

# **Derive 6.1**

## **ŘEŠENÍ VYBRANÝCH ÚLOH Z MATEMATIKY**

**Roman Hašek**

**Jihočeská Univerzita v Č. Budějovicích**

**2004**

# OBSAH

<b>ÚVOD</b>	<b>3</b>
<b>I. OBSLUHA PROGRAMU</b>	<b>4</b>
<b>II. ŘEŠENÍ ÚLOH</b>	<b>8</b>
1. Číselné soustavy	8
2. Funkce teorie čísel	10
2.1 Rozklad na prvočinitele	10
2.2 Největší společný dělitel	11
3. Úprava po krocích	12
4. Úpravy výrazů	13
4.1 Algebraické výrazy	13
4.2 Goniometrické výrazy	15
4.3 Logaritmické výrazy	17
5. Numerické výpočty. Konstanty.	19
6. Rovnice a soustavy rovnic	20
6.1 Grafické řešení rovnic. Tvorba grafu	21
6.2 Algebraické rovnice	22
6.3 Úpravy rovnic	22
6.4 Soustavy rovnic	24
6.5 Goniometrické rovnice	25
7. Nerovnice a soustavy nerovnic	25
8. Relace	27
9. Funkce jedné a více proměnných	29
9.1 Graf funkce	29
9.2 Graf funkce s parametrem. Posuvník	31
9.3 Graf goniometrické funkce	33
9.4 Celá část	34
9.5 Racionální lomená funkce	35
10. Kombinatorika	36
11. Analytická geometrie	37
11.1 Operace s vektory	37
11.2 Vzdálenost mimoběžek	38

11.3 Kreslení geometrických útvarů	39
11.4 Polární souřadnice	40
12. Množinové operace	44
13. 3D grafy	45
13.1 Rotační plochy	46
14. Proložení daných bodů grafem funkce	47
15. Fraktály	50

# ÚVOD

Tento materiál představuje nové vydání příručky **Řešení vybraných úloh z matematiky, Derive 5**, doplněné o další příklady a přepracované na použití s českou lokalizací programu Derive v nejnovější verzi 6.1.

**Derive 6.1** přináší oproti verzi 5 (při kompatibilitě směrem vzhůru) některé nové prvky. Uvedme si ty nejzajímavější:

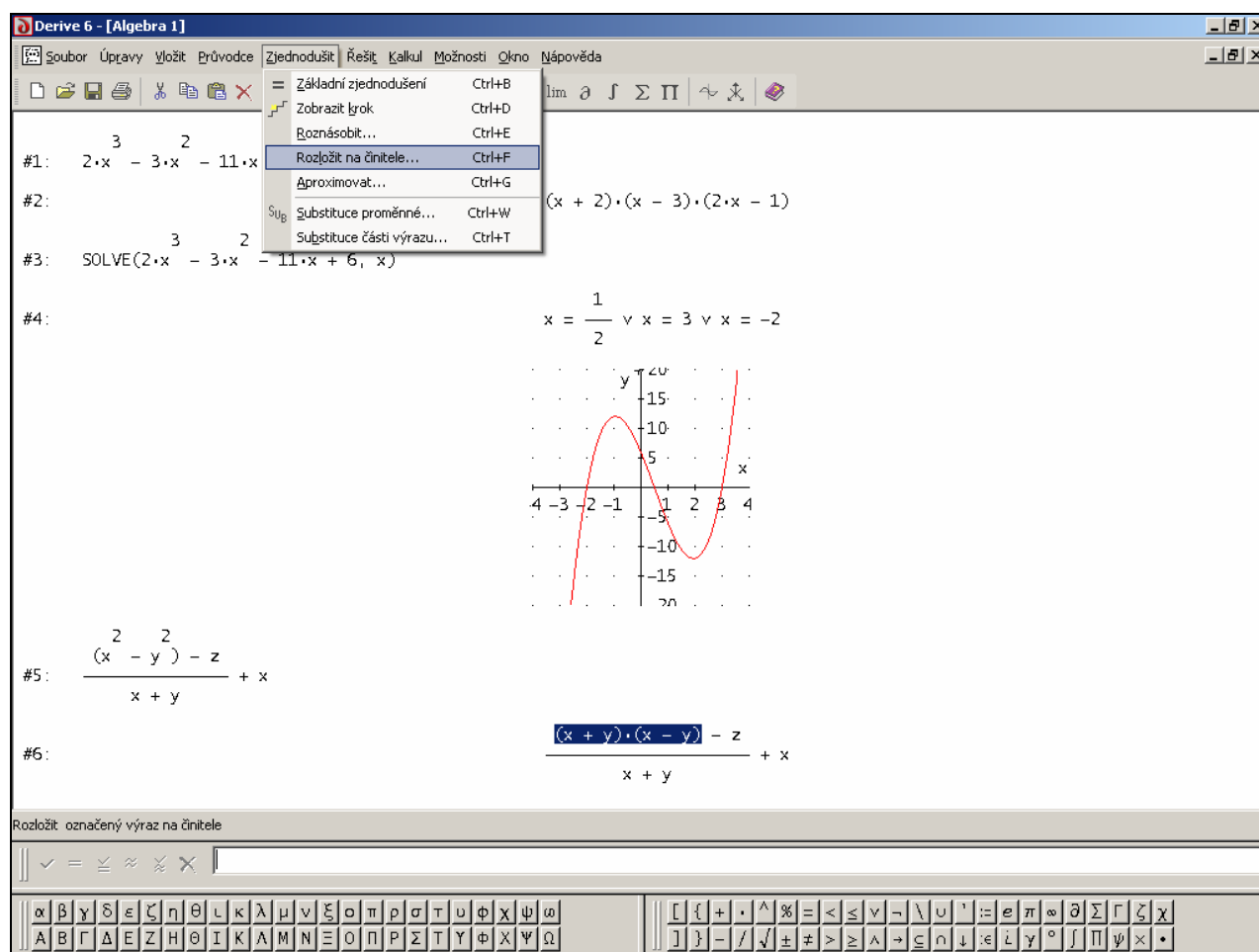
- možnost zobrazení úpravy výrazu krok za krokem s využitím dílčích transformačních pravidel
- přenos dat mezi programem a CAS kalkulačkami TI-89, TI-89 Titanium, TI-92+, Voyage 200 TI
- animace grafu funkce s parametrem pomocí změn hodnot parametru posuvníkem
- možnost automatického popisování jednotlivých grafů pomocí výrazů, které zobrazují
- otáčení 3D grafů pomocí myši
- snadná orientace v nápovědě s pomocí obsahu
- možnost přizpůsobení podoby nabídek, panelů nástrojů a klávesových zkratk vlastním potřebám uživatele
- výhody řady dalších vylepšení, například
  - písmo Derive Unicode pro zápis matematických výrazů
  - podpora znaků Unicode a html odkazů ve vloženém textu
  - stavové proměnné uložené přímo v dfw souborech
  - možnost volby režimu víceřádkové editace
  - párování závorek na příkazovém řádku
  - možnost ovlivnit zobrazení mřížky u 3D grafu a velikost bodů
  - funkce pro určení Gröbnerovy báze systému mnohočlenů

Pro dobré zvládnutí obsluhy programu doporučuji prostudování knihy „**Derive 6, Pokročilá matematika pro vaše PC**“ autorů B. Kutzlera a V. Kokol-Voljc, která je součástí dodávky programu Derive 6.

# I. OBSLUHA PROGRAMU


## 1. Grafické rozhraní

S programem komunikujeme prostřednictvím grafického rozhraní (viz **Obr. 1**). Řada jeho prvků je známa i z jiných programů pracujících pod operačním systémem Windows. Najdeme zde **nabídku akcí** (**Obr. 1**, druhý řádek), za nimiž se skrývají **roletová menu** s konkrétními příkazy. Ukážeme-li na příkaz myší, objeví se na příslušném místě jeho stručná **charakteristika** (**Obr. 1**, řádek pod velkou pracovní plochou). Nechybí ani **ikony pro rychlé provádění** některých příkazů (**Obr. 1**, třetí řádek). Najdeme zde i **panely nástrojů** pro rychlé vkládání písmen řecké abecedy či matematických symbolů. Samozřejmě, podoba rozhraní se dá **změnit** (Okno → Přizpůsobit...). **Nápověda** je tradičně z nabídky **Nápověda** nebo klávesou **F1**.





Obr. 1: Grafické rozhraní programu Derive 6





## 2. Vstup a výstup









Výrazy vkládáme do programu prostřednictvím **příkazového (vstupního) řádku** (Obr. 1, úzký bílý řádek při dolním okraji). Po napsání výrazu a stisknutí klávesy **Enter** nebo ikony  (nalevo od příkazového řádku) se výraz objeví na **pracovní ploše** (okno **Algebra**), která celému oknu dominuje. Každý výraz na pracovní ploše má své **číslo** (např. #1), které slouží i jako **proměnná**, pomocí níž se můžeme na výraz odvolávat. Chceme-li například sečíst dva výrazy, jež se na pracovní ploše vyskytují pod čísly #5 a #13, stačí na příkazový řádek napsat  $\#5 + \#13 =$  **Enter**.

Často budeme potřebovat **přecházet mezi příkazovým řádkem a pracovní plochou**. Můžeme to řešit klikáním myši nebo použít klávesy **Esc** (nahoru) a **F2** (dolů).

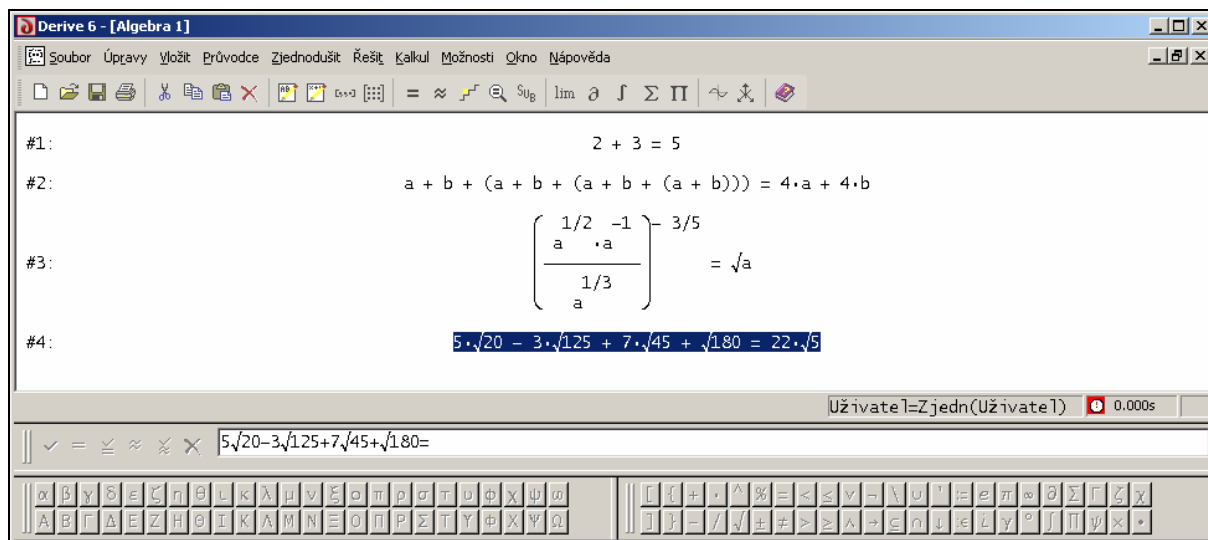
## 3. Manipulace s výrazy

**Po pracovní ploše se pohybuje** opět pomocí myši, nebo pomocí kláves  .

Zaznamenání historie naší práce na pracovní ploše nám může podstatně ušetřit čas. Potřebujeme-li vložit výraz, který na ploše již je, využijeme jeho číslo, jak je zmíněno výše, nebo si tento výraz najdeme a zvýrazníme kliknutím myši. Potom ho můžeme **přenést** na příkazový řádek **tak jak je** klávesou **F3** nebo **uzavřený v závorce** klávesou **F4**. Tuto druhou, klávesovou, možnost oceníme hlavně při potřebě využít jenom část již existujícího výrazu. Při pohybu po pracovní ploše (myši nebo šipkovými klávesami) se zvýrazňují výrazy celé. **Zvýraznění jenom části výrazu** docílíme buď opakovaným klikáním myši umístěnou nad příslušnou částí výrazu (tato část musí mít sama o sobě smysl jako výraz), nebo pomocí **kombinace kláves**     **při současném držení klávesy** **Shift**. Výrazy můžeme po pracovní ploše **přemisťovat** (tj. měnit jejich pořadí) jednoduše **uchopením myši a přetažením** na požadované místo. To, zda jim zůstane jejich původní **číslo**, nebo se změní dle nového pořadí, ovlivníme volbou **Možnosti** → **Zobrazení** → **Přečíslovat výrazy**. **Číslo výrazů** (označení) **můžeme skrýt** volbou **Možnosti** → **Skrýt** → **Označení**.

