

# Počítačová grafika 2 (PGR2)

Pavel Strachota

FJFI ČVUT v Praze

12. února 2024

# Kontakt

**Ing. Pavel Strachota, Ph.D.**

Katedra matematiky

Trojanova 13, místnost 033a

**E-mail:** [pavel.strachota@fjfi.cvut.cz](mailto:pavel.strachota@fjfi.cvut.cz)

**WWW:** <http://saint-paul.fjfi.cvut.cz/POGR>

**Tel:** (+420) 778 546 112

# Přednášky

- 1 přednáška týdně, **Pondělí 12:00 - 13:40, T-201**
  - formálně rozsah "1+1"
- ukončeno **zápočtem** a **zkouškou**, 2 kredity
  - docházka povinná
- prezentace k přednáškám dostupné na webu

# Obsah přednášky (obecně)

- základní problémy 3D grafiky a jejich řešení:
  - (zjednodušený) matematický popis vybraných metod a algoritmů
  - rozbor implementace (pomocí pseudokódu apod.)
  - přehled dalších metod a náměty na jejich studium

## Obsah přednášky (konkrétně) 1/2

- 1 **Úvod do teorie signálu** - frekvenční oblast, vzorkování a rekonstrukce, aliasing a antialiasing
- 2 **Cíle počítačové 3D grafiky** - realismus vs. účelové zjednodušení
- 3 **Křivky a plochy** - parametrický popis křivek a povrchů, dělené povrchy
- 4 **Počítačová reprezentace objektů** - objemová, CSG, polygonální síť
- 5 **Modelovací techniky** - fraktály, procedurální modelování, gramatiky, ...
- 6 **Geometrické transformace pomocí matic** - homogenní souřadnice, afinní transformace

## Obsah přednášky (konkrétně) 2/2

- 7 **Promítání** - rovnoběžná a perspektivní projekce, specifikace pohledu, systémy souřadnic
- 8 **Řešení viditelnosti** - malířův algoritmus, Warnockův algoritmus, Z-buffer
- 9 **Osvětlování a stínování** - světelné zdroje, osvětlovací modely, stínování polygonů
- 10 **Mapování textur** - projekce, mip-mapping, bump mapping, environment mapping
- 11 **Globální a fyzikálně založené zobrazovací metody** - *raytracing*, distribuovaný raytracing, fotonové mapy, radiozita

## Literatura - POGR 2

-  S. Marschner, P. Shirley: *Fundamentals of Computer Graphics (4th ed.)*, CRC Press, Taylor & Francis Group, 2016.
-  J. F. Hughes, A. van Dam, M. McGuire, D. F. Sklar, J. D. Foley, S. K. Feiner, K. Akeley: *Computer Graphics: Principles and Practice (3rd ed.)*, Addison Wesley, 2014.
-  T. Akenine-Möller, E. Haines, N. Hoffman, A. Pesce, M. Iwanicki, S. Hillaire: *Real-Time Rendering (4th ed.)*, CRC Press, Taylor & Francis Group, 2018.
-  D. Shreiner, G. Sellers, J. Kessenich, B. Licea-Kane: *OpenGL Programming Guide (8th ed.)*, Addison Wesley, Pearson Education, 2013.
-  Žára, Beneš, Sochor, Felkel - *Moderní počítačová grafika*. Computer Press, 2005.

# Hodnocení

## Zápočet:

- 1 za docházku na přednášky (min 50% účast)
- 2 za svědomitě a samostatně vypracovanou zápočtovou práci dle pokynů (**PŘEČTĚTE SI JE na webu!!!**)
  - témata viz dále
  - Práce na zápočtovém úkolu by měla obohatit vás a jeho výsledek by mohl obohatit přednášku příští rok :-)

## Zkouška – ústní pohovor:

- 1 prezentace zápočtové práce a demonstrace porozumění tématu
- 2 diskuse o teorii, obvykle související s tématem práce



