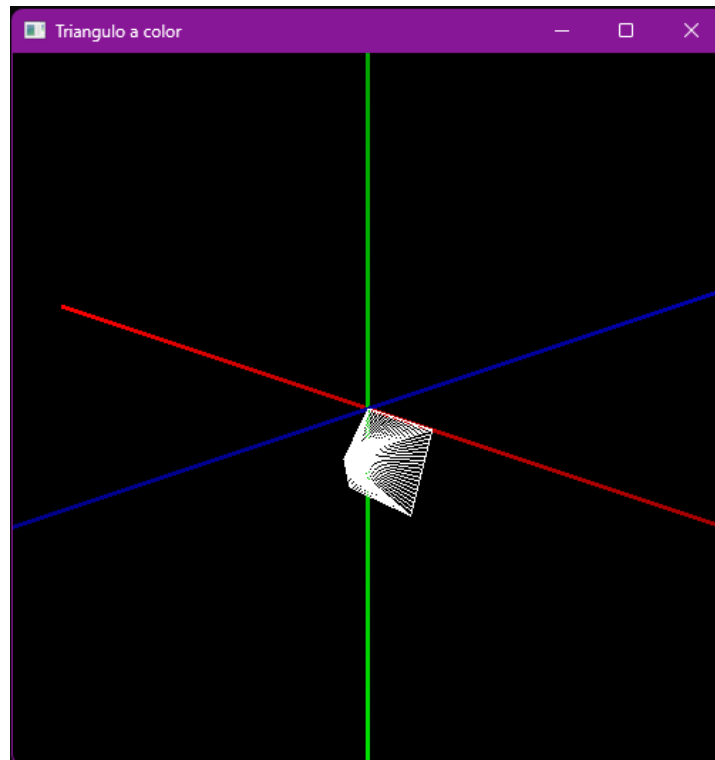
Rotación paralela.

Función que contiene todas las operaciones de rotación para la pirámide.

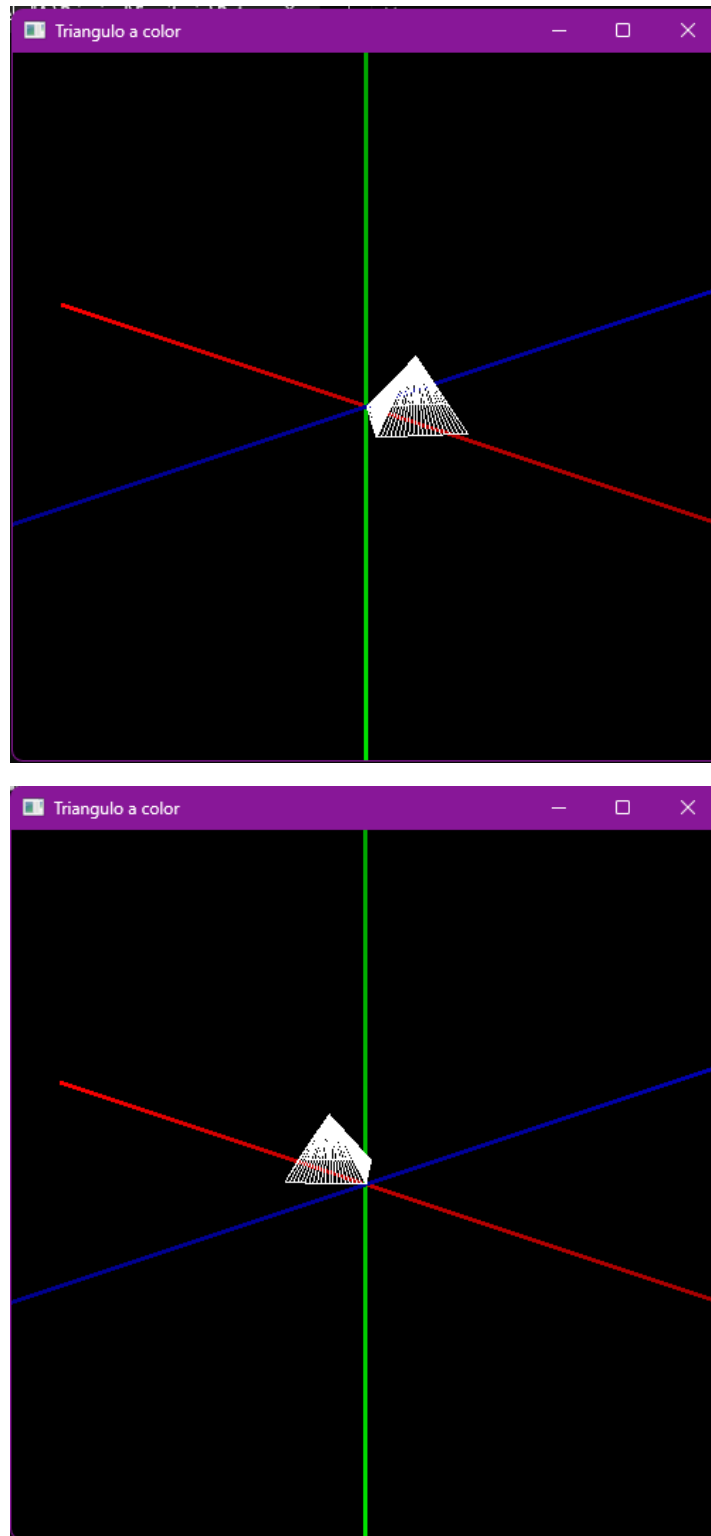
```
1. void Operaciones3D::RotacionParalela(char eje, float theta, float
   distA, float distB)
2. {
3.     switch(eje){
4.         case 'X': //rotacion paralela en "X"
5.             translate(0,-distA,-distB);
6.             rotateX(DegToRad(theta));
7.             MultM(R,T,A);
8.             translate(0,distA,distB);
9.             MultM(T,A,A);
10.            break;
11.         case 'Y': //rotacion paralela en "Y"
12.             translate(0,-distA,-distB);
13.             rotateY(DegToRad(theta));
14.             MultM(R,T,A);
15.             translate(0,distA,distB);
16.             MultM(T,A,A);
17.            break;
18.         case 'Z': //rotacion paralela en "Z"
19.             translate(0,-distA,-distB);
20.             rotateZ(DegToRad(theta));
21.             MultM(R,T,A);
22.             translate(0,distA,distB);
23.             MultM(T,A,A);
24.            break;
25.     }
26. }
```

