**Laporan UTS Grafkom**

**Tema : Military**

****

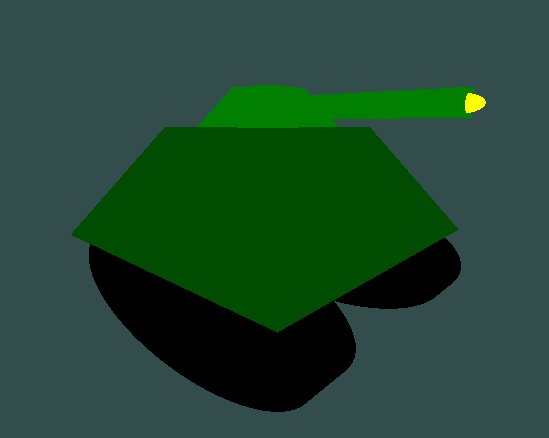
Anggota Kelompok 19:

Sebastian Sutanto (C14200125)

Cliff Leonard (C14190112)  
Jeremy Amadeus Raul Wibisono (C14200113)

**Sebastian Sutanto**

**Objek : Tank**



**Tipe Objek yang Digunakan :**

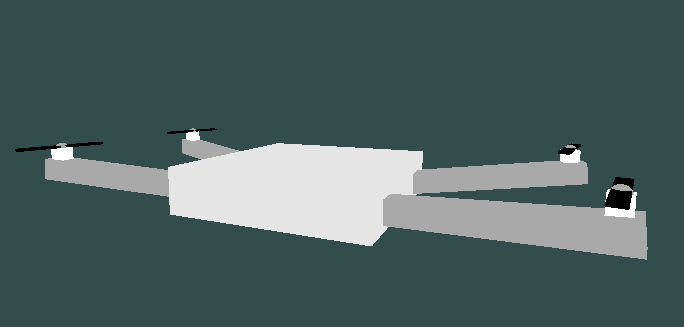
1. Tabung (2 pcs) yang bagian datar-nya mengarah secara horizontal sebagai roda dari tank.
2. Trapesium besar yang menjadi bagian badan dari tank tersebut dengan skala 0,4.
3. Trapesium kecil yang menjadi bagian kepala dari tank tersebut dengan skala 0,275.
4. Persegi panjang sebagai moncong tembakan dari tank tersebut dengan skala 0,11.
5. Ellipsoid kecil terletak di ujung moncong tank sebagai peluru yang akan ditembakan.

**Bagian Yang Dianimasikan :**

Bagian kepala dari tank, moncong tembakannya, serta peluru akan melakukan rotasi melawan arah jarum jam terhadap sumbu Y. Kepala memiliki titik tengah trapesium sebagai titik putarnya sedangkan moncong tembakan dan peluru berotasi sesuai dengan titik tengah kepalanya.

**Jeremy Amadeus Raul Wibisono**

**Objek : Drone**



**Tipe Objek yang Digunakan :**

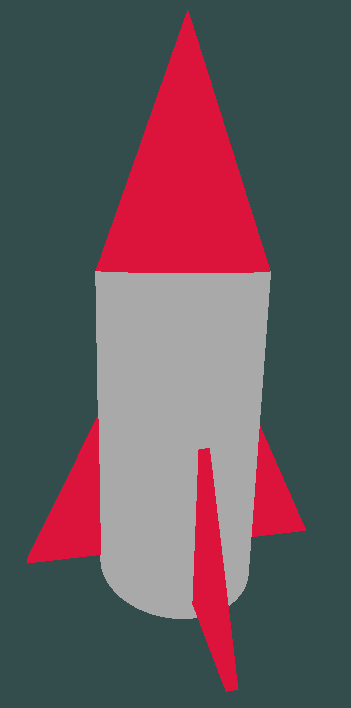
1. Ellipsoid (4 pcs) sebagai titik tengah setiap baling-baling dengan skala 0,015.
2. Persegi Panjang (4 pcs) sebagai ekstensi dari drone dari badan utamanya dan memiliki rotasi 15 derajat dari sumbu Y serta dengan rotasi 5.5 derajat dari sumbu Z dan dengan skala 0,11.
3. Kubus (4 pcs) sebagai bagian letak dari baling-baling drone dan memiliki rotasi 15 derajat dari sumbu Y dan dengan skala 0,035.
4. Persegi Panjang tipis (4 pcs), sebagai baling-baling dari drone tersebut dengan skala 0,05.
5. Kubus yang digepengkan sebagai badan utama dari drone tersebut dengan skala 0,5.

