

Gymnázium, Praha 6, Arabská 14

předmět : Programování

vyučující : Jan Lána

# Kalei-draw-scope



ročníková práce

Lukáš Grosman 4.E

2020

# Gymnázium, Praha 6, Arabská 14

Arabská 14, Praha 6, 160 00

## ROČNÍKOVÁ PRÁCE

**Předmět:** Programování

**Téma:** Malovací editor

**Autor:** Lukáš Grosman

**Třída:** 4.E

**Školní rok:** 2019/2020

**Třídní učitel:** Mgr. Jana Urzová

# Čestné prohlášení

Prohlašuji, že jsem jediným autorem tohoto projektu, všechny citace jsou řádně označené a všechna použitá literatura a další zdroje jsou v práci uvedené. Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská 14 oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu.

V Praze dne: \_\_\_\_\_

\_\_\_\_\_

vlastnoruční podpis autora

# **Anotace**

Cílem mé ročníkové práce je tvorba programu, který slouží jako prostředí pro kreslení buď jednoduchých obrázků s pomocí základních nástrojů, nebo složitějších za použití komplexnějších nástrojů. Práce zahrnuje vzhled uživatelského rozhraní, výpočet náročnějších štětců, funkci programu samotného a popis použitých aplikací.

# **Annotation**

The aim of my class project is to create a program that serves as an environment for drawing either simple pictures with the help of basic tools, or more complex images using more intricate tools. The work includes the appearance of the user interface, the calculation of more elaborate brushes, the function of the program itself and a description of the applications used.

# Zadání práce

Naprogramujte malovací editor. Program by měl být schopen z uživatelského vstupu vytvářet obrázky na tomu určené ploše, nastavení velikosti a barvy štětce, symetrického kreslení podle osové a bodové souměrnosti a měl by umět ukládat výtvary.

# Obsah

<b><u>Úvod</u></b>	<b><u>7</u></b>
<b><u>Tvorba projektu</u></b>	<b><u>8</u></b>
<b>Použitý software</b>	<b>8</b>
NetBeans	8
Java	8
JavaFX	8
FXML	9
Scene builder	9
<b>Soubory projektu</b>	<b>10</b>
KaleiDrawScope.java	10
KDSFXML.fxml	10
KDSFXMLController.java	10
<b>Průběh práce</b>	<b>11</b>
Průzkum	11
Plánování	11
Realizace	12
Vylepšení	12
Dokončení	12
<b>Problémy při práci</b>	<b>13</b>
Nedostatečná znalost JavaFX	13
Přílišné ambice	13
Matematika	14
Hardware	15
<b><u>Zacházení s KaleiDrawScopem</u></b>	<b><u>16</u></b>
Spuštění	16
<b>Ovládání aplikace</b>	<b>17</b>
Ukládání obrázku	17
Velikost štětce	17
Barva	17
Guma	18
Náhodná duha	18
Symetrie	18
Ukončit program	18
<b><u>Galerie</u></b>	<b><u>19</u></b>
<b><u>Závěr</u></b>	<b><u>20</u></b>
<b><u>Zdroje</u></b>	<b><u>21</u></b>
Zdroje obrázků	21



# Úvod

V mojí ročníkové práci se zaměřuji na tvorbu malovacího editoru v Javě. V jednotlivých kapitolách práce je popsáno jak jsem tento editor vytvořil, jakým způsobem jsem došel k výpočtům některých složitějších nástrojů, s jakými problémy jsem se při práci potýkal, prostředí a zdroje, které jsem použil, a způsob fungování editoru samotného. Také se zde dozvíte jak tento editor spustit a ovládat na svém zařízení.

Toto téma jsem si zvolil, jelikož chci po střední škole pokračovat ve studiu počítačové grafiky, umění a multimédií, a tak mě zajímalo, co obnáší vývoj aplikací pro digitální umění. Od realizace projektu očekávám zlepšení svých programovacích schopností a hlubší pochopení nástrojů, se kterými běžně na počítači pracuji.



# Tvorba projektu

## Použitý software

### *NetBeans*

Celý malovací editor je vytvořen v NetBeans IDE 8.2. Toto vývojové prostředí jsem si zvolil převážně z důvodu, že ho používáme běžně na hodinách, a tak jsem se rozhodl upřednostnit již nabrané zkušenosti před výhodami jiných prostředí. “NetBeans IDE je svobodné, zdarma distribuované integrované vývojové prostředí (IDE)”<sup>1</sup> NetBeans jsou napsané v programovacím jazyce Java, se kterým se v nich také nejčastěji pracuje.

### *Java*

“Java je objektově orientovaný programovací jazyk<sup>2</sup>, který vyvinula firma Sun Microsystems a představila 23. května 1995. Jde o jeden z nejpoužívanějších programovacích jazyků na světě. Podle TIOBE indexu je Java nejpopulárnější programovací jazyk.”<sup>3</sup>

### *JavaFX*

Pro tvorbu svého malovacího editoru jsem si vybral platformu JavaFX. Důvodem mého výběru je její jednoduchost a rychlost při vývoji aplikací pro počítače. JavaFX se zakládá na Javě a slouží převážně k tvorbě a funkčnosti uživatelského rozhraní.<sup>4</sup>

---

<sup>1</sup> "NetBeans – Wikipedie." <https://cs.wikipedia.org/wiki/NetBeans>. Datum přístupu: 30 dub. 2020.