Při manipulaci s výrazy oceníme skupinu ikon       vlevo od příkazového řádku. Jejich význam je v pořadí zleva doprava tento: **zobrazit výraz** na pracovní ploše, výraz **zjednodušit a výsledek zobrazit** na ploše, na ploše **zobrazit výraz i výsledek jeho zjednodušení**, výraz **aproximovat**, výraz **zobrazit na ploše spolu s jeho aproximací**. Ikony ,  najdeme ještě na horní liště ikon nebo jim odpovídající akce můžeme vyvolat volbami **Zjednodušit** → **Základní zjednodušení**, **Zjednodušit** → **Aproximovat...**. Chceme-li

mít výraz i jeho zjednodušení na jednom řádku, napíšeme při vkládání za výraz znaménko = (rovná se) a odešleme stisknutím **Enter** (**Obr. 2**).



**Obr. 2: Přímé zjednodušení výrazu**

## 4. Tvorba dokumentů

Kromě výrazů a s nimi spojených výpočtů můžeme na pracovní ploše zobrazit **text** (**Vložit** → **Text...**), **OLE** (Object Linking and Embedding) objekty všech myslitelných aplikací (**Vložit** → **OLE Objekt...**) a samozřejmě **grafy** vytvořené samotným programem, jak vidíme na **Obr.1**, (**Vložit** → **2D-graf...**, **Vložit** → **3D-graf...**). Pokud si otevřeme panel nástrojů **Formátování** (**Okno** → **Přizpůsobit...** → **Panely nástrojů**), můžeme text běžným způsobem upravovat (řez, velikost, typ a barva písma, zarovnání, odrážky apod.). To nám umožňuje vytvářet v programu Derive **kompletní dokumenty** na úrovni textového editoru, například zadání samostatných prací, podklady pro vyučovací hodinu, vypracování laboratorní práce apod. Program umožňuje uložit dokument ve formátu **.rtf** (Rich Text Format) (**Soubor** → **Exportovat** → **RTF...**). Tak nejsme vázáni jenom na prostředí Derive, ale můžeme dokument otevřít a dále upravovat třeba ve Wordu případně převést do formátu **.pdf** apod. Podobně je tomu i u grafů. Ty můžeme uložit v několika grafických formátech a pak je použít v dalších aplikacích.

## 5. Příkazy

Zadání příkazu je v programu Derive chápáno jako zápis výrazu. Chceme-li, aby program zadaný příkaz vykonal, musíme jemu odpovídající výraz **zjednodušit**. Program Derive

podporuje dvojí způsob **zadáání příkazů**. Buď příkaz napíšeme na příkazový řádek, nebo ho zadáme prostřednictvím dialogových oken programu. První možnost je rychlejší, pokud známe syntaxi příkazu, jinak ztrácíme čas jejím hledáním v nápovědě. Druhá možnost je pohodlnější a přibližuje program uživateli. Pokrývá však jen část možností programu. Ten je **mnohem mocnějším nástrojem**, než by se zdálo při pouhém prozkoumání nabídky příkazů jeho rozhraní. Řadu speciálnějších příkazů, například na řešení diferenciálních rovnic, počítání s čísly, zkoumání geometrických vztahů apod., prostě musíme na příkazový řádek napsat. Množství nabízených příkazů můžeme posoudit prostým prolistováním obsahu nápovědy (Nápověda → Témata nápovědy → Obsah).

Kromě vestavěných funkcí pracuje program Derive se systémem **balíčků funkcí** (Nápověda → Témata nápovědy → Uživatelské balíčky matematických funkcí), což jsou soubory typu **.mth** (soubory obsahující funkce naprogramované přímo v Derive), které se **automaticky** načítají do paměti, jakmile je volána funkce v nich definovaná.

## 6. Programování

Derive umožňuje uživateli vytvářet **vlastní procedury a funkce**. Ukládáme je do souborů typu **.mth** a před použitím je musíme načíst do paměti (na rozdíl od vestavěných balíčků funkcí) příkazem Soubor → Importovat → Balíček funkcí...

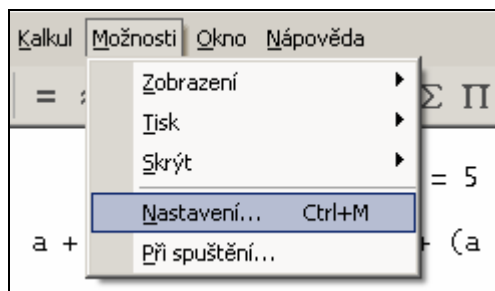
## 7. Deklarace proměnných

Při tvorbě vlastních příkazů a vůbec při práci na nějakém rozsáhlejší projektu v Derive nám mohou značně zkomplikovat život nějaká zapomenutá nastavení hodnot a oborů proměnných, které nevědomky použijeme znovu za jiným účelem. Těmto komplikacím, které jinak mohou přerůst až v naprostou ztrátu důvěry v sebe i v program, se vyhneme dodržováním následujících zásad. Pokud to jde, používáme **vždy nový soubor**, který načítá firemní nastavení z inicializačního souboru Derive6.ini. Pokud to nejde, hlídáme si již použité proměnné. K ověření jejich **hodnot a oborů** použijeme volby Průvodce → Hodnota proměnné... a Průvodce → Definiční obor proměnné... Konečně se nabízí postup běžný z programování. Každou proměnnou, kterou chci použít, **deklarovat**, tj. vymežit její obor, případně hodnotu. Pokud např. chceme, aby proměnná **A** byla bez hodnoty, tj. prázdná, použijeme **prázdný přiřazovací příkaz** **A:=** . Jsme-li tedy na konci nějakého rozsáhlejšího souboru a chceme použít proměnnou **A**, jejímž obsahem si nejsme jisti, provedeme nejprve tento příkaz.



## II. ŘEŠENÍ ÚLOH

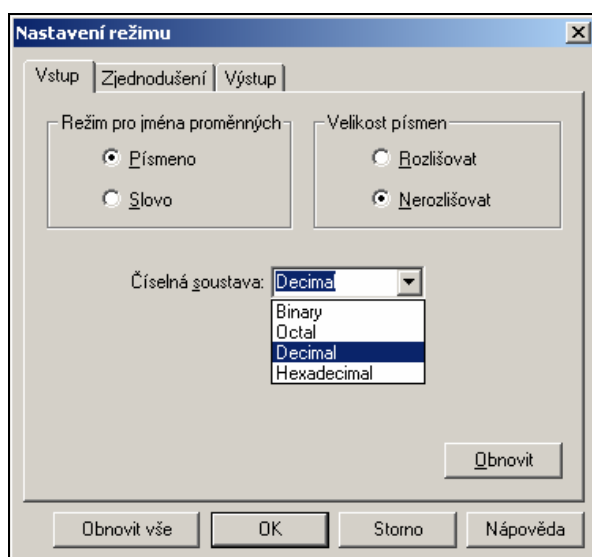
Pro nastavení vstupních a výstupních podmínek, definic a pravidel budeme hojně využívat nabídku **Možnosti** → **Nastavení...** (Obr. 3).



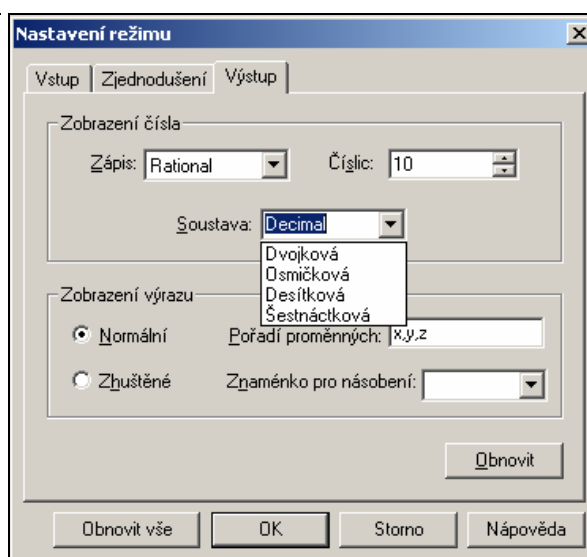
Obr. 3: Nastavení

### 1. Číselné soustavy

Například pro převod zápisu čísla do jiné číselné soustavy nebo přímo pro počítání v jiné číselné soustavě využijeme dialogové okno příkazu **Možnosti** → **Nastavení...** → **Vstup** (Obr. 4), konkrétně volbu **Číselná soustava**, která nám umožní nastavit základ číselné soustavy, v níž je reprezentováno **vkládáné číslo**. Program nám přímo nabízí čtyři základy (2...Binary, 8...Octal, 10...Decimal, 16...Hexadecimal), ale jinak je možné do příslušného bílého políčka vepsat **libovolné číslo - základ od 2 do 36**. Pro označení číslic větších než 10 použijeme odpovídající písmeno abecedy. Chceme-li psát číslice samostatně, píšeme před písmena nulu, např. 0A pro deset, 0C pro dvanáct. Analogicky, pro nastavení reprezentace číselné hodnoty výstupu, použijeme volbu **Možnosti** → **Nastavení...** → **Výstup** → **Soustava** (Obr. 5).



Obr. 4: Nastavení vstupních podmínek



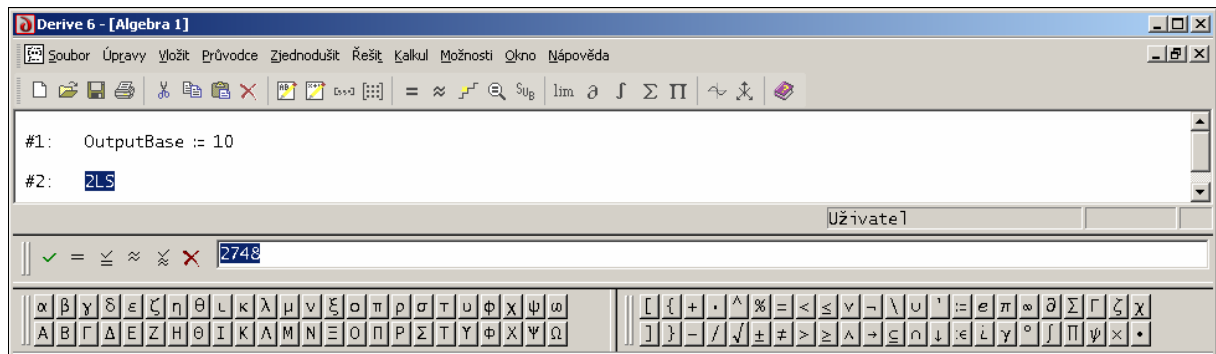
Obr. 5: Nastavení výstupních podmínek

**PŘÍKLAD 1.1 :** Převed'te číslo 2748 z desítkové soustavy do soustavy o základu 32.

*Řešení:*

Možnosti → Nastavení... → Vstup → Číselná soustava → Decimal , Možnosti → Nastavení... → Výstup → Soustava → 32

Na příkazový řádek píšeme 2748, ale na pracovní ploše se po  (nebo ) objeví zápis 2LS v soustavě o základu 32 (viz **Obr. 6**).



**Obr. 6:** Převod do soustavy o základu 32

**PŘÍKLAD 1.2:** Ve dvojkové soustavě proveďte výpočty  $10010 + 111$ ,  $10010 \cdot 111$ ,  $10010/111$

*Řešení:*

Možnosti → Nastavení... → Vstup → Číselná soustava → Binary , Možnosti → Nastavení... → Výstup → Soustava → Binary

```
#1:  InputBase := Binary
#2:  OutputBase := Binary
#3:                                     10010 + 111 = 11001
#4:                                     10010 · 111 = 1111110
#5:                                     10010   10010
                                     ----- = -----
                                     111       111
#6:                                     10010
                                     ----- = 10.10010010
                                     111
```

**Obr. 7:** Počítání ve dvojkové soustavě

Vyjádření výsledku desetinným číslem na řádce #6 (**Obr. 7**) dostaneme zvýrazněním pravé strany rovnosti #5 a stisknutím ikony  pro aproximaci vybraného výrazu. K původnímu nastavení programu se nejrychleji vrátíme volbou Možnosti → Nastavení... → Obnovit vše.