**Bagian Yang Dianimasikan :**

Bagian yang di animasikan adalah baling-baling dari drone tersebut. Baling-baling tersebut memiliki titik rotasi pada tengah baling-balingnya dan berotasi terhadap sumbu Y searah jarum jam.

**Cliff Leonard**

**Objek :Rocket**



**Tipe Objek yang Digunakan :**

1. Tabung raksasa sebagai badan utama roket dan memiliki rotasi 50 derajat dari sumbu X.
2. Prisma (4 pcs) sebagai sirip dari roket tersebut dan memiliki rotasi 50 derajat dari sumbu X dan dengan skala 1,0.
3. Kerucut sebagai bagian atas dari seluruh roket tersebut dan memiliki rotasi 50 derajat dari sumbu X.

**Bagian Yang Dianimasikan :**

Seluruh roket memiliki animasi berotasi terhadap sumbu X. Rotasi tersebut dilakukan berdasarkan titik tengah rotasi yang berada diluar badan roket tersebut.

**Kode yang digunakan untuk membuat objek-objek tersebut:**

**Kode untuk Ellipsoid:**

public void createEllipsoid2(float radiusX, float radiusY, float radiusZ, float \_x, float \_y, float \_z, int sectorCount, int stackCount)

{

float pi = (float)Math.PI;

Vector3 temp\_vector;

float sectorStep = 2 \* (float)Math.PI / sectorCount;

float stackStep = (float)Math.PI / stackCount;

float sectorAngle, StackAngle, x, y, z;

for (int i = 0; i <= stackCount; ++i)

{

StackAngle = pi / 2 - i \* stackStep;

x = radiusX \* (float)Math.Cos(StackAngle);

y = radiusY \* (float)Math.Cos(StackAngle);

z = radiusZ \* (float)Math.Sin(StackAngle);

for (int j = 0; j <= sectorCount; ++j)

{

sectorAngle = j \* sectorStep;

temp\_vector.X = \_x + x \* (float)Math.Cos(sectorAngle);

temp\_vector.Y = \_y + y \* (float)Math.Sin(sectorAngle);

temp\_vector.Z = \_z + z;

\_vertices.Add(temp\_vector);

}

}

uint k1, k2;

for (int i = 0; i < stackCount; ++i)

{

k1 = (uint)(i \* (sectorCount + 1));

k2 = (uint)(k1 + sectorCount + 1);

for (int j = 0; j < sectorCount; ++j, ++k1, ++k2)

{

if (i != 0)

{

\_indices.Add(k1);

\_indices.Add(k2);

\_indices.Add(k1 + 1);

}

if (i != (stackCount - 1))

{

\_indices.Add(k1 + 1);

\_indices.Add(k2);

\_indices.Add(k2 + 1);

}

}

}

}

**Kode untuk Kubus:**

public void createBoxVertices(float x, float y, float z, float length)

{

Vector3 temp\_vector;

//TITIK 1

temp\_vector.X = x - length / 2.0f;

temp\_vector.Y = y + length / 2.0f;

temp\_vector.Z = z - length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 2

temp\_vector.X = x + length / 2.0f;

temp\_vector.Y = y + length / 2.0f;

temp\_vector.Z = z - length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 3

temp\_vector.X = x - length / 2.0f;

temp\_vector.Y = y - length / 2.0f;

temp\_vector.Z = z - length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 4

temp\_vector.X = x + length / 2.0f;

temp\_vector.Y = y - length / 2.0f;

temp\_vector.Z = z - length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 5

temp\_vector.X = x - length / 2.0f;

temp\_vector.Y = y + length / 2.0f;

temp\_vector.Z = z + length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 6

temp\_vector.X = x + length / 2.0f;

temp\_vector.Y = y + length / 2.0f;

temp\_vector.Z = z + length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 7

temp\_vector.X = x - length / 2.0f;

temp\_vector.Y = y - length / 2.0f;

temp\_vector.Z = z + length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 8

temp\_vector.X = x + length / 2.0f;

temp\_vector.Y = y - length / 2.0f;

temp\_vector.Z = z + length / 2.0f;

