

# NURBS-visualizer

## User manual V1.0

Jakub Kubát, Martina Šupolová, Matěj Naar

October 2024

## 1 Úvod

NURBS-visualizer je jednoduchý program vytvořený pro výpočty diskretizace aproximačních NURBS křivek z jejich jednoznačného zadání.

## 2 Skompilování programu

Program je napsaný v jazyce C++, konkrétně ve standardu C++11. Zdrojový kód je dostupný v repozitáři na adrese [https://github.com/KubaVajbik/NURBS-visualizer\\_PIA-2024](https://github.com/KubaVajbik/NURBS-visualizer_PIA-2024). Všechny soubory v repozitáři uložíme lokálně na vlastním počítači do jedné složky. V té samé složce vytvoříme složku „src“.

Ke skompilování zdrojového kódu je potřeba knihovna Simple DirectMedia Layer, dostupná na <https://www.libsdl.org/>.

Pro její stažení:
















1. Navštívíme výše zmíněný odkaz <https://www.libsdl.org/>
2. Nalezneme část „SDL Releases“
3. Stáhneme soubor „SDL2-devel-*aktualni verze*-mingw.tar.gz“. Po rozzipování jeho obsah, tedy složku „SDL2-*aktualni verze*“ uložíme do vhodného adresáře na našem zařízení.

Pro zahrnutí knihovny při kompilování zdrojového kódu programu musíme:

1. V právě uložené složce „SDL2-*aktualni verze*“ nalézt složku „x86\_64-w64-mingw32“
2. V ní se nacházejí složky „include“ a „lib“. Obě tyto složky zkopírujeme do dříve vytvořené složky „src“.
3. Dále pak zpět ve složce „x86\_64-w64-mingw32“ rozklikneme složku „bin“. Soubor „SDL2.dll“ zkopírujeme do složky se soubory z repozitáře.

Po těchto operacích by vše mělo být připraveno na skompilování programu NURBS-visualizer (pro kontrolu porovnejte s obrázkem 1). Tedy za předpokladu, že na vašem systému je nainstalovaný překladač C++. Pokud ne, lze toho jednoduše dosáhnout (pro windows) postupováním podle návodu

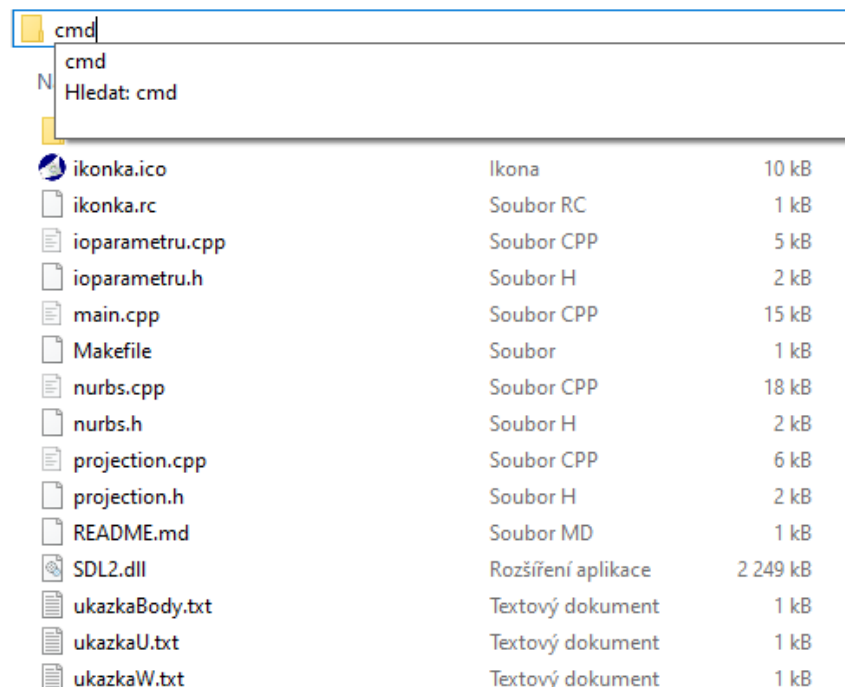
[https://code.visualstudio.com/docs/cpp/config-mingw#\\_prerequisites](https://code.visualstudio.com/docs/cpp/config-mingw#_prerequisites).

src <b>obsahuje dvě složky</b>			Složka souborů
	ikonka.ico	Ikona	10 kB
	ikonka.rc	Soubor RC	1 kB
	ioparametru.cpp	Soubor CPP	5 kB
	ioparametru.h	Soubor H	2 kB
	main.cpp	Soubor CPP	15 kB
	Makefile	Soubor	1 kB
	nurbs.cpp	Soubor CPP	18 kB
	nurbs.h	Soubor H	2 kB
	projection.cpp	Soubor CPP	6 kB
	projection.h	Soubor H	2 kB
	README.md	Soubor MD	1 kB
	SDL2.dll	Rozšíření aplikace	2 249 kB
	ukazkaBody.txt	Textový dokument	1 kB
	ukazkaU.txt	Textový dokument	1 kB
	ukazkaW.txt	Textový dokument	1 kB

Obrázek 1: Obsah složky se soubory z repozitáře po všech výše popsáných operacích

Samotné skompilování programu lze provést následovně:

1. Nejprve ve složce se zdrojovými soubory otevřeme příkazový řádek. Nejjednodušší způsob je zobrazen na následujícím obrázku. Do lišty s adresou složky stačí napsat „cmd“.