## 2. Funkce teorie čísel

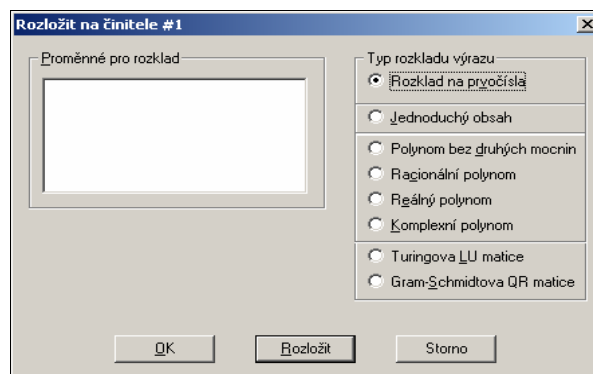
### 2.1. Rozklad na prvočinitele

**PŘÍKLAD 2.1:** Proved'te rozklad čísla 4956 na prvočinitele.

Řešení:

4956

**Zjednodužit** → **Rozložit na činitele...** (předpokládáme, že číslo je na pracovní ploše zvýrazněno). Objeví se dialogové okno (**Obr. 8**). Volíme typ rozkladu (i když, prvočíselný rozklad dostaneme pro jakýkoliv z uvedených typů). Můžeme si na tomto příkladu vyzkoušet, jaký je rozdíl mezi potvrzením tlačítkem **OK** a potvrzením tlačítkem **Rozložit**. V ukázce na **Obr. 9** je použita varianta **OK**, potom se na pracovní ploše objeví **zápis odpovídajícího příkazu** programu Derive a my ho musíme ještě **vykonat** příkazem **Zjednodužit** → **Základní zjednodušení** nebo rychleji ikonou **=** (Pozor, aby se příkaz skutečně vykonal, musí být zvýrazněn). Tato dvojí možnost provedení - **přímo** nebo **se zobrazením syntaxe a následným potvrzením** - je nabízena i u jiných příkazů v nabídce. Pamatujme si tedy, že příkazy vykonáváme příkazem **Zjednodužit** → **Základní zjednodušení** (**=**).



**Obr. 8:** Rozklad výrazu na činitele - Dialogové okno

#1: 4956

#2: FACTOR(4956, Number)

#3:  $2^2 \cdot 3 \cdot 7 \cdot 59$

**Obr. 9:** Rozklad čísla na prvočinitele

## 2.2. Největší společný dělitel

Jak už bylo uvedeno výše, program má mnohem více funkcí, než je vidět v nabídce. V souvislosti s teorií čísel stojí za zmínku například funkce (Nápověda → Témata nápovědy → Vestavěné funkce a konstanty → Funkce teorie čísel):

**GCD(n1,n2,...,nn)** ... největší společný dělitel (*the greatest common divisor*),

**LCM(n1,n2,...,nn)** ... nejmenší společný násobek (*the least common multiple*),

**PRIME?(n)** ... test prvočísla,

**NEXT\_PRIME(n)** (**PREVIOUS\_PRIME(n)**) ... následující (předchozí) prvočíslu.

Další funkce najdeme v balíčku **NUMBER.MTH** (Nápověda → Témata nápovědy → Uživatelské balíčky matematických funkcí).

**PŘÍKLAD 2.2:** Najděte taková celá čísla  $x, y$ , pro která platí

$$754x + 221y = D(754,221)$$

*Řešení:*

Kromě výše zmíněné funkce **GCD** použijeme příkaz **SOLVE**

(**Řešit** → **Výraz**) a příkaz **VECTOR(f(n),n,d,h,k)**, který nám poskytne libovolnou posloupnost hodnot výrazu **f(n)** s proměnnou **n** pokud zadáme ještě dolní a horní meze **d, h** a krok **k** (je-li hodnota kroku 1, nemusíme ji uvádět) (**Obr. 10**).

The screenshot shows the following steps in the software:


- #1:  $754 \cdot x + 221 \cdot y = \text{GCD}(754, 221)$
- #2:  $\text{SOLVE}(754 \cdot x + 221 \cdot y = \text{GCD}(754, 221), [x, y], \text{Real})$
- #3:  $58 \cdot x + 17 \cdot y = 1$
- #4:  $\text{SOLVE}(58 \cdot x + 17 \cdot y = 1, y, \text{Real})$
- #5:  $y = \frac{1 - 58 \cdot x}{17}$
- #6:  $\text{VECTOR}\left(\frac{1 - 58 \cdot x}{17}, x, -15, 15\right)$
- #7: A list of 15 integer solutions for  $x$  and  $y$ :
 

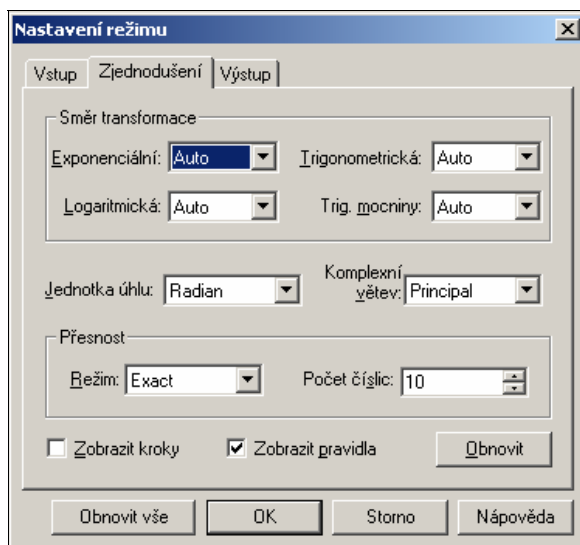
$\frac{871}{17}$	$\frac{813}{17}$	$\frac{755}{17}$	41	$\frac{639}{17}$	$\frac{581}{17}$	$\frac{523}{17}$	$\frac{465}{17}$	$\frac{407}{17}$	$\frac{349}{17}$	$\frac{291}{17}$	$\frac{233}{17}$	$\frac{175}{17}$	$\frac{117}{17}$	$\frac{59}{17}$	$\frac{1}{17}$
$\frac{57}{17}$	$-\frac{115}{17}$	$-\frac{173}{17}$	$-\frac{231}{17}$	-17	$-\frac{347}{17}$	$-\frac{405}{17}$	$-\frac{463}{17}$	$-\frac{521}{17}$	$-\frac{579}{17}$	$-\frac{637}{17}$	$-\frac{695}{17}$	-	-	-	-
$\frac{753}{17}$	$-\frac{811}{17}$	$-\frac{869}{17}$	-	-	-	-	-	-	-	-	-	-	-	-	-

**Obr. 10:** Největší společný dělitel GCD. Příkaz VECTOR

### 3. Úprava po krocích

Na výše uvedeném příkladu 2.1 si ukážeme použití voleb **Zobrazit krok(y)** a **Zobrazit pravidla**, které jsou od verze 6 novinkou programu Derive.

Úpravy po krocích můžeme provádět buď pomocí tlačítka (ikony)  na panelu nástrojů nebo tak, že potvrdíme volbu **Zobrazit kroky** na kartě **Možnosti** → **Nastavení...** → **Zjednodušení** (Obr. 11), kde najdeme i volbu **Zobrazit pravidla** prováděných úprav.



Obr. 11: Nastavení režimu Zjednodušení

Pojďme tedy potvrdit obě uvedené volby a znovu zkusit řešit příklad 2.1. Rozklad daného čísla má potom jiný průběh – provádí se skutečně krok za krokem. Je třeba si to uvědomit a provádět zadaný příkaz až do konce. Řešení rozkladu daného čísla na prvočinitele potom na pracovní ploše vypadá takto:

```
#1: DisplaySteps := true
```

```
#2: 4956
```

```
#3: FACTOR(4956, Number)
```

```
If  $n^p$  divides  $m$ ,
```

$$m \rightarrow n \cdot \frac{m}{n^p}$$

```
#4: FACTOR(22 · 1239)
```

```
If  $n$  divides  $m$ ,
```

$m \rightarrow n \cdot \frac{m}{n}$	
#5:	$\text{FACTOR}(2^2 \cdot 3 \cdot 413)$
If n divides m,	
$m \rightarrow n \cdot \frac{m}{n}$	
#6:	$\text{FACTOR}(2^2 \cdot 3 \cdot 7 \cdot 59)$
If n is fully factored,	
$\text{FACTOR}(n) \rightarrow n$	
#7:	$2^2 \cdot 3 \cdot 7 \cdot 59$

## 4. Úpravy výrazů

Podívejme se znovu na **Obr. 4: Nastavení vstupních podmínek**. Kromě volby nastavení základu číselné soustavy reprezentace vkládaných hodnot tam najdeme ještě volbu **Režim pro jména proměnných**, jíž můžeme ovlivnit **reprezentaci proměnných**. Zvolíme-li **Písmeno**, program uvažuje každé písmeno jako zvláštní proměnnou. Protože **můžeme vynechávat symbol násobení (\*)** stejně jako při zápisu na papír, chápe program při nastavení volby **Písmeno** zápis `abc` jako výraz `a*b*c`. Zvolíme-li **Slovo**, chápe zápis `abc` jako symbol jedné proměnné. Volba **Velikost písmen** ovlivňuje, zda záleží (**Rozlišovat**) či nezáleží (**Nerozlišovat**) na velikosti písmene (tj., zda `X`, `x` znamená to samé, či nikoliv).

### 4.1. Algebraické výrazy

Při úpravách algebraických výrazů využijeme volby **Zjednodušit** → **Základní zjednodušení**, **Roznásobit...**, **Rozložit na činitele...**. Význam voleb **Typ rozkladu...** v příslušných dialogových oknech ilustruje **Obr. 12**. Zároveň vidíme syntaxi příkazu **FACTOR(f(x), Typ rozkladu, x)**. Typy výrazu jsou definovány klíčovými slovy programu Derive, proto musí být v případě přímého zadání příkazu uvedeny v originálním znění (Trivial, Squarefree, ...).

#1:	$v := 3 \cdot x^9 + 6 \cdot x^8 - 6 \cdot x^7 - 6 \cdot x^6 - 15 \cdot x^5 - 48 \cdot x^4 + 66 \cdot x^3 + 72 \cdot x^2 - 72 \cdot x$
#2:	$\text{FACTOR}(v, \text{Trivial}, x)$
#3:	$3 \cdot x \cdot (x^8 + 2 \cdot x^7 - 2 \cdot x^6 - 2 \cdot x^5 - 5 \cdot x^4 - 16 \cdot x^3 + 22 \cdot x^2 + 24 \cdot x - 24)$
#4:	$\text{FACTOR}(v, \text{Squarefree}, x)$
#5:	$3 \cdot x \cdot (x^2 + x - 2)^2 \cdot (x^4 + x^2 - 6)$
#6:	$\text{FACTOR}(v, \text{Rational}, x)$
#7:	$3 \cdot x \cdot (x - 1)^2 \cdot (x + 2)^2 \cdot (x^2 - 2) \cdot (x^2 + 3)$
#8:	$\text{FACTOR}(v, \text{Radical}, x)$
#9:	$3 \cdot x \cdot (x - 1)^2 \cdot (x + 2)^2 \cdot (x + \sqrt{2}) \cdot (x - \sqrt{2}) \cdot (x^2 + 3)$
#10:	$\text{FACTOR}(v, \text{Complex}, x)$
#11:	$3 \cdot x \cdot (x - 1)^2 \cdot (x + 2)^2 \cdot (x + \sqrt{2}) \cdot (x - \sqrt{2}) \cdot (x + \sqrt{3} \cdot i) \cdot (x - \sqrt{3} \cdot i)$

Obr. 12: Volba *Typu rozkladu* příkazu **FACTOR** (**EXPAND**)

**PŘÍKLAD 4.1:** Zjednodušte lomený výraz  $\frac{2x-1}{x+1}$ .

*Řešení:*

#1: 
$$\frac{2 \cdot x - 1}{x + 1}$$

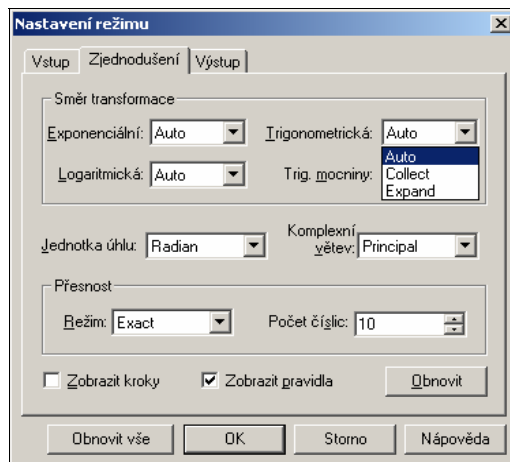
#2: 
$$\text{EXPAND}\left(\frac{2 \cdot x - 1}{x + 1}, \text{Trivial}, x\right)$$

#3: 
$$2 - \frac{3}{x + 1}$$

Naopak, pro převedení zlomku na společného jmenovatele použijeme příkaz **FACTOR** (**Rozložit na činitele...**).