<sup>2</sup> "Programovací jazyk – Wikipedie." [https://cs.wikipedia.org/wiki/Programovac%C3%AD\\_jazyk](https://cs.wikipedia.org/wiki/Programovac%C3%AD_jazyk). Datum přístupu: 30 dub. 2020.

<sup>3</sup> "Java (programovací jazyk) – Wikipedie." [https://cs.wikipedia.org/wiki/Java\\_\(programovac%C3%AD\\_jazyk\)](https://cs.wikipedia.org/wiki/Java_(programovac%C3%AD_jazyk)). Datum přístupu: 30 dub. 2020.

<sup>4</sup> "JavaFX – Wikipedie." <https://cs.wikipedia.org/wiki/JavaFX>. Datum přístupu: 30 dub. 2020.

## *FXML*

Jedná se o značovací jazyk zakládající se na XML - eXtensible Markup Language, který se používá pro definování uživatelského rozhraní JavaFX aplikací. Soubor FXML se nazývá FXML dokument, ze kterého se přímo v JavaFX pomocí třídy FXMLLoader<sup>5</sup> načte hierarchie objektů.<sup>6</sup>

## *Scene builder*

Dokud nebyl soubor KDSfxml.fxml příliš komplexní, používal jsem Scene Builder z JavaFX. To je nástroj pro vytváření uživatelského rozhraní v JavaFX. Jeho výhodou je méně kódu, který musí programátor psát. Funguje na principu drag and drop, což znamená, že se FXML dokument generuje sám z takové “programátorské skládačky”. Vyberu komponent, přetáhnu na pracovní plochu, specifikuji parametry a hotovo. Má ale zásadní nevýhodu. Čím je uživatelské rozhraní komplikovanější, tím pomaleji a náročněji funguje.<sup>7</sup>

---

<sup>5</sup> "FXMLLoader (JavaFX 8) - Oracle Docs."

<https://docs.oracle.com/javase/8/javafx/api/javafx/fxml/FXMLLoader.html>. Datum přístupu: 30 dub. 2020.

<sup>6</sup> "FXML - Wikipedia." <https://en.wikipedia.org/wiki/FXML>. Datum přístupu: 30 dub. 2020.

<sup>7</sup> "JavaFX Scene Builder – Wikipedie." [https://cs.wikipedia.org/wiki/JavaFX\\_Scene\\_Builder](https://cs.wikipedia.org/wiki/JavaFX_Scene_Builder). Datum přístupu: 30 dub. 2020.

# Soubory projektu

## *KaleiDrawScope.java*

Hlavní třída celé aplikace. Obsahuje metody start a main. Metoda start načítá data z KDSfxml.fxml dokumentu ze kterých dále tvoří vzhled uživatelského rozhraní. Metoda main<sup>8</sup> je nejdůležitější metodou v Javě, jelikož spouští celý program.

## *KDSfxml.fxml*

Jedná se o již zmíněný FXML dokument. Obsahuje data o umístění a vlastnostech komponentů uživatelského rozhraní, které z něj čte metoda FXMLLoader.<sup>9</sup>

## *KDSfxmlController.java*

Tak jako název v angličtině napovídá, jde o ovladač. Obsahuje akce a podmínky, jako například “při stisknutí tlačítka A udělej akci A”. Tyto akce následně přiřazuje komponentům z FXML dokumentu. Dále obsahuje metody initialize, onSave a onExit. Initialize spouští funkce FXML komponentů, onSave zajišťuje uložení obrázku při vybrání možnosti “Save” a onExit ukončuje běh programu při vybrání možnosti “Exit”.

---

<sup>8</sup> "public static void main(String[] args) - Java main method ...."

<https://www.journaldev.com/12552/public-static-void-main-string-args-java-main-method>. Datum přístupu: 30 dub. 2020.

<sup>9</sup> "FXMLLoader (JavaFX 8) - Oracle Docs."

<https://docs.oracle.com/javase/8/javafx/api/javafx/fxml/FXMLLoader.html>. Datum přístupu: 30 dub. 2020.

# Průběh práce

## *Průzkum*

Předtím než jsem se pustil do samotného psaní kódu svého programu, vydal jsem se na internet, abych zjistil, co vše vlastně potřebuji znát k vytvoření takového malovacího editoru. Z několika tutoriálů jsem zjistil, jaké komponenty FXML jsou ideální pro mé účely. Nejdůležitější bylo naučit se s canvasem<sup>10</sup> - plátnem. Ostatní komponenty už byly snazší, jelikož je dobře znám z hodin programování. Jde o všemožné checkboxy<sup>11</sup>, labels<sup>12</sup>, slidery<sup>13</sup> a tlačítka<sup>14</sup>. Nejvíce inspirace jsem našel u tvůrce Almas Baimagambetova<sup>15</sup>. Má na svém GitHubu<sup>16</sup> mnoho materiálů pro snadné pochopení funkcí JavaFX.

## *Plánování*

Proces plánování byl poměrně krátký, jelikož rád pracuji stylem okamžitého zpracování myšlenek. Samozřejmě jsem si trochu promyslel, jak přibližně chci aby aplikace vypadala, ale i tak jsem pracoval spíše neorganizovaně. Z tohoto stylu vzešla zásadní

---

<sup>10</sup> "Canvas (JavaFX 8) - Oracle Docs."