2. V příkazové řádce pak stačí použít příkaz „make“ nebo jeho ekvivalent. Pokud jste následovali návod pro instalaci překladače z tohoto dokumentu, pak je možné použít příkaz „mingw32-make“.

```

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.5011]
(c) Microsoft Corporation. Všechna práva vyhrazena.

C:\Users\Kuba\Downloads\NURBS-visualizer-main>mingw32-make_

```

3. Jestliže vše proběhlo v pořádku, uvidíte ve složce se zdrojovým kódem aplikaci „NURBS-visualizer.exe“

```

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.5011]
(c) Microsoft Corporation. Všechna práva vyhrazena.

C:\Users\Kuba\Downloads\NURBS-visualizer-main>mingw32-make
windres -i ikonka.rc -o ikonka.o
g++ -Wall -g -c nurbs.cpp
g++ -Wall -g -c projection.cpp
g++ -Wall -g -c ioparametru.cpp
g++ -I src/include -L src/lib -o NURBS-visualizer main.cpp nurbs.o proj
ection.o ioparametru.o ikonka.o -lmingw32 -lSDL2main -lSDL2

C:\Users\Kuba\Downloads\NURBS-visualizer-main>

```

src	Složka souborů	
ikonka.ico	Ikona	10 kB
ikonka.o	Soubor O	10 kB
ikonka.rc	Soubor RC	1 kB
ioparametru.cpp	Soubor CPP	5 kB
ioparametru.h	Soubor H	2 kB
ioparametru.o	Soubor O	130 kB
main.cpp	Soubor CPP	15 kB
Makefile	Soubor	1 kB
nurbs.cpp	Soubor CPP	18 kB
nurbs.h	Soubor H	2 kB
nurbs.o	Soubor O	567 kB
<b>NURBS-visualizer.exe</b>	Aplikace	743 kB
projection.cpp	Soubor CPP	6 kB
projection.h	Soubor H	2 kB
projection.o	Soubor O	231 kB
README.md	Soubor MD	1 kB
SDL2.dll	Rozšíření aplikace	2 249 kB
ukazkaBody.txt	Textový dokument	1 kB
ukazkaU.txt	Textový dokument	1 kB
ukazkaW.txt	Textový dokument	1 kB

## 3 Jak program pracuje

Program si nejprve vyžádá 3 soubory, které obsahují hodnoty potřebné pro jednoznačné zadání aproximační NURBS křivky. Tj. soubor se souřadnicemi řídicích bodů křivky, soubor s hodnotami intervalů parametru křivky a soubor s vahami jednotlivých řídicích bodů. Pro každý soubor provede kontrolu hodnot v něm obsažených.

Jakmile jsou načteny všechny soubory, program se zeptá na stupeň křivky  $p$ , která je zadaná načtenými soubory. Je to kontrola, jestli jsou v souborech správné hodnoty. Stupeň křivky se dopočítá z rovnice

$$p = m - n - 1 \quad (1)$$

kde  $(m+1)$  je počet uzlů,  $(n+1)$  je počet řídicích bodů a  $p$  je stupeň křivky. Pokud zadáný stupeň křivky  $p$  souhlasí se stupněm vypočítaným programem, tak se program posouvá dál.

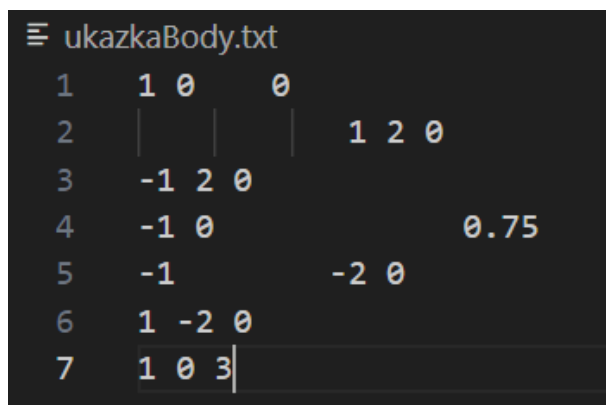
Program se zeptá na počet úseků, kterými chceme křivku aproximovat. po zadání této hodnoty je vygenerován soubor diskretizaceKrivky.txt, ve kterém jsou uloženy souřadnice bodů křivky.