## 4.2. Goniometrické výrazy

Chceme-li upravovat goniometrické výrazy, můžeme (někdy musíme) sdělit programu, jak si to přesně představujeme. Provedeme to volbou **Možnosti** → **Nastavení...** → **Zjednodušení** (viz **Obr. 3**), která otevře dialogové okno (**Obr. 14**),



**Obr. 14: Nastavení režimu zjednodušování**

v němž jsou nabídnuty různé možnosti a režimy úprav výrazů:

**Trigonometrická** → Auto, Collect, Expand (určuje **směr využití goniometrických vzorců**),

**Trig. mocniny** → Auto, Cosines, Sines (**reprezentace sudých mocnin gon. funkcí**),

**Jednotka úhlu** → Radian, Degree (**jednotka velikosti úhlu**),

**Komplexní větev** → Principal, Real, Any (**volba větve řešení odmocniny v oboru komplexních čísel**, oceníme např. při výpočtu  $\sqrt[3]{-8}$ ).

**PŘÍKLAD 4.2:** Zjednodušte výraz  $\frac{(\sin \alpha + \sin \beta)^2}{\cos^2 \alpha - \cos^2 \beta}$

*Řešení:*

$$\begin{aligned} \#1: & \frac{(\text{SIN}(\alpha) + \text{SIN}(\beta))^2}{\text{COS}(\alpha)^2 - \text{COS}(\beta)^2} \\ \#2: & \text{FACTOR} \left( \frac{(\text{SIN}(\alpha) + \text{SIN}(\beta))^2}{\text{COS}(\alpha)^2 - \text{COS}(\beta)^2}, \text{Radical}, \alpha, \beta \right) \\ \#3: & \frac{\text{SIN}(\alpha) + \text{SIN}(\beta)}{\text{SIN}(\beta) - \text{SIN}(\alpha)} \end{aligned}$$



**PŘÍKLAD 4.3:** Rozložte výraz  $\sin(x + y)$ 

Řešení:

Možnosti → Nastavení... → Zjednodušení → Trigonometrická → Expand

$$\sin(x+y) =$$

Pracovní plocha (okno Algebra) při potvrzené volbě **Zobrazit kroky**:

#1: Trigonometry := Expand

#2:  $\text{SIN}(x + y)$  $\text{SIN}(z + w) \rightarrow \text{SIN}(z) \cdot \text{COS}(w) + \text{COS}(z) \cdot \text{SIN}(w)$ #3:  $\text{SIN}(x) \cdot \text{COS}(y) + \text{COS}(x) \cdot \text{SIN}(y)$ **PŘÍKLAD 4.4:** Upravte výraz  $\sin(x)^2 + 2 \cos(x)^2 - 1$ 

Řešení:

$$\sin(x)^2 + 2\cos(x)^2 - 1 =$$

#1:  $\text{SIN}(x)^2 + 2 \cdot \text{COS}(x)^2 - 1$  $\text{SIN}(z)^2 + \text{COS}(z)^2 \rightarrow 1$ #2:  $\text{COS}(x)^2$ **PŘÍKLAD 4.5:** Zjednodušte výraz  $\cos(x)^2 - \sin(x)^2$ 

Řešení:

$$\cos(x)^2 - \sin(x)^2 =$$

#1:  $\text{COS}(x)^2 - \text{SIN}(x)^2$  $\text{SIN}(z)^2 + \text{COS}(z)^2 \rightarrow 1$ #2:  $2 \cdot \text{COS}(x)^2 - 1$ 

Možnosti → Nastavení... → Zjednodušení → Trigonometrická → Collect

$$\cos(x)^2 - \sin(x)^2 = =$$

*Poznámka: Nesmíme zapomenout na to, že při nastaveném režimu **Zobrazit kroky** odpovídá jednomu zjednodušení, tj. jednomu stisknutí **=**, jeden krok. Proto často musíme mačkat ikonu **=** opakovaně (viz následující pokračování řešení příkladu 4.5).*

#3: Trigonometry := Collect

$\text{COS}(z + \pi) \rightarrow -\text{COS}(z)$

#4: 
$$\frac{\text{COS}(2 \cdot x)}{2} + \frac{1}{2} - \text{SIN}(x)^2$$

#5: 
$$\text{COS}(2 \cdot x)$$

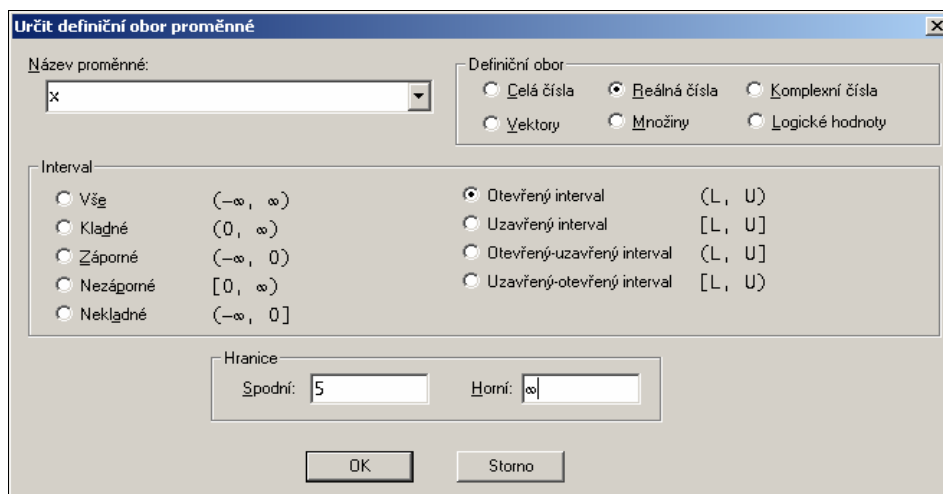
### 4.3. Logaritmické výrazy

**LN(x)** nebo **LOG(x)** jsou zápisy **přirozeného logaritmu** čísla  $x$ . Logaritmus  $x$  při základu  $z$  zapisujeme ve tvaru **LOG(x,z)**. Při úpravách logaritmických výrazů může hrát roli definiční obor logaritmické funkce. Program tuto skutečnost zohledňuje a některým úpravám proto musí předcházet náležité vymezení definičních oborů příslušných proměnných. K tomu použijeme volbu

**Průvodce** → **Definiční obor proměnné ...** (**Obr. 16**)

Stejně jako u goniometrických funkcí můžeme specifikovat způsob úpravy volbami

**Možnosti** → **Nastavení...** → **Zjednodušení** → **Logaritmická** → Auto, Collect, Expand



**Obr. 16: Definiční obor proměnné ...  $x \in (5, \infty)$**

**PŘÍKLAD 4.6:** Rozložte na součet logaritmů výrazy  $\ln(xy)$ ,  $\ln((x-5)(y-10))$ .

*Řešení:*

Aby se logaritmus součinu rozložil na součet logaritmů, musíme náležitě nastavit způsob (směr) úprav logaritmických výrazů a definovat definiční obor alespoň jednoho z činitelů.

```
#1: DisplaySteps := true
```

```
#2: LOG(x·y)
```

```
LOG(z) → LN(z)
```

```
#3: LN(x·y)
```

```
#4: Logarithm := Expand
```

```
#5: x :∈ Real (0, ∞)
```

Protože máme nastaven režim **Zobrazit kroky**, zjednodušení se provede ve dvou krocích.

Z komentáře řešení (máme zapnutou i volbu **Zobrazit pravidla**) je vidět, že program bere při úpravě v potaz definiční obor jedné z proměnných (If  $x > 0$  ...).

```
LOG(z) → LN(z)
```

```
#6: LN(x·y)
```

```
If x>0,
```

```
LN(x·z) → LN(x) + LN(z)
```

```
#7: LN(x) + LN(y)
```

```
#8: LN((x - 5)·(y - 10))
```

```
#9: x :∈ Real (5, ∞)
```



```
If x>0,
```

```
LN(x·z) → LN(x) + LN(z)
```

```
#10: LN(x - 5) + LN(y - 10)
```

## 5. Numerické výpočty, konstanty

Číselné výrazy mohou být reprezentovány jak přesnými zápisy (např.  $2^{(1/2)} = \sqrt{2}$ ) tak i přibližnými hodnotami ( $2^{(1/2)} = 1.414213562$ ). Zápis desetinným číslem přitom může mít libovolný počet míst (jsme omezeni jenom možnostmi počítače).

Při úpravách výrazů tyto dvě možnosti volíme pomocí příkazů (ikon) **Zjednodušit** → **Základní zjednodušení** () a **Zjednodušit** → **Aproximovat...** ().

Pro trvalé nastavení režimu reprezentace hodnot použijeme volbu (**Obr. 14**) **Možnosti** → **Nastavení...** → **Zjednodušení** → **Přesnost** → **Režim** → Approximate, Exact, Mixed (v režimu Mixed se aproximují pouze iracionální čísla).

Počet míst číselného zápisu nastavíme tamtéž **Nastavení...** → **Zjednodušení** → **Přesnost** → **Počet číslic**. Tak můžeme zobrazit třeba  $\pi$  klidně i na 1000 desetinných míst apod. Konstanty jako je Ludolfovo nebo Eulerovo číslo vkládáme z příslušného panelu nástrojů nebo použijeme kombinace kláves **Ctrl** + **P** (**E**).

**PŘÍKLAD 5.1:** Jaké jsou poslední dvě číslice čísla  $3^{1234}$  ?

*Řešení:*

```
#1:      1234
#1:      3

#2: 585636752993207126904960872641502843975714362344306378546878484353~
6946634255003858796842453459905186846046851774292445753149532138~
6484091405615938232224285592002838163675935715569427707112038949~
4731892250998612899503423317714024722169594167914617121567515815~
4358685224309073135481486540427499644363484815218991488230440687~
3590219651150676785452123657256782428784859208854369270303725801~
4725677578888237430186684544106507672649961347986821041054836605~
6401445912760529623658838510267592199677099676046090140514526228~
7975259269775149323949547964577274163980814826612628072882293894~
63819882569
```

## 6. Rovnice a soustavy rovnic

Program Derive umožňuje jednoduše a rychle kombinovat symbolické, numerické a grafické řešení rovnic. Jak je patrné z následující ukázky (**Obr. 18**), můžeme v programu vytvářet i přehledné dokumenty o řešení určitého problému.

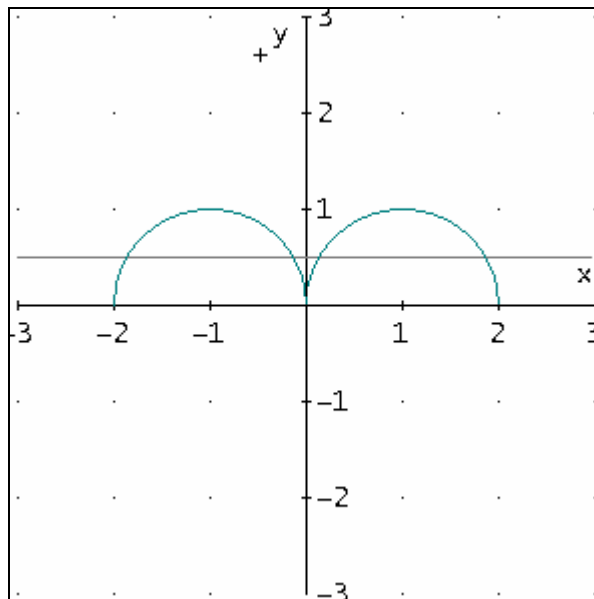
Řešte v  $\mathbf{R}$  rovnici:

$$\sqrt{(2 \cdot |x| - x^2)} = \frac{1}{2}$$

$$\text{SOLVE}\left(\sqrt{(2 \cdot |x| - x^2)} = \frac{1}{2}, x, \text{Real}\right)$$

$$x = -\frac{\sqrt{3}}{2} - 1 \vee x = 1 - \frac{\sqrt{3}}{2} \vee x = \frac{\sqrt{3}}{2} - 1 \vee x = \frac{\sqrt{3}}{2} + 1$$

$$x = -0.1339745962 \vee x = 0.1339745962 \vee x = -1.866025403 \vee x = 1.866025403$$



Obr. 18


## 6.1. Grafické řešení rovnic. Tvorba grafu.

**PŘÍKLAD 6.1:** V R řešte rovnici  $\sqrt{2|x| - x^2} = \frac{1}{2}$

Řešení:

$$\sqrt{2|x| - x^2} = \frac{1}{2} \text{ Enter}$$





**Řešit** → **Výraz...** zvolíme algebraické (symbolické) řešení v oboru reálných čísel (**Obr. 19**).

Chceme-li znát výsledek i ve formě desetinného čísla (jako je na **Obr. 18**), zvýrazníme na pracovní ploše výsledek algebraického (symbolického) řešení a stiskneme tlačítko  (Aproximovat).