\_vertices.Add(temp\_vector);

\_indices = new List<uint>

{

//SEGITIGA DEPAN 1

0,1,2,

//SEGITIGA DEPAN 2

1,2,3,

//SEGITIGA ATAS 1

0,4,5,

//SEGITIGA ATAS 2

0,1,5,

//SEGITIGA KANAN 1

1,3,5,

//SEGITIGA KANAN 2

3,5,7,

//SEGITIGA KIRI 1

0,2,4,

//SEGITIGA KIRI 2

2,4,6,

//SEGITIGA BELAKANG 1

4,5,6,

//SEGITIGA BELAKANG 2

5,6,7,

//SEGITIGA BAWAH 1

2,3,6,

//SEGITIGA BAWAH 2

3,6,7

};

}

**Kode untuk Kubus yang digepengkan:**

public void createFlatBoxVertices(float x, float y, float z, float length)

{

Vector3 temp\_vector;

//TITIK 1

temp\_vector.X = x - length / 2.0f;

temp\_vector.Y = y + length / 16.0f;

temp\_vector.Z = z - length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 2

temp\_vector.X = x + length / 2.0f;

temp\_vector.Y = y + length / 16.0f;

temp\_vector.Z = z - length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 3

temp\_vector.X = x - length / 2.0f;

temp\_vector.Y = y - length / 6.0f;

temp\_vector.Z = z - length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 4

temp\_vector.X = x + length / 2.0f;

temp\_vector.Y = y - length / 6.0f;

temp\_vector.Z = z - length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 5

temp\_vector.X = x - length / 2.0f;

temp\_vector.Y = y + length / 16.0f;

temp\_vector.Z = z + length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 6

temp\_vector.X = x + length / 2.0f;

temp\_vector.Y = y + length / 16.0f;

temp\_vector.Z = z + length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 7

temp\_vector.X = x - length / 2.0f;

temp\_vector.Y = y - length / 6.0f;

temp\_vector.Z = z + length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 8

temp\_vector.X = x + length / 2.0f;

temp\_vector.Y = y - length / 6.0f;

temp\_vector.Z = z + length / 2.0f;

\_vertices.Add(temp\_vector);

\_indices = new List<uint>

{

//SEGITIGA DEPAN 1

0,1,2,

//SEGITIGA DEPAN 2

1,2,3,

//SEGITIGA ATAS 1

0,4,5,

//SEGITIGA ATAS 2

0,1,5,

//SEGITIGA KANAN 1

1,3,5,

//SEGITIGA KANAN 2

3,5,7,

//SEGITIGA KIRI 1

0,2,4,

//SEGITIGA KIRI 2

2,4,6,

//SEGITIGA BELAKANG 1

4,5,6,

//SEGITIGA BELAKANG 2

5,6,7,

//SEGITIGA BAWAH 1

2,3,6,

//SEGITIGA BAWAH 2

3,6,7

};

}

**Kode untuk Persegi panjang:**

public void createLongBoxVertices(float x, float y, float z, float length)

{

Vector3 temp\_vector;

//TITIK 1

temp\_vector.X = x - length \* 2.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z - length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 2

temp\_vector.X = x + length \* 2.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z - length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 3

temp\_vector.X = x - length \* 2.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z - length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 4

temp\_vector.X = x + length \* 2.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z - length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 5

temp\_vector.X = x - length \* 2.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z + length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 6

temp\_vector.X = x + length \* 2.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z + length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 7

temp\_vector.X = x - length \* 2.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z + length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 8

temp\_vector.X = x + length \* 2.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z + length / 4.0f;

\_vertices.Add(temp\_vector);

\_indices = new List<uint>

{

//SEGITIGA DEPAN 1

0,1,2,

//SEGITIGA DEPAN 2

1,2,3,

//SEGITIGA ATAS 1

0,4,5,

//SEGITIGA ATAS 2

0,1,5,

//SEGITIGA KANAN 1

1,3,5,

//SEGITIGA KANAN 2

3,5,7,

//SEGITIGA KIRI 1

0,2,4,

//SEGITIGA KIRI 2

2,4,6,

//SEGITIGA BELAKANG 1

4,5,6,

//SEGITIGA BELAKANG 2

5,6,7,

//SEGITIGA BAWAH 1

2,3,6,

//SEGITIGA BAWAH 2

3,6,7

};