### 3.1 Vstupní soubory

#### 3.1.1 Soubor s řídicími body křivky

Textový soubor.

Na každém jeho řádku se za sebou nachází jednotlivé souřadnice jednoho řídicího bodu. Souřadnice jsou oddělené mezerami nebo jinými prázdnými znaky. Podporované znaky jsou všechny číslice, pak desetinná tečka a symbol minus. Všechny řídicí body musí mít stejný počet souřadnic.



```
ukazkaBody.txt
1 1 0 0
2 | 1 2 0
3 -1 2 0
4 -1 0 0.75
5 -1 -2 0
6 1 -2 0
7 1 0 3|
```

Obrázek 2: Příklad platného vstupního souboru s řídicími body

#### 3.1.2 Soubor s hodnotami intervalů parametru křivky

Textový soubor.

V jednom řádku se za sebou nachází jednotlivé hodnoty parametru křivky. Musí jít o neklesající posloupnost hodnot (ovšem tohle program nekontroluje!!). Souřadnice jsou oddělené mezerami nebo jinými prázdnými znaky. Podporované znaky jsou všechny číslice, pak desetinná tečka a symbol minus.

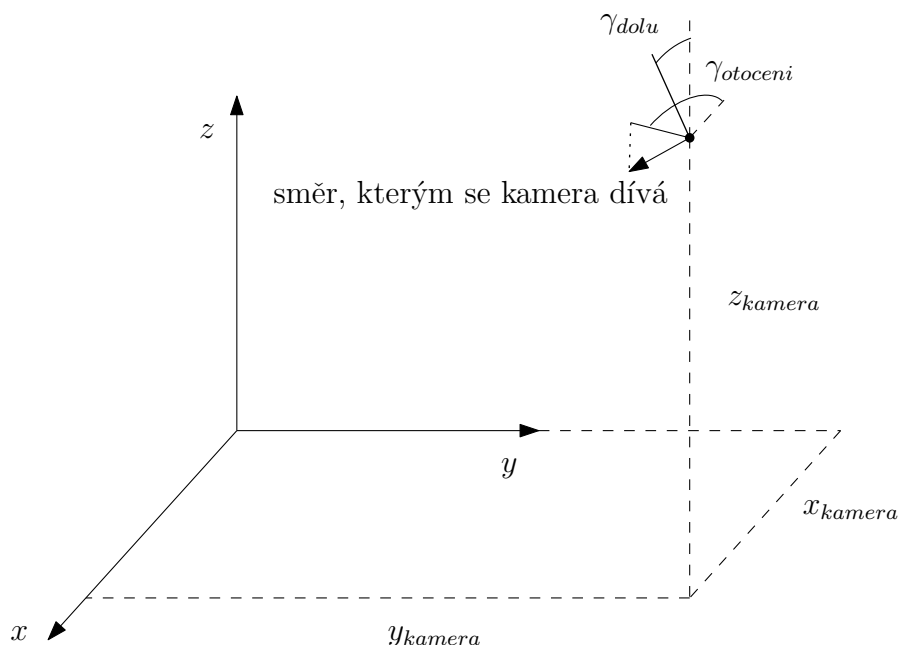
### 3.1.3 Soubor s vahami jednotlivých řídicích bodů

Textový soubor.

V jednom řádku se za sebou nachází jednotlivé hodnoty vah řídicích bodů křivky. Souřadnice jsou oddělené mezerami nebo jinými prázdnými znaky. Podporované znaky jsou všechny číslice, pak desetinná tečka a symbol minus.

## 3.2 Křivka v prostoru $\mathbb{E}^3$

Pokud se zadaná křivka nachází v prostoru  $\mathbb{E}^3$ , tak program ještě vygeneruje její náhled v perspektivě. Pro vykreslení ale potřebuje zadat polohu kamery, která se na křivku dívá. Ze znalosti souřadnic řídicích bodů známe i umístění aproximační křivky, a z uzlového vektoru taky její rozsah. Program pracuje v pravotočivém souřadnicovém systému zobrazeném na následujícím obrázku. Kamera je zadána svými kartézskými souřadnicemi  $(x, y, z)$ , nakloněním „dolů“ a otočením od záporné osy  $x$  okolo osy  $z$  v kladném směru.



## 4 Spuštění programu

Nejjednodušší spuštění programu je pomocí aplikace NURBS-visualizer.exe, sestavené v kapitole 2. Po dvojkliku se spustí příkazový řádek, kdy proběhne zadávání a kontrola vstupních parametrů.

Tipy pro hodnoty ukázkových souborů:

- Stupeň  $p$  křivky = 3
- kamera (4, 4, 6), natočení dolů = 45, natočení od záporné osy  $x$  = 45