Při grafickém řešení uplatníme následující postup:

### Tvorba 2D (3D) grafu

**Grafické znázornění** nějakého **výrazu** (výsledek rovnice či nerovnice, předpis funkce, podmínky definující relaci, ...) dostaneme následujícím způsobem:

- 1) výraz v okně Algebra zvýrazníme,
- 2) klikneme na  (  ), objeví se okno 2-D (3-D) grafu,
- 3) klikneme ještě jednou na  (  ) (tlačítko je teď na jiném místě), nakreslí se graf,
- 4) pro větší názornost můžeme rozdělit pracovní plochu na dvě části, jednu pro okno 2-D (3-D) grafu, druhé pro okno Algebra : **Okno** → **Vertikální dlaždice**

Program nám umožňuje výše uvedeným postupem zobrazit **i jenom část výrazu** (když jí zvýrazníme). To využijeme při grafickém řešení rovnic a nerovnic.

Výše uvedeným způsobem zobrazíme zvlášť levou a zvlášť pravou stranu rovnice (ovšem do jednoho 2-D grafického okna). Grafické řešení přeneseme na pracovní plochu do okna Algebra příkazem **Soubor** → **Přemístit**. Tento příkaz patří do souboru příkazů spojených s grafickým oknem, můžeme ho tedy najít v nabídce a provést jenom tehdy, když je aktivní grafické okno. Okna aktivujeme poklepáním myši. Všimněte si, že se tím mění i skladba příkazů a ikon. Při **změně velikosti** grafu přeneseného do okna Algebra se zhoršuje kvalita obrázku. **Původní kvality**, při nové velikosti, dosáhneme příkazem **Soubor** → **Aktualizovat** z nabídky 2-D grafického okna.

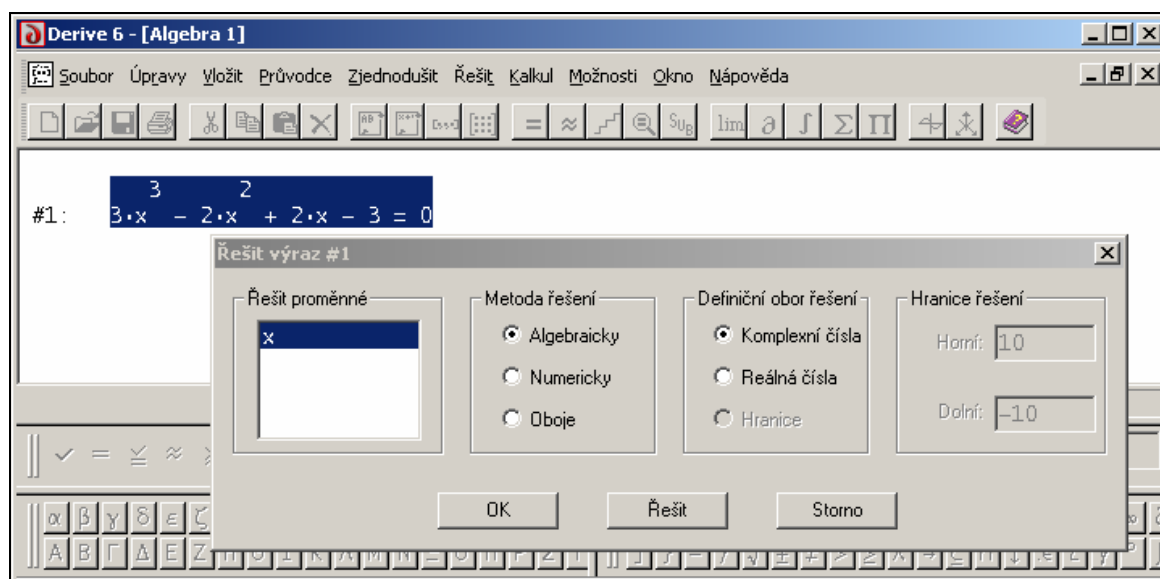
## 6.2. Algebraické rovnice

**PŘÍKLAD 6.2:** V R řešte rovnici  $3x^3 - 2x^2 + 2x - 3 = 0$

Řešení:

$$3x^3 - 2x^2 + 2x - 3 = 0 \quad \text{Enter}$$

Řešit → Výraz... zvolíme metodu a obor řešení a potvrdíme OK nebo Řešit. Rozdíl už známe.



Obr.19: Řešení algebraické rovnice

$$\#1: \quad 3 \cdot x^3 - 2 \cdot x^2 + 2 \cdot x - 3 = 0$$

$$\#2: \quad \text{SOLVE}(3 \cdot x^3 - 2 \cdot x^2 + 2 \cdot x - 3 = 0, x)$$

$$\#3: \quad x = -\frac{1}{6} - \frac{\sqrt{35 \cdot i}}{6} \vee x = -\frac{1}{6} + \frac{\sqrt{35 \cdot i}}{6} \vee x = 1$$

## 6.3. Úpravy rovnic

Program Derive umožňuje provádět **ekvivalentní úpravy rovnic** přímo na pracovní ploše.

Chceme-li například k oběma stranám rovnice

$$\#10: 2x - 5 = 10$$

přičíst 5, napíšeme na příkazový řádek jednoduše #10+5 nebo rovnici zkopírujeme do závorek pomocí  $\boxed{F4}$  a přičteme k ní onu pětku

$$\#11: (2x - 5 = 10) + 5$$

**PŘÍKLAD 6.3:** Řešte soustavu rovnic převedením na Gaussův tvar

$$2x - 5y = 4$$

$$x + 3y = 1$$

*Řešení:*

Výše uvedeným způsobem upravujeme dané rovnice.

Pro dosazení hodnoty  $y$  (řádek #6) do zbývající rovnice (#5) použijeme příkaz **Zjednodušit** → **Substituce proměnné...**

#1:  $2 \cdot x - 5 \cdot y = 4$

#2:  $x + 3 \cdot y = 1$

#3:  $x + 3 \cdot y = 1$

#4:  $- 11 \cdot y = 2$

#5:  $x + 3 \cdot y = 1$

#6:  $y = - \frac{2}{11}$

#7:  $x - \frac{6}{11} = 1$

#8:  $x = \frac{17}{11}$

Nejde-li nám o názornost, použijeme samozřejmě příkazu **SOLVE**, který vyvoláme volbou

**Řešit** → **Soustavu rovnic...**

#9:  $\text{SOLVE}([2 \cdot x - 5 \cdot y = 4, x + 3 \cdot y = 1], [x, y])$

#10:  $\left[ x = \frac{17}{11} \wedge y = - \frac{2}{11} \right]$



## 6.4. Soustavy rovnic

**PŘÍKLAD 6.3:** Řešte v  $\mathbb{R}$  soustavu lineárních rovnic

$$x + 3y + z = 5$$

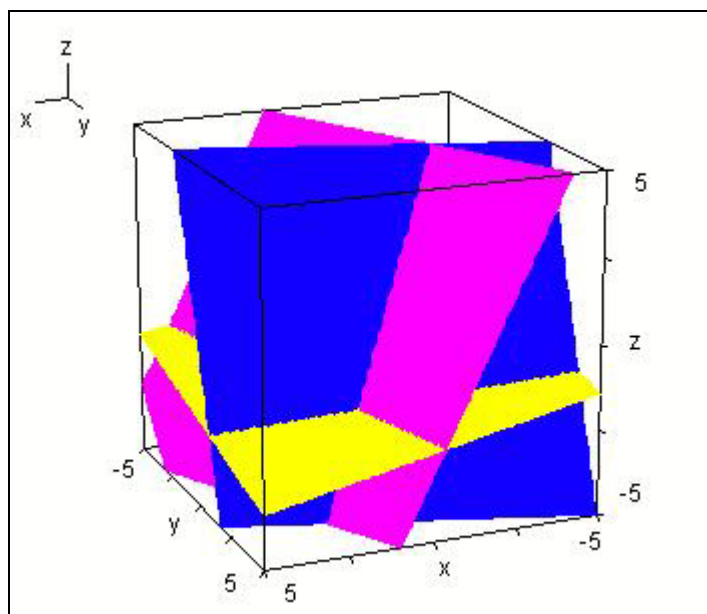
$$2x + y + z = 2$$

$$x + y + 5z = -7$$

Řešení:

Řešit → Soustavu rovnic...

Algebraické řešení můžeme velice snadno a rychle doplnit obrázkem. Stačí postupně zvýrazňovat rovnice, které se, po potvrzení tlačítka **Řešit** v příslušném dialogovém okně, objeví jako parametry příkazu **SOLVE** na pracovní ploše a výše popsáním způsobem je zobrazovat v 3-D grafickém okně.



Obr. 20: Grafické řešení soustavy rovnic

Uvedený postup řešení soustavy je jednoduchý a rychlý, protože k zadání příkazu **SOLVE** využívá grafické rozhraní. V programu Derive jsou, mimo rámec grafického rozhraní, k dispozici mnohé další příkazy použitelné při zkoumání soustav lineárních rovnic. Zájemce odkazují na **Nápovědu**, konkrétně na témata: **Témata nápovědy** → **Vybraná témata** → **Řešení rovnic, Vektory a matice**; **Témata nápovědy** → **Knihovna funkcí** → **Řešení rovnic, Lineární algebra**; **Témata nápovědy** → **Uživatelské balíčky matematických funkcí**.

## 6.5. Goniometrické rovnice

**PŘÍKLAD 6.4:** Řešte v  $\mathbb{R}$  rovnici  $\sin\left(x + \frac{\pi}{2}\right) = \frac{1}{2}$

Řešení:

#1:  $\text{SIN}\left(x + \frac{\pi}{2}\right) = \frac{1}{2}$

#2:  $\text{SOLVE}\left(\text{SIN}\left(x + \frac{\pi}{2}\right) = \frac{1}{2}, x\right)$

#3:  $x = \frac{5 \cdot \pi}{3} \vee x = -\frac{\pi}{3} \vee x = \frac{\pi}{3}$

## 7. Nerovnice a soustavy nerovnic

**PŘÍKLAD 7.1:** V  $\mathbb{R}$  řešte nerovnici  $|x - 3| > 1$ . Řešení graficky znázorněte.

Řešení:

`abs(x-3)>1`

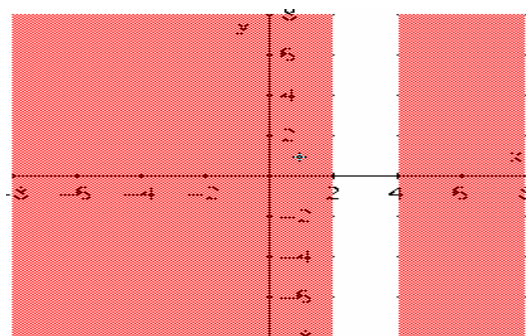
→

Grafické řešení dle výše uvedeného postupu (zobrazujeme výsledek řešení nerovnice nebo nerovnici samotnou).

#1:  $|x - 3| > 1$

#2:  $\text{SOLVE}(|x - 3| > 1, x, \text{Real})$

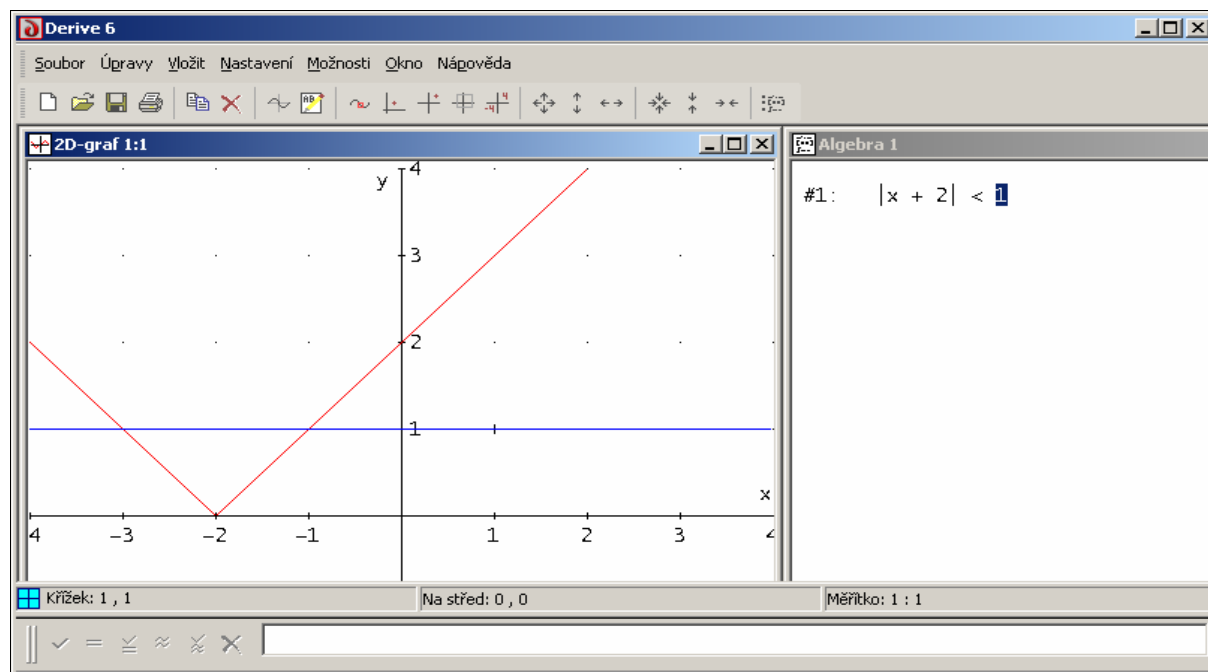
#3:  $x < 2 \vee x > 4$



**PŘÍKLAD 7.2:** Řešte graficky nerovnici  $|x + 2| < 1$

*Řešení:*

V tomto případě zobrazíme každou stranu nerovnice zvlášť.