}

**Kode untuk Persegi panjang tipis (Baling-baling):**

public void createFlatLongBoxVertices(float x, float y, float z, float length)

{

Vector3 temp\_vector;

//TITIK 1

temp\_vector.X = x - length \* 2.0f;

temp\_vector.Y = y + length / 16.0f;

temp\_vector.Z = z - length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 2

temp\_vector.X = x + length \* 2.0f;

temp\_vector.Y = y + length / 16.0f;

temp\_vector.Z = z - length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 3

temp\_vector.X = x - length \* 2.0f;

temp\_vector.Y = y - length / 16.0f;

temp\_vector.Z = z - length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 4

temp\_vector.X = x + length \* 2.0f;

temp\_vector.Y = y - length / 16.0f;

temp\_vector.Z = z - length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 5

temp\_vector.X = x - length \* 2.0f;

temp\_vector.Y = y + length / 16.0f;

temp\_vector.Z = z + length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 6

temp\_vector.X = x + length \* 2.0f;

temp\_vector.Y = y + length / 16.0f;

temp\_vector.Z = z + length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 7

temp\_vector.X = x - length \* 2.0f;

temp\_vector.Y = y - length / 16.0f;

temp\_vector.Z = z + length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 8

temp\_vector.X = x + length \* 2.0f;

temp\_vector.Y = y - length / 16.0f;

temp\_vector.Z = z + length / 4.0f;

\_vertices.Add(temp\_vector);

\_indices = new List<uint>

{

//SEGITIGA DEPAN 1

0,1,2,

//SEGITIGA DEPAN 2

1,2,3,

//SEGITIGA ATAS 1

0,4,5,

//SEGITIGA ATAS 2

0,1,5,

//SEGITIGA KANAN 1

1,3,5,

//SEGITIGA KANAN 2

3,5,7,

//SEGITIGA KIRI 1

0,2,4,

//SEGITIGA KIRI 2

2,4,6,

//SEGITIGA BELAKANG 1

4,5,6,

//SEGITIGA BELAKANG 2

5,6,7,

//SEGITIGA BAWAH 1

2,3,6,

//SEGITIGA BAWAH 2

3,6,7

};

}

**Kode untuk Trapezoid kepala tank:**

public void createTrapezoidHeadVertices(float x, float y, float z, float length)

{

Vector3 temp\_vector;

//TITIK 1

temp\_vector.X = x - length / 4.0f;

temp\_vector.Y = y + length / 8.0f;

temp\_vector.Z = z - length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 2

temp\_vector.X = x + length / 4.0f;

temp\_vector.Y = y + length / 8.0f;

temp\_vector.Z = z - length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 3

temp\_vector.X = x - length / 2.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z - length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 4

temp\_vector.X = x + length / 2.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z - length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 5

temp\_vector.X = x - length / 4.0f;

temp\_vector.Y = y + length / 8.0f;

temp\_vector.Z = z + length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 6

temp\_vector.X = x + length / 4.0f;

temp\_vector.Y = y + length / 8.0f;

temp\_vector.Z = z + length / 4.0f;

\_vertices.Add(temp\_vector);

//TITIK 7

temp\_vector.X = x - length / 2.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z + length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 8

temp\_vector.X = x + length / 2.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z + length / 2.0f;

\_vertices.Add(temp\_vector);

\_indices = new List<uint>

{

//SEGITIGA DEPAN 1

0,1,2,

//SEGITIGA DEPAN 2

1,2,3,

//SEGITIGA ATAS 1

0,4,5,

//SEGITIGA ATAS 2

0,1,5,

//SEGITIGA KANAN 1

1,3,5,

//SEGITIGA KANAN 2

3,5,7,

//SEGITIGA KIRI 1

0,2,4,

//SEGITIGA KIRI 2

2,4,6,

//SEGITIGA BELAKANG 1

4,5,6,

//SEGITIGA BELAKANG 2

5,6,7,

//SEGITIGA BAWAH 1

2,3,6,

//SEGITIGA BAWAH 2

3,6,7

};