# Zápočet

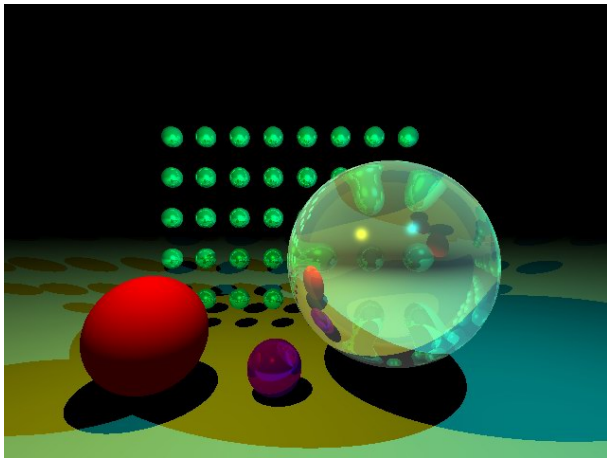
## 1. Program

**Program** implementující některý ze složitějších algoritmů počítačové grafiky:

- softwarový rendering: objekt ze souboru ze standardního formátu
- raytracing + nějaké jeho vylepšení - více druhů objektů, stochastické vzorkování apod.
- program využívající grafické API (OpenGL, DirectX, Vulkan ...)  $\implies$  rychlost (realtime pohyb kamery atd.)
- procedurální modelování (fraktály, gramatiky)
- modelování systémů částic (numerické řešení pohybových rovnic)
- GUI pro modelování polygonálních křivek a ploch (Bézierovy křivky a pláty)

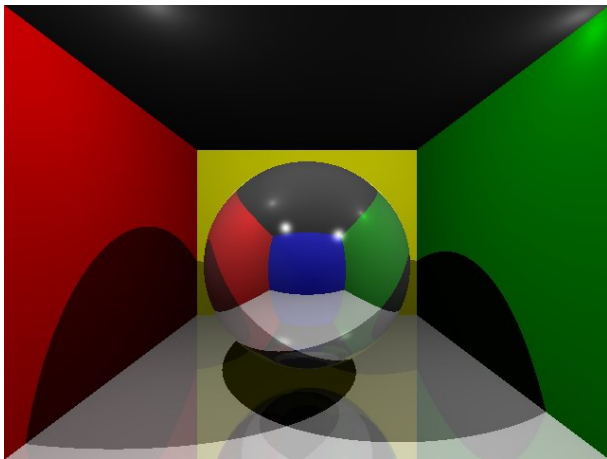
# Zápočet

Raytracing:



# Zápočet

Raytracing:



# Zápočet

## Vlastnosti správně vypracovaného programu

- přehledná struktura kódu, (doporučen) objektový návrh
- program se musí přeložit bez chyb a fungovat
- dokumentace
  - návod, jak přeložit + jaké knihovny jsou třeba
  - jak se používá
- **musíte ho napsat VY !**
  - můžete používat hotový kód (knihovny, úryvky kódu z internetu), ale nikoliv ke splnění samotného úkolu v zadání (**Příklad - raytracing:** využít knihovnu SDL pro vykreslování bodu na „plátno“ je OK, ale použít knihovnu LibTrace (<http://libtrace.sourceforge.net>) implementující algoritmus samotný NENÍ OK)

# Zápočet

## 2. 3D scéna (objekt, animace)

- vytvořená pomocí 3D modelovacího software (*Blender*, *3ds Max*, *Maya*, *SketchUp*, konec konců i *AutoCAD...*)
- PDF soubor s detailním popisem postupu (tutorial)

# Zápočet

## 2. 3D scéna (objekt, animace)

- vytvořená pomocí 3D modelovacího software (*Blender*, *3ds Max*, *Maya*, *SketchUp*, konec konců i *AutoCAD...*)
- PDF soubor s detailním popisem postupu (tutorial)



# Zápočet

## 2. 3D scéna (objekt, animace)

- vytvořená pomocí 3D modelovacího software (*Blender*, *3ds Max*, *Maya*, *SketchUp*, konec konců i *AutoCAD*...)
- PDF soubor s detailním popisem postupu (tutorial)



# Zápočet

## 2. 3D scéna (objekt, animace)

- vytvořená pomocí 3D modelovacího software (*Blender*, *3ds Max*, *Maya*, *SketchUp*, konec konců i *AutoCAD*...)
- PDF soubor s detailním popisem postupu (tutorial)