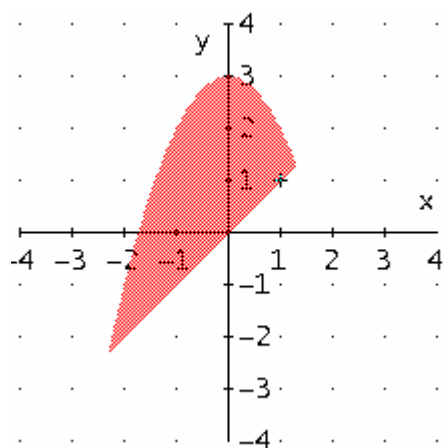
**PŘÍKLAD 7.3:** Řešte graficky soustavu nerovnic:

$$\begin{aligned}x^2 + y &\leq 3 \\ y - x &\geq 0\end{aligned}$$

*Řešení:*

Nerovnice spojíme logickou spojkou AND, celý takto vzniklý výraz zvýrazníme a zobrazíme.

#1:  $x^2 + y \leq 3 \wedge y - x \geq 0$



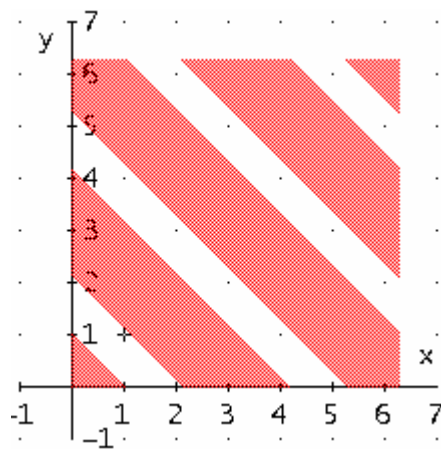
## 8. Relace

**PŘÍKLAD 8.1:** Vyšetřete množinu bodů v rovině, jejichž pravoúhlé souřadnice  $x, y$  splňují soustavu daných nerovnic. Načrtněte obrázek.

a) *Poznámka:* Při úpravě zadání relace (pro zobrazení ale není nutná) můžeme sledovat postup úprav pomocí režimu **Zobrazit kroky**.

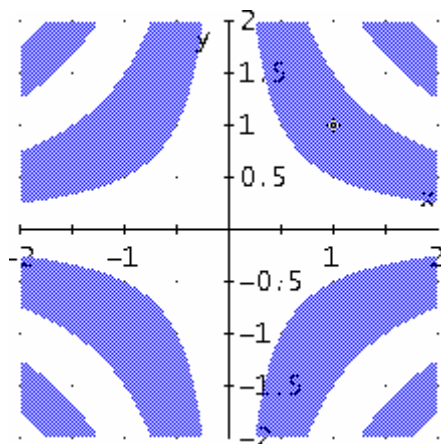
#1:  $\left| \sin\left(x + y + \frac{\pi}{2}\right) \right| \geq \frac{1}{2} \wedge 0 \leq x \leq 2 \cdot \pi \wedge 0 \leq y \leq 2 \cdot \pi$

#2:  $\left( \cos(x + y) \leq -\frac{1}{2} \vee \cos(x + y) \geq \frac{1}{2} \right) \wedge x \leq 2 \cdot \pi \wedge y \leq 2 \cdot \pi \wedge x \geq 0 \wedge y \geq 0$



b)

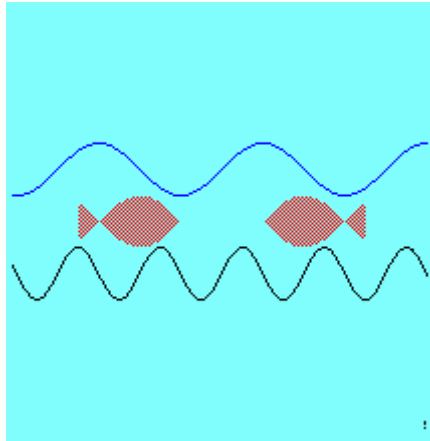
#3:  $|x| < 2 \wedge |y| < 2 \wedge \cos(\pi \cdot x \cdot y) \leq 0$



### c) Setkání v akváriu

(Emil Calda: *Z umělecké dílny profesora Ypsilon*, Rozhledy M-F, č. 9-10, 1983)

$$\#1: \left( |x| \geq \frac{\pi}{2} \wedge |x| \leq \frac{7 \cdot \pi}{4} \wedge |y| \leq |\cos(x)| \right) \vee y = 2 + \sin(x) \vee y = -2 + \sin(2 \cdot x)$$



### d) Měsíční noc

(Emil Calda: *Z umělecké dílny profesora Ypsilon*, Rozhledy M-F, č. 9-10, 1983)

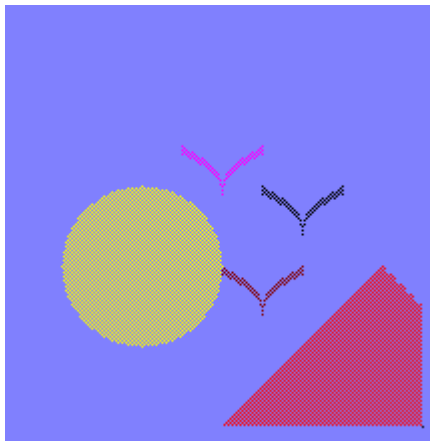
$$\#1: |x| \leq 1 \wedge y \leq 1 + \sqrt{|x|} \wedge y \geq 0.8 + \sqrt{|x|}$$

$$\#2: |x - 1| \leq 1 \wedge y \leq -2 + \sqrt{|x - 1|} \wedge y \geq -2.2 + \sqrt{|x - 1|}$$

$$\#3: |x - 2| \leq 1 \wedge y \leq \sqrt{|x - 2|} \wedge y \geq -0.2 + \sqrt{|x - 2|}$$

$$\#4: (x + 2)^2 + (y + 1)^2 \leq 4$$

$$\#5: -\frac{3}{2} \leq x \leq 5 \wedge y \leq -1 - |x - 4|$$



## 9. Funkce jedné a více proměnných

Pojem funkce se v Derive objevuje na dvou různých úrovních. Jednak jako reálná (komplexní) funkce jedné nebo více proměnných, jednak v obecnějším významu, jako akce na nějakých vstupních údajích, kterou si může uživatel nadefinovat sám.

### 9.1. Graf funkce

**PŘÍKLAD 9.1:** Sestrojte graf funkce  $f: y = \sqrt[3]{x}$

*Řešení:*

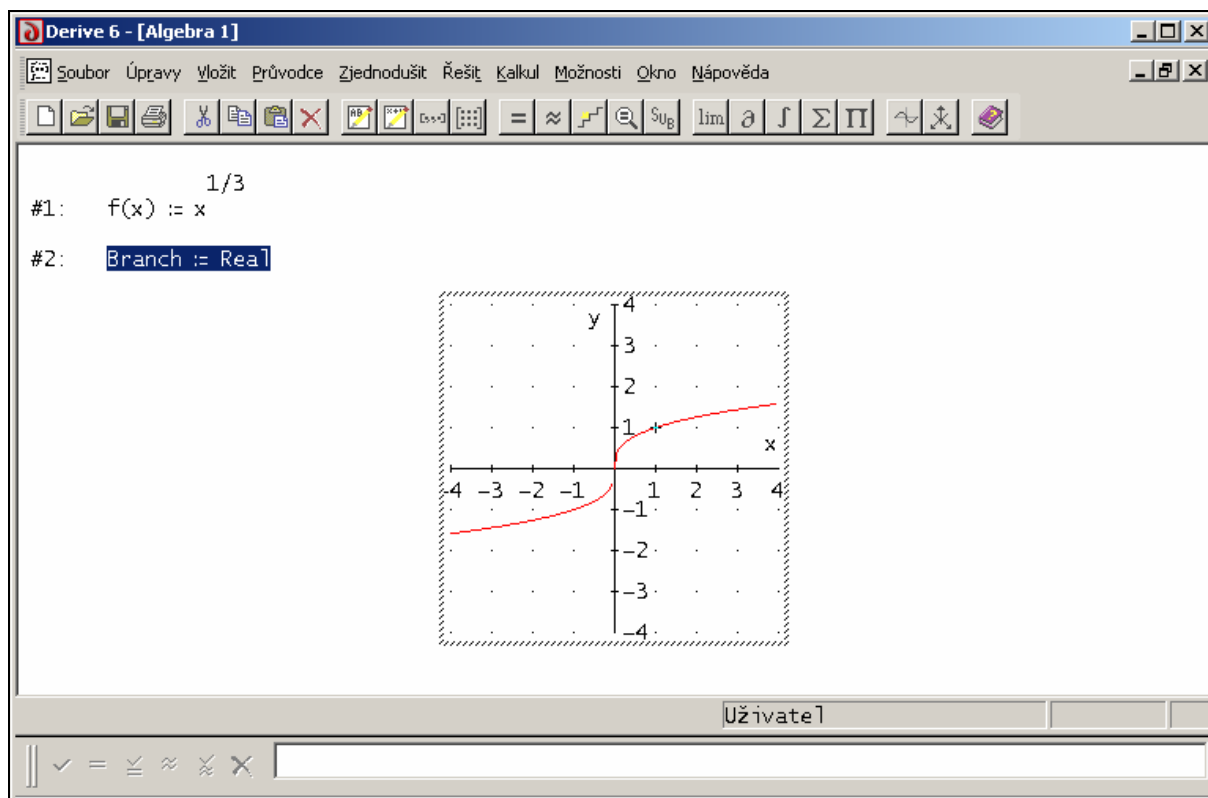
Pro vykreslení grafu funkce i pro záporné hodnoty  $x$  je třeba náležitě nastavit obor řešení odmocniny:

Možnosti → Nastavení → Komplexní větev → Real (Obr. 14)

Graf přeneseme do okna Algebra buď dříve uvedeným příkazem (při aktivním okně 2-D grafu) Soubor → Přemístit

nebo můžeme použít (při aktivním okně Algebra) příkaz

Vložit → 2D-graf...