}

**Kode untuk Trapezoid badan tank:**

public void createTrapezoidBodyVertices(float x, float y, float z, float length)

{

Vector3 temp\_vector;

//TITIK 1

temp\_vector.X = x - length / 2.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z - length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 2

temp\_vector.X = x + length / 2.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z - length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 3

temp\_vector.X = x - length;

temp\_vector.Y = y - length / 2.0f;

temp\_vector.Z = z - length;

\_vertices.Add(temp\_vector);

//TITIK 4

temp\_vector.X = x + length;

temp\_vector.Y = y - length / 2.0f;

temp\_vector.Z = z - length;

\_vertices.Add(temp\_vector);

//TITIK 5

temp\_vector.X = x - length / 2.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z + length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 6

temp\_vector.X = x + length / 2.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z + length / 2.0f;

\_vertices.Add(temp\_vector);

//TITIK 7

temp\_vector.X = x - length;

temp\_vector.Y = y - length / 2.0f;

temp\_vector.Z = z + length;

\_vertices.Add(temp\_vector);

//TITIK 8

temp\_vector.X = x + length;

temp\_vector.Y = y - length / 2.0f;

temp\_vector.Z = z + length;

\_vertices.Add(temp\_vector);

\_indices = new List<uint>

{

//SEGITIGA DEPAN 1

0,1,2,

//SEGITIGA DEPAN 2

1,2,3,

//SEGITIGA ATAS 1

0,4,5,

//SEGITIGA ATAS 2

0,1,5,

//SEGITIGA KANAN 1

1,3,5,

//SEGITIGA KANAN 2

3,5,7,

//SEGITIGA KIRI 1

0,2,4,

//SEGITIGA KIRI 2

2,4,6,

//SEGITIGA BELAKANG 1

4,5,6,

//SEGITIGA BELAKANG 2

5,6,7,

//SEGITIGA BAWAH 1

2,3,6,

//SEGITIGA BAWAH 2

3,6,7

};

}

**Kode untuk Roda tank:**

public void createWheels(float radiusX, float radiusY, float radiusZ, float \_x, float \_y, float \_z, int sectorCount, int stackCount)

{

float pi = (float)Math.PI;

Vector3 temp\_vector;

float sectorStep = 2 \* (float)Math.PI / sectorCount;

float stackStep = (float)Math.PI / stackCount;

float sectorAngle, StackAngle, x, y, z;

for (int i = 0; i <= stackCount; ++i)

{

StackAngle = pi / 2 - i \* stackStep;

x = radiusX \* (float)Math.Cos(StackAngle);

y = radiusY \* (float)Math.Cos(StackAngle);

z = radiusZ \* (i / 2);

for (int j = 0; j <= sectorCount; ++j)

{

sectorAngle = j \* sectorStep;

temp\_vector.X = \_x + x \* (float)Math.Cos(sectorAngle);

temp\_vector.Y = \_y + y \* (float)Math.Sin(sectorAngle);

temp\_vector.Z = \_z + z;

\_vertices.Add(temp\_vector);

}

}

uint k1, k2;

for (int i = 0; i < stackCount; ++i)

{

k1 = (uint)(i \* (sectorCount + 1));

k2 = (uint)(k1 + sectorCount + 1);

for (int j = 0; j < sectorCount; ++j, ++k1, ++k2)

{

if (i != 0)

{

\_indices.Add(k1);

\_indices.Add(k2);

\_indices.Add(k1 + 1);

}

if (i != (stackCount - 1))

{

\_indices.Add(k1 + 1);

\_indices.Add(k2);

\_indices.Add(k2 + 1);

}

}

}

}

**Kode untuk Tabung badan roket:**

public void createTube(float radiusX, float radiusY, float radiusZ, float \_x, float \_y, float \_z, int sectorCount, int stackCount)

{

float pi = (float)Math.PI;

Vector3 temp\_vector;

float sectorStep = 2 \* (float)Math.PI / sectorCount;

float stackStep = (float)Math.PI / stackCount;

float sectorAngle, StackAngle, x, y, z;

for (int i = 0; i <= stackCount; ++i)

{

StackAngle = pi / 2 - i \* stackStep;

x = radiusX \* (float)Math.Cos(StackAngle);

y = radiusY \* (i / 2);

z = radiusZ \* (float)Math.Cos(StackAngle);

for (int j = 0; j <= sectorCount; ++j)

