1. **Přepínání scén za běhu (Ano):** každá scéna je také v aplikaci reprezentovaná proměnnou bool. Na začátku v konstruktoru aplikace je nastavena pouze scéna s testovacím trojúhelníkem na true. Scény přepínáme pomocí kláves a změn bool hodnot u dalších scén
   1. Obsah obrázku snímek obrazovky, řada/pruh

      Popis byl vytvořen automaticky**Základní scéna (Ano):** testovací se statickým trojúhelníkem
   2. Obsah obrázku měsíc, vánoční stromeček

      Popis byl vytvořen automaticky**Scéna s lesem (Ano):** 50 stromů, 50 keřů, slunce (sphere) osvětlené Lamber shaderem, země (plain)
   3. Obsah obrázku měsíc, Astronomický objekt, snímek obrazovky, Noční nebe

      Popis byl vytvořen automaticky**Scéna se čtyřmi kuličkami (Ano):** osvětleny Phong shaderem
   4. **Scéna pro demonstraci použití všech shaderů (Konstantní, Lambert, Phong a Blinn) (Ano):** strom (konstantní), sphere (Lambert), sphere (Phong)

Obsah obrázku vánoční stromeček, snímek obrazovky

Popis byl vytvořen automaticky

1. **Světlo (Ano)**
2. **Základní třídy (ShaderProgram, DrowableObject, Camera, Controller) (Ano):** DrawableObject vykresluje objekty, Scene renderuje scény, Application vytváří a scény. Shadery jsou uloženy jako .txt v Resource Files
3. **Transformace (Composite pattern) (Ne) (Máte pro transformace základní třídy? Co a jak jste použili?)**
4. **Základy OOP**

**a) Encapsulation (zapouzdření) (Ano):** ve většíně tříd jsou proměnné privátní a veřejné jsou jen metody

**b) Inheritance (dědičnost) (Ne)**

**c) Polymorphism (polymorfismus neboli mnohotvárnost) (Ne)**

1. **Vertex a fragment shadery prosím uložte do textových souboru a použijte přiložený ShaderLoader pro jejich načítání (Ano):** uloženy v Resource Files, v konstruktoru ShaderProgram jsou vertex a fragment soubory, a ShaderLoader, který načítá soubory