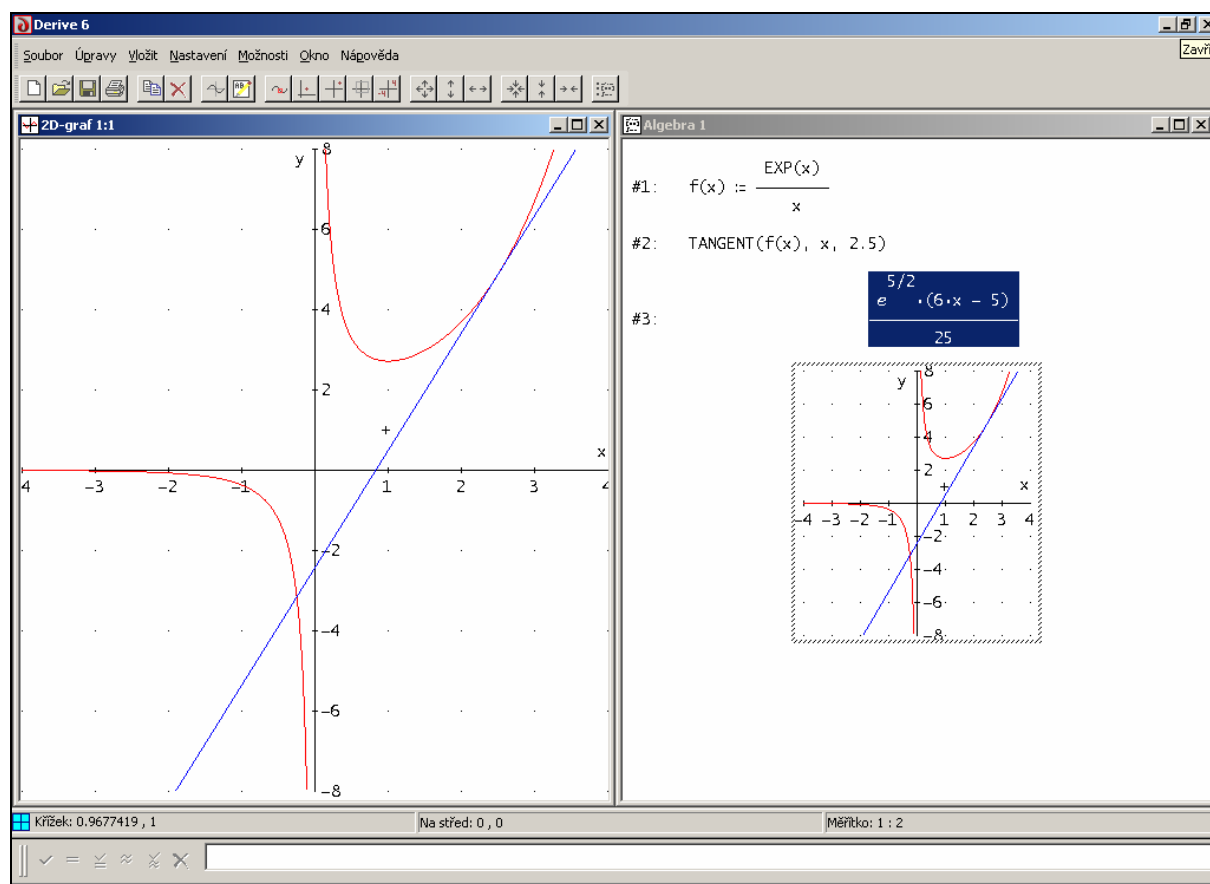


**PŘÍKLAD 9.2:** Načrtněte graf funkce  $f : y = \frac{e^x}{x}$ , určete rovnici tečny grafu v bodě  $x=2.5$

a tečnu sestrojte.

*Řešení:*

Použijeme příkaz **TANGENT(f(x),x,x0)**. Jeho výsledkem je rovnice tečny grafu funkce **f(x)** v bodě  $x = x_0$  (viz **Obr. 23**).



**Obr. 23**

V partii věnované racionální lomené funkci poznáme, že na příkazu **TANGENT** můžeme požadovat rovnici tečny grafu funkce i pro  $x$  jdoucí **do nekonečna**.

## 9.2. Graf funkce s parametrem. Posuvník.

Při zkoumání vlivu hodnoty nějakého parametru na graf funkce můžeme využít příkaz **VECTOR** a od verze 6 programu Derive také funkci **POSUVNÍK**.

### VECTOR

Tento příkaz nám dává možnost zobrazit do jednoho grafu více funkcí lišících se právě hodnotou parametru. Funkce můžeme zobrazit najednou, nebo i postupně, abychom lépe postřehli vliv parametru.

**PŘÍKLAD 9.3:** Vyzkoumejte, jak ovlivní hodnoty parametru  $b \in R$  průběh funkce

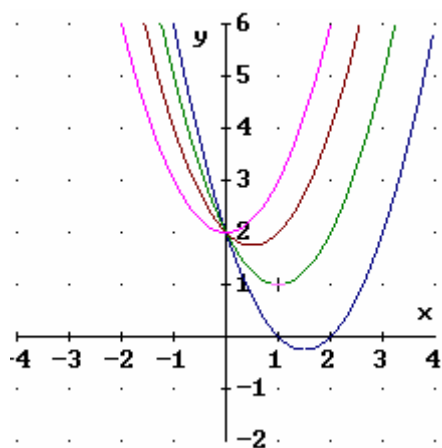
$$f : y = x^2 + bx + 2$$

*Řešení:*

Postup je patrný z následujícího výřezu okna Algebra:

- 1) zadáme funkci  $f(x)$
- 2) definujeme vektor funkcí lišících se hodnotou parametru
- 3) příkaz **VECTOR(...)** na ploše zvýrazníme a zjednodušíme
- 4) pokud necháme zvýrazněny všechny složky výsledku, zobrazí se všechny funkce najednou, ale pokud zvýrazníme jenom jednu jeho složku, zobrazí se jenom jedna jí odpovídající funkce.
- 5) provedeme zobrazení v okně 2-D graf obvyklým způsobem

```
#1:  f(x) := x2 + b·x + 2
#2:  VECTOR(f(x), b, -3, 0)
#3:  [ x2 - 3·x + 2, x2 - 2·x + 2, x2 - x + 2, x2 + 2 ]
```





## POSUVNÍK

Tato funkce nám umožňuje pouhým pohybem myši měnit hodnoty parametru daného výrazu a okamžitě zobrazovat grafy příslušných funkcí.

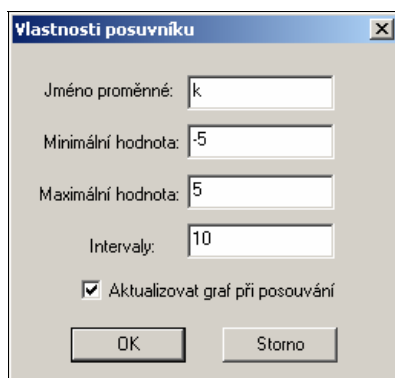
**PŘÍKLAD 9.4:** Vyzkoumejte, jak ovlivní hodnoty parametru  $k \in R$  průběh funkce

$$g : y = kx - 3$$

*Řešení:*

Při aktivním grafickém okně a zvýrazněném výrazu v okně Algebra volíme příkaz

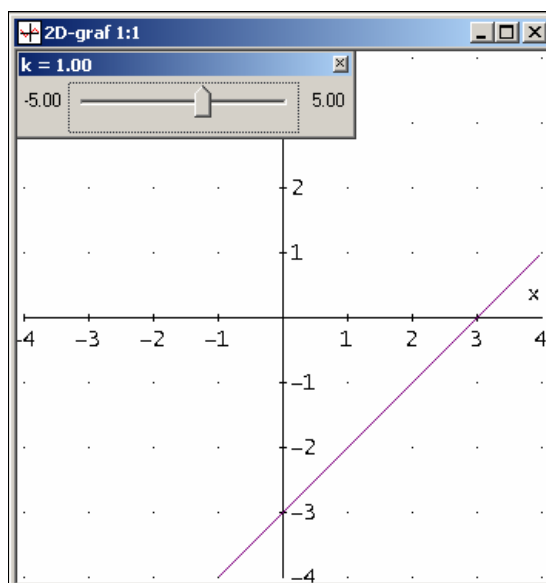
**Vložit** → **Posuvník...**. Objeví se dialogové okno (**Obr. 24**), v němž vyplníme potřebné údaje.



**Obr. 24**

Po potvrzení se objeví posuvník (**Obr. 25**).

Poté nakreslíme graf a sledujeme, jak se pohybem posuvníku mění jeho průběh v závislosti na odpovídající hodnotě parametru.



**Obr. 25**

### 9.3. Graf goniometrické funkce

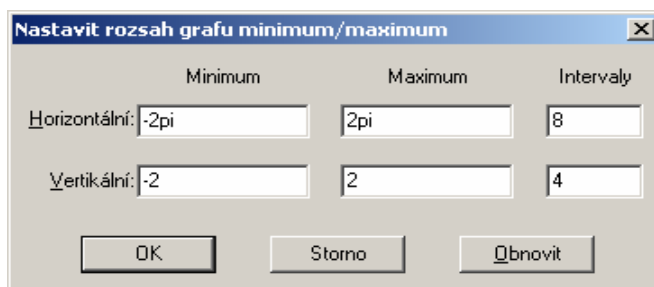
**PŘÍKLAD 9.5:** Vyšetřete vliv hodnot parametrů  $a$ ,  $b$  na průběh funkce dané předpisem

$$y = \sin(ax + b).$$

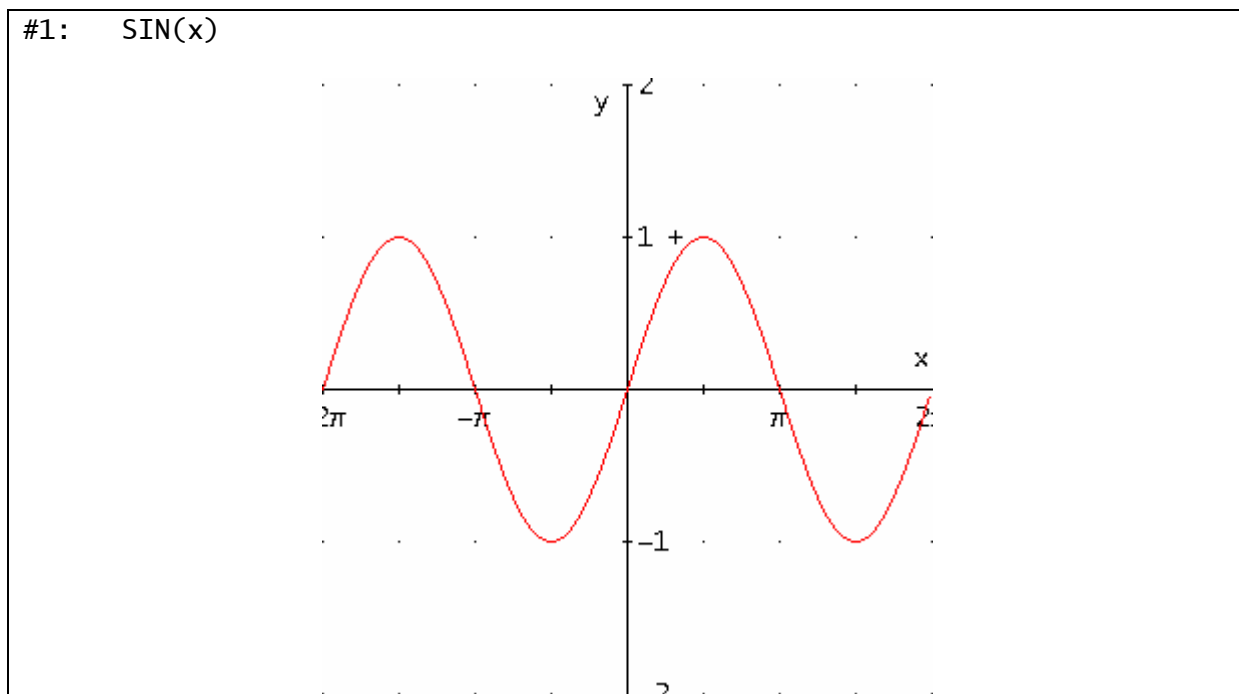
*Řešení:*

Nejprve si na funkci  $y = \sin x$  ukažme, jak získáme podobu osového kříže obvyklou pro znázorňování goniometrických funkcí, tj. na vodorovné ose jsou násobky  $\pi$ .

1) Při aktivním 2-D grafickém okně otevřeme dialogové okno příkazu **Nastavení** → **Rozsah grafu** → **Minimum/maximum...**, které vyplníme třeba takto:



2) Na kartě **Možnosti** → **Zobrazení...** → **Osy** vepíšeme do pole **Vodorovné násobky**: hodnotu  $\pi$  (napíšeme slovo „pi“ nebo použijeme panel mat. symbolů). Po zobrazení funkce  $\sin(x)$  dostaneme následující graf:



Vliv hodnot parametrů  $a$ ,  $b$  na průběh grafu funkce  $y = \sin(ax + b)$  nyní můžeme vyšetřit tak, že do 2-D grafického okna vložíme dva posuvníky, jeden pro parametr  $a$ , druhý pro  $b$ . Další postup je zřejmý.

## 9.4. Celá část

Derive má k dispozici funkci **celá část čísla**  $x$  ve tvaru **FLOOR(x)**.