{

sectorAngle = j \* sectorStep;

temp\_vector.X = \_x + x \* (float)Math.Cos(sectorAngle);

temp\_vector.Y = \_y + y;

temp\_vector.Z = \_z + z \* (float)Math.Sin(sectorAngle);

\_vertices.Add(temp\_vector);

}

}

uint k1, k2;

for (int i = 0; i < stackCount; ++i)

{

k1 = (uint)(i \* (sectorCount + 1));

k2 = (uint)(k1 + sectorCount + 1);

for (int j = 0; j < sectorCount; ++j, ++k1, ++k2)

{

if (i != 0)

{

\_indices.Add(k1);

\_indices.Add(k2);

\_indices.Add(k1 + 1);

}

if (i != (stackCount - 1))

{

\_indices.Add(k1 + 1);

\_indices.Add(k2);

\_indices.Add(k2 + 1);

}

}

}

}

**Kode untuk Kerucut kepala roket:**

public void createCone(float radiusX, float radiusY, float radiusZ, float \_x, float \_y, float \_z, int sectorCount, int stackCount)

{

float pi = (float)Math.PI;

Vector3 temp\_vector;

float sectorStep = 2 \* (float)Math.PI / sectorCount;

float stackStep = (float)Math.PI / stackCount;

float sectorAngle, StackAngle, x, y, z;

for (int i = 0; i <= stackCount; ++i)

{

StackAngle = pi / 2 - i \* stackStep;

x = radiusX \* (float)Math.Cos(StackAngle);

y = radiusY \* (float)Math.Cos(StackAngle);

z = radiusZ \* (float)Math.Cos(StackAngle);

for (int j = 0; j <= sectorCount; ++j)

{

sectorAngle = j \* sectorStep;

temp\_vector.X = \_x + x \* (float)Math.Cos(sectorAngle);

temp\_vector.Y = \_y - y;

temp\_vector.Z = \_z + z \* (float)Math.Sin(sectorAngle);

\_vertices.Add(temp\_vector);

}

}

uint k1, k2;

for (int i = 0; i < stackCount; ++i)

{

k1 = (uint)(i \* (sectorCount + 1));

k2 = (uint)(k1 + sectorCount + 1);

for (int j = 0; j < sectorCount; ++j, ++k1, ++k2)

{

if (i != 0)

{

\_indices.Add(k1);

\_indices.Add(k2);

\_indices.Add(k1 + 1);

}

if (i != (stackCount - 1))

{

\_indices.Add(k1 + 1);

\_indices.Add(k2);

\_indices.Add(k2 + 1);

}

}

}

}

**Kode untuk Prisma sirip roket:**

public void createBackPrism(float x, float y, float z, float length)

{

Vector3 temp\_vector;

//TITIK 1

temp\_vector.X = x - length / 64.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z + length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 2

temp\_vector.X = x + length / 64.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z + length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 3

temp\_vector.X = x - length / 64.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z - length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 4

temp\_vector.X = x + length / 64.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z - length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 5

temp\_vector.X = x - length / 64.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z + length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 6

temp\_vector.X = x + length / 64.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z + length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 7

temp\_vector.X = x - length / 64.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z + length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 8

temp\_vector.X = x + length / 64.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z + length / 8.0f;

\_vertices.Add(temp\_vector);

\_indices = new List<uint>

{

//SEGITIGA DEPAN 1

0,1,2,

//SEGITIGA DEPAN 2

1,2,3,

//SEGITIGA ATAS 1

0,4,5,

//SEGITIGA ATAS 2

0,1,5,

//SEGITIGA KANAN 1

1,3,5,

//SEGITIGA KANAN 2

3,5,7,

//SEGITIGA KIRI 1

0,2,4,

//SEGITIGA KIRI 2

2,4,6,

//SEGITIGA BELAKANG 1

4,5,6,

//SEGITIGA BELAKANG 2

5,6,7,

//SEGITIGA BAWAH 1

2,3,6,

//SEGITIGA BAWAH 2

3,6,7

};

}

public void createFrontPrism(float x, float y, float z, float length)

{

Vector3 temp\_vector;

