

A2 - Questions?

$$Q(u, v) = \begin{cases} x(u) \cos v \\ x(u) \sin v \\ z(u) \end{cases} \quad 0 \leq v \leq 2\pi$$



$$C(u) = \begin{cases} x(u) \\ y(u) \\ z(u) \end{cases}$$

Bezier $\rightarrow 0 \leq u \leq 1$

de Casteljau

$\rightarrow$ ~~increment~~ # v steps = 10
 $\rightarrow$ # u step = 10

$$\begin{aligned} u_{inc} &= 1.0 / (\# \text{usteps} - 1) \rightarrow 9 \\ v_{inc} &= 2\pi / (\# \text{vsteps} - 1) \rightarrow 0.7 \end{aligned}$$

i=0; j=0

for (u=0; u ≤ 1; u+=uinc)

for (v=0; v ≤ 2π; v+=vinc)

generate(u, v) → store x, y, z into 2D grid.

subdivision

vinc = 2π / (vsteps - 1)
for (i=0; i < curvepoints.size(), i++)
for (v=0; v ≤ 2π; v+=vinc)
generate(curvepoints[i], v)

$$Q(u, v) = \begin{pmatrix} \text{curvepoint}.x \cos v \\ \text{curvepoint}.x \sin v \\ \text{curvepoint}.z \end{pmatrix}$$

Tensor Product Surface

- start with vector $\langle \text{Point} \rangle$ control points.
- "all" you need to do is subdivide the points

→ subdividing rows is easy!

You have → $\text{vector} \langle \text{Point} \rangle \text{subdivideCurve}(\text{vector} \langle \text{Point} \rangle \text{coarseCurve});$

```
for (int i=0; i < controlPoints.size(); i++)  
    controlPoints[i] = subdivideCurve(controlPoints[i]);
```

EASY!

→ columns are trickier

~~new CP grid (new size)~~
for → mention resize, "new cp grid"

for each column:

$\text{vector} \langle \text{Point} \rangle \text{col};$

```
for (i=0; i < controlPoints.size(); i++)
```

$\text{col.pushBack}[i][\text{which column}];$

$\text{col} = \text{subdivideCurve}(\text{col});$

→ then copy col into your new control points grid.

when done replace controlPoints with new control points.

→ just be careful with boundaries!