<https://docs.oracle.com/javase/8/javafx/api/javafx/scene/canvas/Canvas.html>. Datum přístupu: 30 dub. 2020.

<sup>11</sup> "CheckBox (JavaFX 8) - Oracle Docs."

<https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/CheckBox.html>. Datum přístupu: 30 dub. 2020.

<sup>12</sup> "Label (JavaFX 8) - Oracle Docs."

<https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/Label.html>. Datum přístupu: 30 dub. 2020.

<sup>13</sup> "Slider (JavaFX 8) - Oracle Docs."

<https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/Slider.html>. Datum přístupu: 30 dub. 2020.

<sup>14</sup> "Button (JavaFX 8) - Oracle Docs."

<https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/Button.html>. Datum přístupu: 30 dub. 2020.

<sup>15</sup> "AlmasB (Almas Baimagambetov) · GitHub." <https://github.com/AlmasB>. Datum přístupu: 30 dub. 2020.

<sup>16</sup> "GitHub – Wikipedie." <https://cs.wikipedia.org/wiki/GitHub>. Datum přístupu: 30 dub. 2020.

nevýhoda na které by jistě chtělo zapracovat. A to nepřehlednost kódu. Jak přibývaly metody, funkce a výpočty, kód byl čím dál tím více labyrintem<sup>17</sup> nežli uspořádaným kódem.

## *Realizace*

Realizace je poměrně jednoduchý proces. Vše co jsem si připravil a našel převést do podoby fungujícího programu. Jednoduchý proces v případě, že vše funguje jak má. Což ze začátku platilo, ale jak je popsáno v kapitole “Problémy při práci” každá další úprava byla složitější a složitější. Nakonec ale pocit zadostiučinění, když nakonec vše nějak funguje, stojí za všechno to trápení.

## *Vylepšení*

Když vše funguje jak má, už zbývá jen doladit nesrovnalosti. Stejně tak zde, když už program vesele kreslil na plátno jak má, práce pokračovala laděním velikostí komponent, textů a několika menších problémů v kódu.

## *Dokončení*

Spokojenost s výsledkem na řádcích NetBeans znamená jediné. Build<sup>18</sup> - převedení celého projektu do souboru .jar<sup>19</sup> pro snadné spuštění i bez NetBeans. Po splnění tohoto úkonu lze považovat programovací část za dokončenou.

---

<sup>17</sup> "Labyrint – Wikipedie." <https://cs.wikipedia.org/wiki/Labyrint>. Datum přístupu: 30 dub. 2020.

<sup>18</sup> "Build Tools - NetBeans IDE." <https://netbeans.org/features/java/build-tools.html>. Datum přístupu: 30 dub. 2020.

<sup>19</sup> "JAR (souborový formát) – Wikipedie." [https://cs.wikipedia.org/wiki/JAR\\_\(souborov%C3%BD\\_form%C3%A1t\)](https://cs.wikipedia.org/wiki/JAR_(souborov%C3%BD_form%C3%A1t)). Datum přístupu: 30 dub. 2020.

# Problémy při práci

## *Nedostatečná znalost JavaFX*

I přes několik hodin ve škole zaměřených právě na JavaFX jsem musel poměrně hodně hledat odpovědi na internetu. Převážně na stránkách StackOverflow<sup>20</sup> a Oracle<sup>21</sup>. Neustálé chyby v kódu, neporozumění NetBeansu, že nemůžu kombinovat některé formáty a typy dohromady a bezradné čtení errorů<sup>22</sup> z outputu<sup>23</sup> byly všudypřítomné.

## *Přílišné ambice*

Dalším problémem při tvorbě byly rozhodně mé představy o tom, co vše by ještě šlo přidat. Naprosté selhání pokusů o “nepodstatné” vylepšení přišlo po několikahodinovém snažení pro zprovoznění FileChooseru<sup>24</sup>. Ten měl být nápomocen při načítání uložených obrázků pro úpravy, ale zkrátka jsem nepřišel na to, jak ho napojit. Úplně odněkud zkopírovat řešení by bylo zbytečné, protože by pak celé snažení ztratilo smysl.

---

<sup>20</sup> "Stack Overflow." <https://stackoverflow.com/>. Datum přístupu: 30 dub. 2020.

<sup>21</sup> "Oracle." <https://www.oracle.com/index.html>. Datum přístupu: 30 dub. 2020.

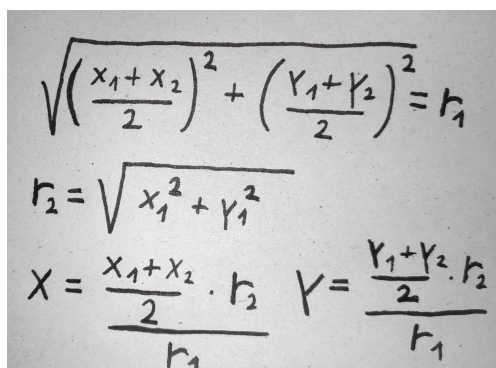
<sup>22</sup> "error - ABZ.cz: slovník cizích slov." <https://slovník-cizich-slov.abz.cz/web.php/slovo/error>. Datum přístupu: 30 dub. 2020.