//TITIK 1

temp\_vector.X = x - length / 64.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z - length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 2

temp\_vector.X = x + length / 64.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z - length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 3

temp\_vector.X = x - length / 64.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z - length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 4

temp\_vector.X = x + length / 64.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z - length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 5

temp\_vector.X = x - length / 64.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z - length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 6

temp\_vector.X = x + length / 64.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z - length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 7

temp\_vector.X = x - length / 64.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z + length / 8.0f;

\_vertices.Add(temp\_vector);

//TITIK 8

temp\_vector.X = x + length / 64.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z + length / 8.0f;

\_vertices.Add(temp\_vector);

\_indices = new List<uint>

{

//SEGITIGA DEPAN 1

0,1,2,

//SEGITIGA DEPAN 2

1,2,3,

//SEGITIGA ATAS 1

0,4,5,

//SEGITIGA ATAS 2

0,1,5,

//SEGITIGA KANAN 1

1,3,5,

//SEGITIGA KANAN 2

3,5,7,

//SEGITIGA KIRI 1

0,2,4,

//SEGITIGA KIRI 2

2,4,6,

//SEGITIGA BELAKANG 1

4,5,6,

//SEGITIGA BELAKANG 2

5,6,7,

//SEGITIGA BAWAH 1

2,3,6,

//SEGITIGA BAWAH 2

3,6,7

};

}

public void createRightPrism(float x, float y, float z, float length)

{

Vector3 temp\_vector;

//TITIK 1

temp\_vector.X = x - length / 8.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z - length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 2

temp\_vector.X = x - length / 8.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z - length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 3

temp\_vector.X = x - length / 8.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z - length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 4

temp\_vector.X = x + length / 8.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z - length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 5

temp\_vector.X = x - length / 8.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z + length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 6

temp\_vector.X = x - length / 8.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z + length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 7

temp\_vector.X = x - length / 8.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z + length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 8

temp\_vector.X = x + length / 8.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z + length / 64.0f;

\_vertices.Add(temp\_vector);

\_indices = new List<uint>

{

//SEGITIGA DEPAN 1

0,1,2,

//SEGITIGA DEPAN 2

1,2,3,

//SEGITIGA ATAS 1

0,4,5,

//SEGITIGA ATAS 2

0,1,5,

//SEGITIGA KANAN 1

1,3,5,

//SEGITIGA KANAN 2

3,5,7,

//SEGITIGA KIRI 1

0,2,4,

//SEGITIGA KIRI 2

2,4,6,

//SEGITIGA BELAKANG 1

4,5,6,

//SEGITIGA BELAKANG 2

5,6,7,

//SEGITIGA BAWAH 1

2,3,6,

//SEGITIGA BAWAH 2

3,6,7

};

}

public void createLeftPrism(float x, float y, float z, float length)

{

Vector3 temp\_vector;

//TITIK 1

temp\_vector.X = x + length / 8.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z - length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 2

temp\_vector.X = x + length / 8.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z - length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 3

temp\_vector.X = x - length / 8.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z - length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 4

temp\_vector.X = x + length / 8.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z - length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 5

temp\_vector.X = x + length / 8.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z + length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 6

temp\_vector.X = x + length / 8.0f;

temp\_vector.Y = y + length / 4.0f;

temp\_vector.Z = z + length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 7

temp\_vector.X = x - length / 8.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z + length / 64.0f;

\_vertices.Add(temp\_vector);

//TITIK 8

temp\_vector.X = x + length / 8.0f;

temp\_vector.Y = y - length / 4.0f;

temp\_vector.Z = z + length / 64.0f;

\_vertices.Add(temp\_vector);

\_indices = new List<uint>

{

//SEGITIGA DEPAN 1

0,1,2,

//SEGITIGA DEPAN 2

1,2,3,

//SEGITIGA ATAS 1

0,4,5,

//SEGITIGA ATAS 2

0,1,5,

//SEGITIGA KANAN 1

1,3,5,

//SEGITIGA KANAN 2

3,5,7,

//SEGITIGA KIRI 1

0,2,4,

//SEGITIGA KIRI 2

2,4,6,

//SEGITIGA BELAKANG 1

4,5,6,

//SEGITIGA BELAKANG 2

5,6,7,

//SEGITIGA BAWAH 1

2,3,6,

//SEGITIGA BAWAH 2

3,6,7

};

}