EJECUCIONES

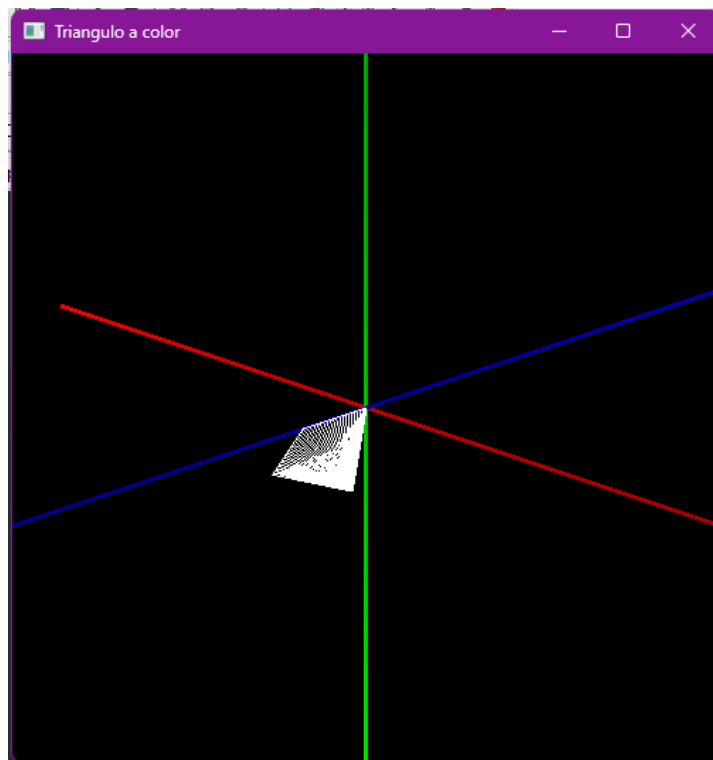
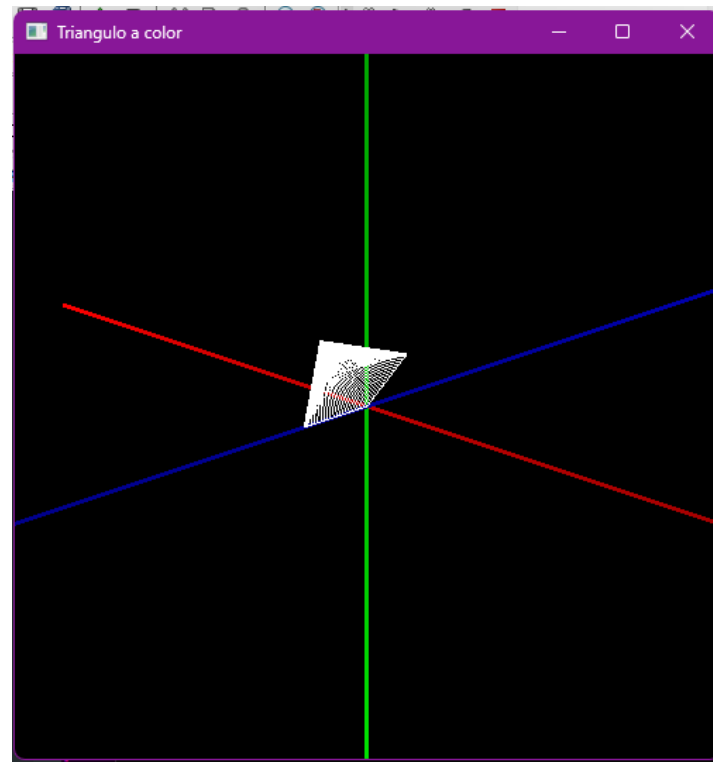
EJE X:



EJE Y:



EJE Z:



CONCLUSIONES

Aprendí el uso de las matrices para rotar en tres dimensiones la pirámide dada por default en el código del profesor, usando las mismas funciones en la clase 3D_BIB pero reestructurándolas para dar un nuevo sentido para el objetivo de la tarea, que es, rotar la pirámide en el eje Y y Z. además de experimentar con el código, por ejemplo, que gire más rápido o modificando el tamaño de la pirámide.

BIBLIOGRAFIA

- Olmos, I. (2022b, octubre 4). Reunión en «General». sharepoint. Recuperado 4 de octubre de 2022, de https://correobuap.sharepoint.com/sites/Section_202235-CCOS261-10592-001/Shared%20Documents/Forms/AllItems.aspx?FolderCTID=0x01200042EC72C454FA1E45805A79EC6BAB0F21&isAscending=false&sortField=Modified&id=%2Fsites%2FSection_202235-CCOS261-10592-001%2FShared%20Documents%2FGeneral%2FRecordings%2FReunión%20en%20_General_-20221004_071146-Grabación%20de%20la%20reunión%20Emp4&viewid=e5d04394-2fba-4809-9262-9608cc0326d4&parent=%2Fsites%2FSection_202235-CCOS261-10592-001%2FShared%20Documents%2FGeneral%2FRecordings