<sup>23</sup> "output - ABZ.cz: slovník cizích slov." <https://slovník-cizich-slov.abz.cz/web.php/slovo/output-autput>. Datum přístupu: 30 dub. 2020.

<sup>24</sup> "FileChooser (JavaFX 8) - Oracle Docs." <https://docs.oracle.com/javase/8/javafx/api/javafx/stage/FileChooser.html>. Datum přístupu: 30 dub. 2020.

## Matematika

Když se řekne slovo symetrie<sup>25</sup>, většině lidí se vybaví právě matematika. A jelikož symetrické kreslení bylo jedním z hlavních úkolů v mém projektu, matematiky bylo mnoho. Pro dvojnásobnou a čtyřnásobnou symetrii nebyl problém, protože jde pouze o změnu pozic souřadnic  $x$  a  $y$ . Zrcadlová symetrie taky nebyla problém. Opět v ní jde pouze o záměnu souřadnic. Problém však přišel u osminásobné symetrie. Tím, že ve hře nejsou už jen osy  $x$  a  $y$ , složitost výpočtů na lidský mozek působí mnohem silněji. Pokoušel jsem se problém vyřešit pomocí vzorců kružnic<sup>26</sup>, dělení, dosazování, sínů a geometrie. Po zdlouhavém počítání na papíře jsem konečně došel ke vzorci, který fungoval jak měl. A tak byla osminásobná spirála na světě. Ruku v ruce s ní také osminásobné zrcadlo. Pro zajímavost, vzorec ke kterému jsem došel je :


$$\sqrt{\left(\frac{x_1+x_2}{2}\right)^2 + \left(\frac{y_1+y_2}{2}\right)^2} = r_1$$
$$r_2 = \sqrt{x_1^2 + y_1^2}$$
$$X = \frac{\frac{x_1+x_2}{2} \cdot r_2}{r_1} \quad Y = \frac{\frac{y_1+y_2}{2} \cdot r_2}{r_1}$$

Po menším heureka<sup>27</sup> efektu jsem tento vzorec použil ve svém kódu a nevěřícně jsem hleděl, že funguje jak má. Šestnáctinásobná symetrie už byla jen variace tohoto vzorce a tak jsem snadno dosáhl i té.

*(vysvětlení principu výpočtu preferuji osobně, bylo by to na dlouho, a tato ročníková práce je z programování a ne z matematiky)*

<sup>25</sup> "Symetrie – Wikipedie." <https://cs.wikipedia.org/wiki/Symetrie>. Datum přístupu: 30 dub. 2020.

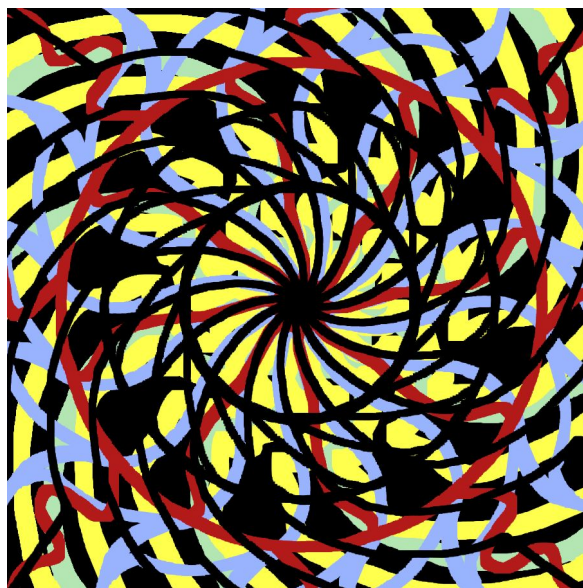
<sup>26</sup> "Analytická geometrie - Kružnice - Matematika pro každého." 14 čvn. 2008, <https://maths.cz/clanky/132-analyticka-geometrie-kruznice>. Datum přístupu: 30 dub. 2020.

<sup>27</sup> "Heuréka – Wikipedie." <https://cs.wikipedia.org/wiki/Heur%C3%A9ka>. Datum přístupu: 30 dub. 2020.



## Hardware

Problémy s hardwarem<sup>28</sup> se nevyskytovaly tak často jako ostatní potíže, ale byly neméně nepříjemné. 28 záložek v Google Chrome<sup>29</sup> a tři běžící instance<sup>30</sup> mého programu můj počítač moc neschvaloval.

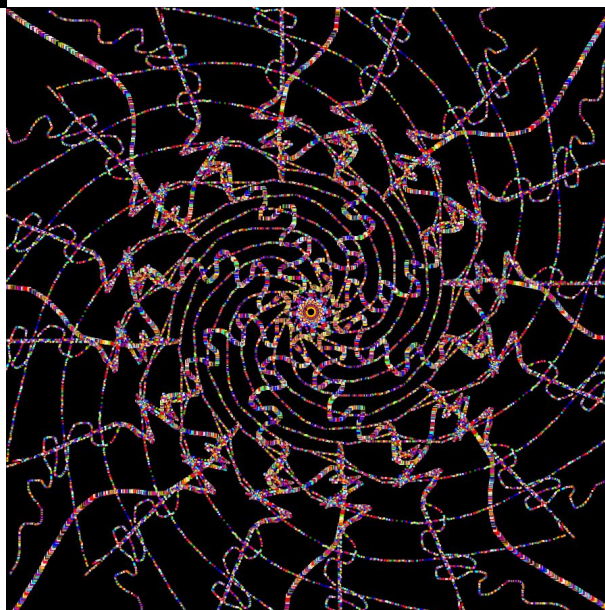


### Příklad č.1

Tento obrazec vznikl kombinací osmi a šestnáctinásobné symetrie, různých velikostí štětce a výběrem z předpřipravených barev.

