**Idea Documentation: Retrowave City OpenGL Project**

**Participants: Hussein Galal 221001810**

**Marwan Elsayed 221003166**  
**DR: Hatem Khater**

**TA: Eng/ Bassant yasser**

**1. Project Overview**

**Project Name:** Retrowave City  
**Genre:** 3D Interactive Visual Experience  
**Theme:** Retro-futuristic / Synthwave / Neon Aesthetic  
**Technology Stack:**

* **OpenGL** (for 3D rendering)
* **GLUT** (for window management & basic shapes)
* **Windows Multimedia API** (for audio playback)

**Objective:**  
Create an immersive 3D environment inspired by 1980s retrowave/synthwave aesthetics, featuring:

* Neon-lit cityscapes
* Animated futuristic elements (spinners, tunnels)
* Dynamic lighting & pulsing effects
* Interactive camera controls
* Background synthwave music

**2. Core Features**

**2.1 Visual Elements**

| **Feature** | **Description** |
| --- | --- |
| **Neon Grid Floor** | Infinite scrolling grid with pink/cyan pulsing lines |
| **Floating Buildings** | Procedurally generated skyscrapers with glowing windows |
| **Spinners** | Rotating circular/spiral shapes with neon outlines |
| **Tunnel Effect** | Hypnotic vortex in the sky with depth animation |
| **Cars** | Low-poly vehicles with light trails |
| **Stars** | Twinkling stars in the night sky |
| **Pyramid & Torus** | Additional geometric shapes with dynamic lighting |

**2.2 Audio Integration**

* **MP3 Playback:** Uses Windows MCI commands (mciSendString)
* **Controls:** Play/pause, volume adjustment
* **Suggested Music:** Synthwave/retrowave tracks (e.g., "Retrowave Music.mp3")

**2.3 User Interaction**

| **Key** | **Action** |
| --- | --- |
| **WASD** | Move camera forward/backward/strafe |
| **QE** | Move camera up/down |
| **Arrow Keys** | Rotate camera view |
| **P** | Toggle music playback |
| **+/-** | Adjust volume |
| **ESC** | Exit program |

**3. Technical Implementation**

**3.1 Code Structure**

| **File/Module** | **Purpose** |
| --- | --- |
| **main()** | Initializes GLUT, sets up callbacks |
| **init()** | Configures OpenGL settings, generates objects |
| **display()** | Main rendering loop |
| **timer()** | Handles animations (60 FPS) |
| **drawBuilding()** | Renders procedurally generated skyscrapers |
| **drawGrid()** | Creates the neon ground grid |
| **drawSpinners()** | Manages rotating vortex/spiral effects |
| **drawTunnel()** | Sky tunnel with depth scrolling |
| **drawCar()** | Low-poly cars with light trails |
| **drawPyramid()** | Animated pyramid with neon glow |
| **drawTorus()** | Uses glutWireTorus + glutSolidTorus |
| **SimpleAudioPlayer** | Handles MP3 playback & volume |

**3.2 Animation Techniques**

* **Time-based transformations** (glutGet(GLUT\_ELAPSED\_TIME))
* **Sine/cosine waves** for smooth pulsing effects
* **Procedural generation** (e.g., buildings, stars)
* **Dynamic line/point sizing** (neon glow effects)

**4. Future Enhancements**

**4.1 Planned Features**

✔ **Day/Night Cycle** (gradual sky color shifts)  
✔ **More Vehicles** (motorcycles, flying cars)  
✔ **Interactive Elements** (clickable objects, UI)  
✔ **Particle Effects** (engine exhaust, rain)

**4.2 Optimization**

✔ **Level of Detail (LOD)** for distant objects  
✔ **Frustum Culling** to skip off-screen rendering  
✔ **Shader Integration** (GLSL for better lighting)

**5. Assets & Dependencies**

| **Resource** | **Usage** |
| --- | --- |
| **OpenGL/GLUT** | Core rendering |
| **Windows.h** | Audio playback |
| **Retrowave Music** | Background soundtrack |

**6. Conclusion**

This project combines **nostalgic 80s aesthetics** with **modern OpenGL rendering** to create an interactive synthwave experience. The focus is on:

* **Visual appeal** (neon colors, smooth animations)
* **Atmosphere** (dynamic lighting, music integration)
* **Performance** (60 FPS target, efficient rendering)