### Příklad č.2

Tento obrazec vznikl stejnou kombinací symetrií jako minulý, ale se zvolenou možností “rainbow” - duha, která při každém obnovení mění náhodně barvu.



<sup>28</sup> "Hardware – Wikipedie." <https://cs.wikipedia.org/wiki/Hardware>. Datum přístupu: 30 dub. 2020.

<sup>29</sup> "Google Chrome." <https://www.google.com/intl/cs/chrome/>. Datum přístupu: 30 dub. 2020.

<sup>30</sup> "Instance – Wikipedie." <https://cs.wikipedia.org/wiki/Instance>. Datum přístupu: 30 dub. 2020.



# Zacházení s KaleiDrawScopem

## Spuštění

Zapnout KaleiDrawScope je poněkud složité. Nemá žádný instalační .exe<sup>31</sup> soubor, ale lze ho bez vývojového prostředí spustit pomocí souboru .jar<sup>32</sup>. Nejprve musíme rozbalit projekt pomocí dekompresního<sup>33</sup> programu (např. WinRAR<sup>34</sup>). Další krok je cesta k samotnému .jar souboru. Nejdříve otevřeme složku, kterou nám připravil WinRAR - "KaleiDrawScope". Zde je připraven .jar soubor "KaleiDrawScope". Na ten dvakrát klikneme. Pokud se nic nestane, druhá možnost je pokračovat do složky "KaleiDrawScope", z té pokračujeme do složky "dist" a zde je další .jar soubor "KaleiDrawScope". Pokud po dvojím kliknutí na něj opět nenastane spuštění programu, pak je ještě jedna možnost. Otevřeme Start, zadáme "cmd" a stiskneme klávesu "enter". Otevře se příkazový řádek. Do toho napíšeme "java -jar " a umístění .jar souboru "KaleiDrawScope". To lze zjistit pravým kliknutím na .jar soubor "KaleiDrawScope", vybráním možnosti "Vlastnosti" a zde zkopírováním hodnoty "Umístění:". Pokud ani tento způsob nezabere, doporučuji otevřít pomocí vývojového prostředí podporující Javu.

---

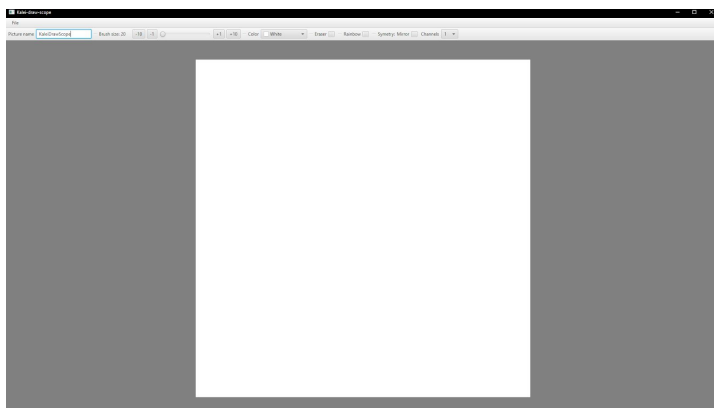
<sup>31</sup> "EXE – Wikipedie." <https://cs.wikipedia.org/wiki/EXE>. Datum přístupu: 30 dub. 2020.

<sup>32</sup> "JAR (souborový formát) – Wikipedie." [https://cs.wikipedia.org/wiki/JAR\\_\(souborov%C3%BD\\_form%C3%A1t\)](https://cs.wikipedia.org/wiki/JAR_(souborov%C3%BD_form%C3%A1t)). Datum přístupu: 30 dub. 2020.

<sup>33</sup> "Dekomprese – Wikipedie." <https://cs.wikipedia.org/wiki/Dekomprese>. Datum přístupu: 30 dub. 2020.

<sup>34</sup> "WinRAR – Wikipedie." <https://cs.wikipedia.org/wiki/WinRAR>. Datum přístupu: 30 dub. 2020.

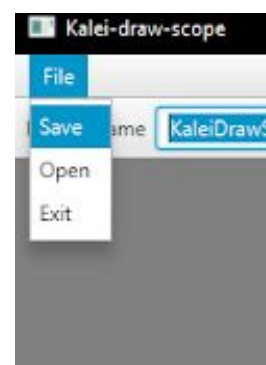
# Ovládání aplikace



Ovládání aplikace je poměrně intuitivní. Stačí základní znalost angličtiny, popřípadě překladáč nebo slovník.

## *Ukládání obrázku*

Textové pole úplně nalevo s popiskem Picture name je název obrázku, pod kterým se bude obrázek ukládat. Pro uložení obrázku stiskneme nahoře vlevo tlačítko “File” a zvolíme možnost “Save”. Obrázky se ukládají do složky “KaleiDrawScope”.



## *Velikost štětce*



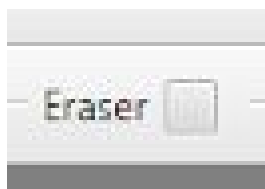
Druhý zleva je “Brush size”. Tento posuvník mění velikost štětce. Tu lze měnit také tlačítka na jeho stranách.

## *Barva*

Barva je v základu bílá. Na bílém pozadí logicky nebude vidět. Pomocí výběru vpravo od popisku “Color” můžeme zvolit barvu z výběru, nebo si “namíchat” svou vlastní. A můžeme začít kreslit.



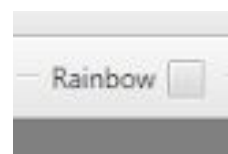
## *Guma*



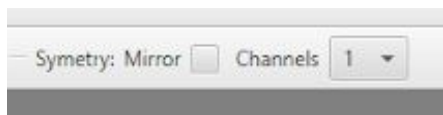
Nástroj guma se zapíná pomocí tlačítka vpravo od popisku “Eraser”. Guma ignoruje zvolenou barvu a smaže všechno přes co ji přetáhnete. Včetně pozadí plátna. Pokud je pozadí plátna někde prázdné, program si tam domyslí bílou barvu. Tak při uložení obrázku na prázdných místech vzniká bílé pozadí.

## *Náhodná duha*

Tlačítko vpravo od popisku “Rainbow” aktivuje “náhodnou duhu”. Ta ignoruje zvolenou barvu a na každém obnovení (seběmenší pohyb štětcem po plátně) změní barvu štětce.



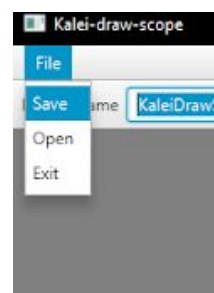
## *Symetrie*



Nastavení symetrie má dva módy. Zrcadlení a spirála. Zrcadlení se aktivuje tlačítkem vpravo od popisku “Mirror”. Spirála je přednastavený režim. Výběr napravo od popisku “Channels” určuje kolik odrazů bude program kreslit. To znamená, že při hodnotě “4” budeme jedním tahem kreslit 4 čáry.

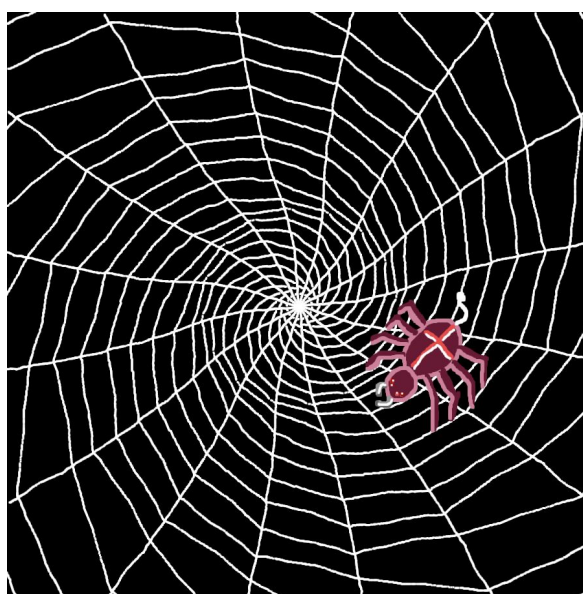
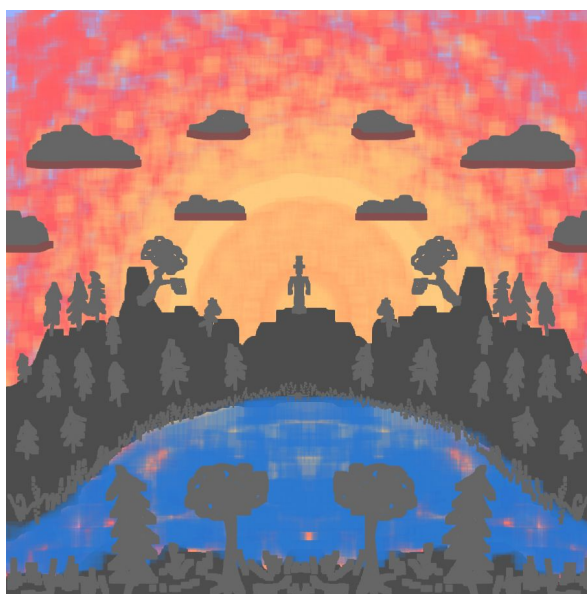
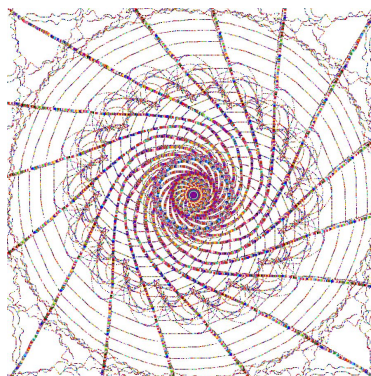
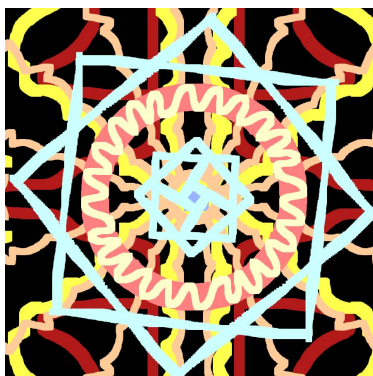
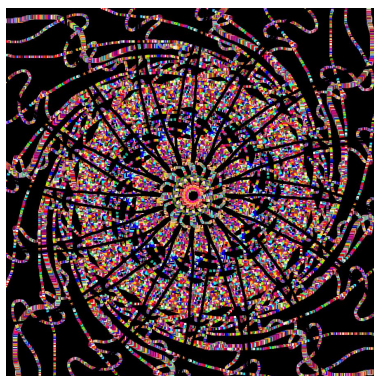
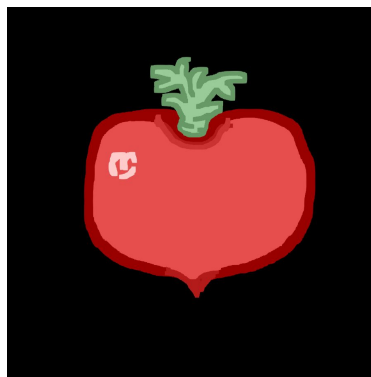
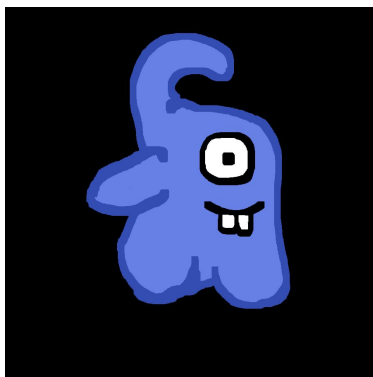
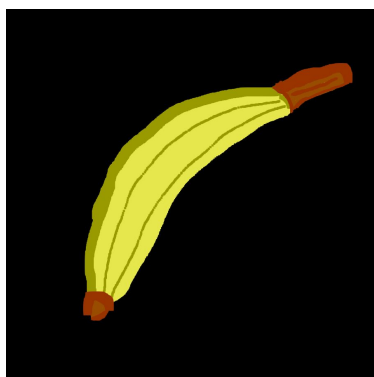
## *Ukončit program*

Nakonec stejným způsobem jako uložení - ukončení programu. To lze buď klasicky křížkem vpravo nahoře, nebo přes tlačítko vlevo nahoře s popiskem “File” a zde tlačítkem “Exit” zavřeme okno s programem a ukončíme ho.



# Galerie

Několik ukázek výtvorů z KaleiDrawScopu.



# Závěr

Ve své ročníkové práci jsem dosáhl svého hlavního cíle. Vytvořit aspoň do jisté míry fungující malovací aplikaci. Ač jsem byl ze začátku skeptický, několik hodin strávených kreslením v mé ročníkové práci mě přesvědčilo o tom, že výsledek předčil má očekávání. I přes problémy s nefunkčními metodami a pro mě šílenými matematickými vzorci jsem zvládl vytvořit to, co jsem chtěl.

Samozřejmě KaleiDrawScope není perfektní. Má své chyby a nedostatky. Například postrádá načítání obrázků pro upravování uložené práce. Co mi chybí asi nejvíc je undo/redo. Pokud se vám něco při kresbě nepovede, jen kliknete na “undo” a obrázek se vrátí o jeden krok zpět. Kliknete mockrát na “undo”? “Redo” vám vrátí vrácený krok. Bohužel jsem nebyl schopen přijít na to, jak tuto funkci zprovoznit ve svém projektu.

Největší překážky v tvorbě této ročníkové práce byly: má nedostatečná znalost platformy JavaFX, černá magie v podobě matematiky a dohady s NetBeansy.

Možné rozšíření práce jsou: lepší zobecnění parametrů např. pro jinak velké monitory, nastavitelná velikost kreslící plochy, lepší systém ukládání a načítání obrázků, více tvarů štětců, více nástrojů a hezčí uživatelské rozhraní.

# Zdroje

- 1) "NetBeans – Wikipedie." <https://cs.wikipedia.org/wiki/NetBeans>. Datum přístupu: 30 dub. 2020.
- 2) "Programovací jazyk – Wikipedie." [https://cs.wikipedia.org/wiki/Programovac%C3%AD\\_jazyk](https://cs.wikipedia.org/wiki/Programovac%C3%AD_jazyk). Datum přístupu: 30 dub. 2020.
- 3) "Java (programovací jazyk) – Wikipedie." [https://cs.wikipedia.org/wiki/Java\\_\(programovac%C3%AD\\_jazyk\)](https://cs.wikipedia.org/wiki/Java_(programovac%C3%AD_jazyk)). Datum přístupu: 30 dub. 2020.
- 4) "JavaFX – Wikipedie." <https://cs.wikipedia.org/wiki/JavaFX>. Datum přístupu: 30 dub. 2020.
- 5) "FXMLLoader (JavaFX 8) - Oracle Docs." <https://docs.oracle.com/javase/8/javafx/api/javafx/fxml/FXMLLoader.html>. Datum přístupu: 30 dub. 2020.
- 6) "FXML - Wikipedia." <https://en.wikipedia.org/wiki/FXML>. Datum přístupu: 30 dub. 2020.
- 7) "JavaFX Scene Builder – Wikipedie." [https://cs.wikipedia.org/wiki/JavaFX\\_Scene\\_Builder](https://cs.wikipedia.org/wiki/JavaFX_Scene_Builder). Datum přístupu: 30 dub. 2020.
- 8) "public static void main(String[] args) - Java main method ...."  
<https://www.journaldev.com/12552/public-static-void-main-string-args-java-main-method>. Datum přístupu: 30 dub. 2020.
- 9) "FXMLLoader (JavaFX 8) - Oracle Docs." <https://docs.oracle.com/javase/8/javafx/api/javafx/fxml/FXMLLoader.html>. Datum přístupu: 30 dub. 2020.
- 10) "Canvas (JavaFX 8) - Oracle Docs." <https://docs.oracle.com/javase/8/javafx/api/javafx/scene/canvas/Canvas.html>. Datum přístupu: 30 dub. 2020.
- 11) "CheckBox (JavaFX 8) - Oracle Docs." <https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/CheckBox.html>. Datum přístupu: 30 dub. 2020.
- 12) "Label (JavaFX 8) - Oracle Docs." <https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/Label.html>. Datum přístupu: 30 dub. 2020.
- 13) "Slider (JavaFX 8) - Oracle Docs." <https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/Slider.html>. Datum přístupu: 30 dub. 2020.
- 14) "Button (JavaFX 8) - Oracle Docs." <https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/Button.html>. Datum přístupu: 30 dub. 2020.
- 15) "AlmasB (Almas Baimagambetov) · GitHub." <https://github.com/AlmasB>. Datum přístupu: 30 dub. 2020.
- 16) "GitHub – Wikipedie." <https://cs.wikipedia.org/wiki/GitHub>. Datum přístupu: 30 dub. 2020.
- 17) "Labyrint – Wikipedie." <https://cs.wikipedia.org/wiki/Labyrint>. Datum přístupu: 30 dub. 2020.
- 18) "Build Tools - NetBeans IDE." <https://netbeans.org/features/java/build-tools.html>. Datum přístupu: 30 dub. 2020.
- 19) "JAR (souborový formát) – Wikipedie." [https://cs.wikipedia.org/wiki/JAR\\_\(souborov%C3%BD\\_form%C3%A1t\)](https://cs.wikipedia.org/wiki/JAR_(souborov%C3%BD_form%C3%A1t)). Datum přístupu: 30 dub. 2020.
- 20) "Stack Overflow." <https://stackoverflow.com/>. Datum přístupu: 30 dub. 2020.
- 21) "Oracle." <https://www.oracle.com/index.html>. Datum přístupu: 30 dub. 2020.
- 22) "error - ABZ.cz: slovník cizích slov." <https://slovník-cizich-slov.abz.cz/web.php/slovo/error>. Datum přístupu: 30 dub. 2020.
- 23) "output - ABZ.cz: slovník cizích slov." <https://slovník-cizich-slov.abz.cz/web.php/slovo/output-autput>. Datum přístupu: 30 dub. 2020.
- 24) "FileChooser (JavaFX 8) - Oracle Docs." <https://docs.oracle.com/javase/8/javafx/api/javafx/stage/FileChooser.html>. Datum přístupu: 30 dub. 2020.
- 25) "Symetrie – Wikipedie." <https://cs.wikipedia.org/wiki/Symetrie>. Datum přístupu: 30 dub. 2020.
- 26) "Analytická geometrie - Kružnice - Matematika pro každého." 14 čvn. 2008,  
<https://maths.cz/clanky/132-analyticka-geometrie-kruznice>. Datum přístupu: 30 dub. 2020.
- 27) "Heuréka – Wikipedie." <https://cs.wikipedia.org/wiki/Heur%C3%A9ka>. Datum přístupu: 30 dub. 2020.
- 28) "Hardware – Wikipedie." <https://cs.wikipedia.org/wiki/Hardware>. Datum přístupu: 30 dub. 2020.
- 29) "Google Chrome." <https://www.google.com/intl/cs/chrome/>. Datum přístupu: 30 dub. 2020.
- 30) "Instance – Wikipedie." <https://cs.wikipedia.org/wiki/Instance>. Datum přístupu: 30 dub. 2020.
- 31) "EXE – Wikipedie." <https://cs.wikipedia.org/wiki/EXE>. Datum přístupu: 30 dub. 2020.
- 32) "JAR (souborový formát) – Wikipedie." [https://cs.wikipedia.org/wiki/JAR\\_\(souborov%C3%BD\\_form%C3%A1t\)](https://cs.wikipedia.org/wiki/JAR_(souborov%C3%BD_form%C3%A1t)). Datum přístupu: 30 dub. 2020.
- 33) "Dekomprese – Wikipedie." <https://cs.wikipedia.org/wiki/Dekomprese>. Datum přístupu: 30 dub. 2020.
- 34) "WinRAR – Wikipedie." <https://cs.wikipedia.org/wiki/WinRAR>. Datum přístupu: 30 dub. 2020.

## Zdroje obrázků

Všechny použité obrázky v této dokumentaci jsou mé výtvoř. Jedná se o obrázky a snímky obrazovky z programu KaleiDrawScope.