**PŘÍKLAD 9.6:** V oboru reálných čísel řešte rovnici

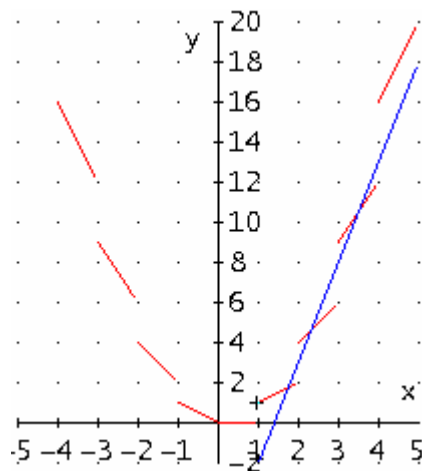
$$x[x] - 5x + 7 = 0.$$

*Řešení:*

Úlohu můžeme vyřešit nejprve graficky a potom na vybraných intervalech provést numerické řešení. Rovnici řešíme numericky tak, že při již zmiňované proceduře řešení rovnice zatrhneme v příslušném okně (**Obr. 19**) volbu **Metoda řešení** → **Numericky**. Poté jsme ještě v tomtéž okně dotázáni na meze intervalu, v němž řešení hledáme: **Hranice** → **Horní (Dolní)**. Příslušné meze odečteme z grafu. V tomto případě je patrné (viz níže, řádky #3, #5, #7), že rovnici řešíme numericky postupně v intervalech (1,2), (2,3) a (3,4).

#1:  $x \cdot \text{FLOOR}(x) - 5 \cdot x + 7 = 0$

#2:  $x \cdot \text{FLOOR}(x) = 5 \cdot x - 7$



#3:  $\text{NSOLVE}(x \cdot \text{FLOOR}(x) - 5 \cdot x + 7 = 0, x, 1, 2)$

#4:  $x = 1.75$

#5:  $\text{NSOLVE}(x \cdot \text{FLOOR}(x) - 5 \cdot x + 7 = 0, x, 2, 3)$

#6:  $x = 2.333333333$

#7:  $\text{NSOLVE}(x \cdot \text{FLOOR}(x) - 5 \cdot x + 7 = 0, x, 3, 4)$

#8:  $x = 3.5$

## 9.5. Racionální lomená funkce

**PŘÍKLAD 9.7:** Grafem funkce  $f$ , dané předpisem

$$f : y = \frac{2x - 1}{x + 2},$$

je rovnoosá hyperbola. Nakreslete tuto hyperbolu i s jejími asymptotami.

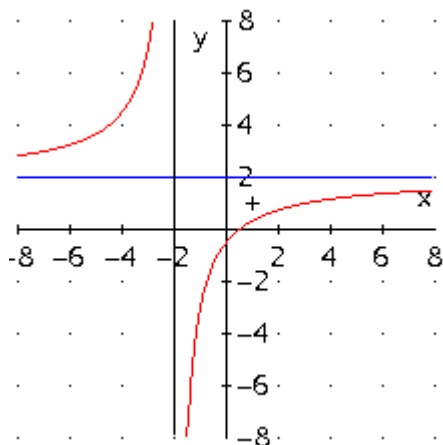
*Řešení:*

Pro sestavení grafu racionální lomené funkce i s jeho asymptotami si můžeme definovat jednoduchou funkci. Nazveme ji **ASYMPTOTY(f(x))** a jejím jediným parametrem bude předpis funkce **f(x)**. Hodnotou funkce **ASYMPTOTY** bude vektor jehož složkami jsou v uvedeném pořadí předpis funkce **f(x)**, rovnice vodorovné asymptoty a rovnice svislé asymptoty. Předpis funkce **ASYMPTOTY** je uveden níže v rámečku na řádku #1: ve stejné podobě, v jaké ho píšeme na příkazový řádek. Předpis **vodorovné asymptoty** získáme příkazem **TANGENT(f(x),x,∞)** jako předpis tečny grafu funkce **f(x)** v nekonečnu. Rovnici **svislé asymptoty** je určena řešením rovnice, v níž položíme roven nule jmenovatel výrazu definujícího funkci **f(x)**. Rovnici řešíme příkazem **SOLVE(l(x)=0,x)**. Jmenovatele zlomku **z** získáme příkazem **DENOMINATOR(z)**. Protože předpis funkce **f(x)** nemusí být v ryze lomeném tvaru, raději ho rovnou do tohoto tvaru, tj. na společného jmenovatele, převedeme příkazem **FACTOR(f(x),x,Trivial)**

#1: ASYMPTOTY(f) := [f, TANGENT(f, x, ∞), SOLVE(DENOMINATOR(FACTOR(f, x, Trivial)) = 0, x)]

#2: ASYMPTOTY $\left(\frac{2 \cdot x - 1}{x + 2}\right)$

#3:  $\left[\frac{2 \cdot x - 1}{x + 2}, 2, x = -2\right]$



## 10. Kombinatorika

K dispozici jsou nám mimo jiné tyto příkazy:

**PERM(m, n)** ... počet variací **n**-té třídy z **m** prvků

**PERM(n, n) = n!**.

**COMB(m, n)** ... počet kombinací **n**-té třídy z **m** prvků (kombinační číslo)

**RANDOM(n)** ... generátor náhodných čísel

**PŘÍKLAD 10.1:** Zobrazte prvních 5 řádků Pascalova trojúhelníku

*Řešení:*

**VECTOR(VECTOR(COMB(m, n), n, 0, 5), m, 0, 5)**

#1: VECTOR(VECTOR(COMB(m, n), n, 0, 5), m, 0, 5)

#2: 
$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 \\ 1 & 3 & 3 & 1 & 0 & 0 \\ 1 & 4 & 6 & 4 & 1 & 0 \\ 1 & 5 & 10 & 10 & 5 & 1 \end{bmatrix}$$

**PŘÍKLAD 10.2:** Řešte rovnici  $\frac{(n+2)!}{n!} = 2 \frac{n!}{(n-2)!} + 3!$

*Řešení:*

Řešíme přímo (#2, #3), nebo postupně upravujeme obě strany rovnice (#5, #6).

#1: 
$$\frac{(n+2)!}{n!} = \frac{2 \cdot n!}{(n-2)!} + 3!$$

#2: 
$$\text{SOLVE}\left(\frac{(n+2)!}{n!} = \frac{2 \cdot n!}{(n-2)!} + 3!, n, \text{Real}\right)$$

#3: 
$$n = 4 \vee n = 1$$

#4: 
$$\frac{(n+2)!}{n!} = \frac{2 \cdot n!}{(n-2)!} + 3!$$

#5: 
$$(n+1) \cdot (n+2) = 2 \cdot (n^2 - n + 3)$$

#6: 
$$n^2 + 3 \cdot n = 2 \cdot n^2 - 2 \cdot n + 4$$

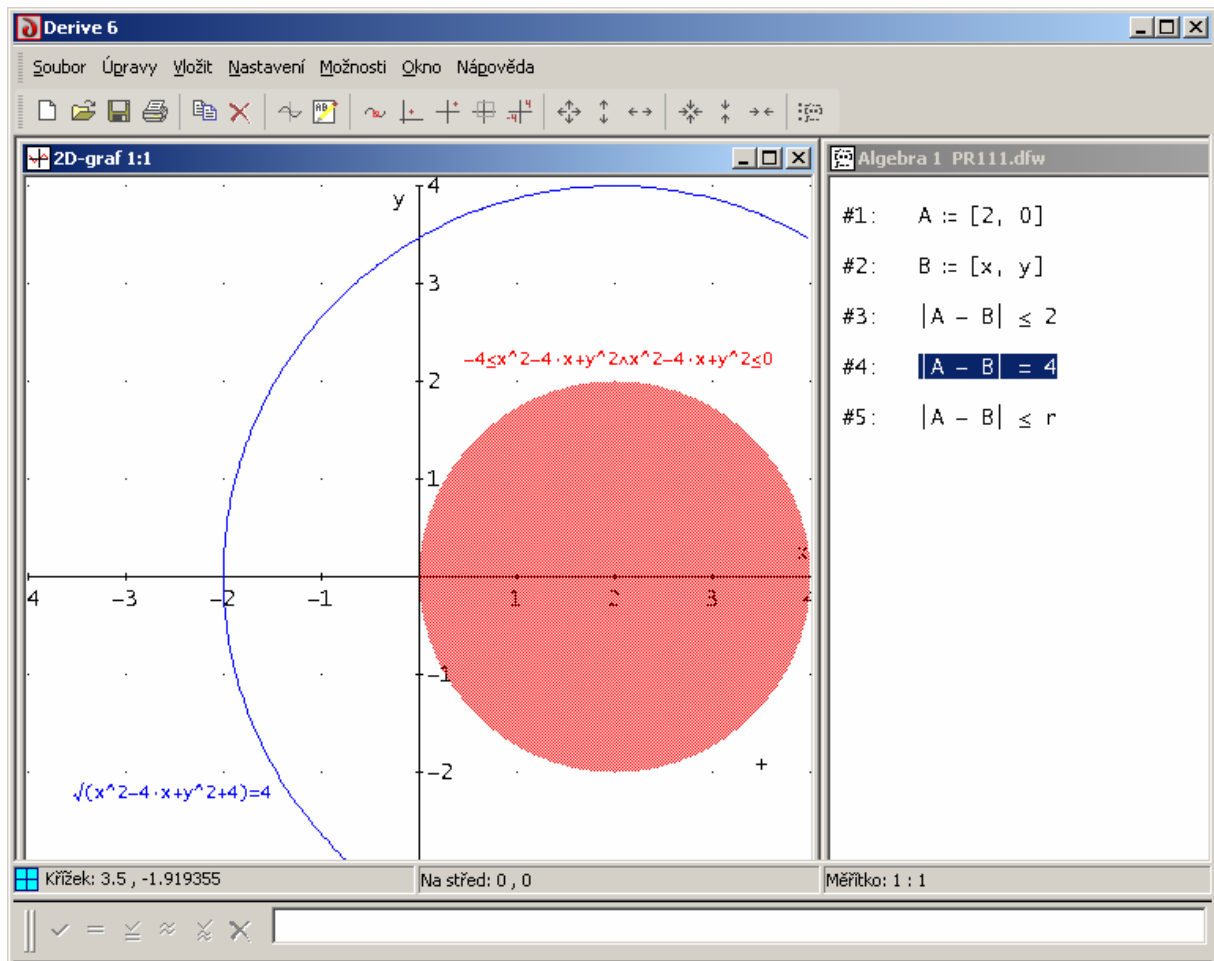
## 11. Analytická geometrie

### 11.1. Operace s vektory

**PŘÍKLAD 11.1:** Co je množinou všech bodů roviny, jejichž vzdálenost od bodu  $A=[2;0]$  je menší nebo rovna  $r$  ?

*Řešení:*

Vzdálenost dvou bodů  $A, B$  zapišeme pomocí funkce **ABS** jako **ABS(A-B)** (viz **Obr. 26**). Pro obecnější řešení úlohy můžeme použít parametr  $r$ , jehož hodnotu měníme posuvníkem. Potom je ale třeba počítat s větší časovou náročností úlohy. Zkrácení doby vykreslování grafů může pomoci, když pro potřeby programu uvolníme maximum paměti.



Obr. 26

Operace s vektory:

**DIM(v)** ... dimenze vektoru  $v$

**ABS(v)** ... velikost (absolutní hodnota) vektoru

**MEMBER?(u, v)** ... test, zda  $u$  je prvkem (členem)  $v$

+ ( - ) ... sčítání (odčítání) vektorů

\* ( / ) ... násobení (dělení) vektoru číslem

. (\*) ... skalární součin

CROSS(u,v) ... vektorový součin

ELEMENT(v,n) ... n-tý prvek vektoru v

APPEND(u,v) ... spojení vektorů u, v

SELECT(f(k),k,v) ... vybere z vektoru v ty členy k, pro které platí f(k)

## 11.2. Vzdálenost mimoběžek

**PŘÍKLAD 11.2:** Napište v Derive funkci pro výpočet vzdálenosti dvou mimoběžek, z nichž každá je určena bodem a směrovým vektorem.

*Řešení:*

Funkci nazveme VM, bude mít syntaxi VM(A,u,B,v) a pro její definici využijeme vzorec,

$$VM(A, u, B, v) = v = \frac{\left| \vec{AB} \cdot (\vec{u} \times \vec{v}) \right|}{|\vec{u} \times \vec{v}|}$$

kde v je vzdálenost dvou mimoběžek, z nichž jedna je dána bodem A a směrovým vektorem  $\vec{u}$ , zatímco druhá je dána bodem B a směrovým vektorem  $\vec{v}$ .

Než budeme funkci definovat, je dobré **deklarovat použité proměnné A, B, u, v jako vektory**: Průvodce → Definiční obor proměnné... →

*napišeme jméno proměnné a puntíkem vyznačíme* Definiční obor: Vektory → OK

Toto opakujeme pro všechny čtyři proměnné.

Pro zápis absolutní hodnoty reálného čísla (v čitateli zlomku) i normy vektoru (ve jmenovateli) použijeme funkci ABS().

Na příkazovém řádku definujeme funkci VM takto:

VM(A, u, B, v) := ABS((A - B)·CROSS(u, v))/ABS(CROSS(u, v))

#1: A :∈ Vector

#2: B :∈ Vector

#3: u :∈ Vector

#4: v :∈ Vector

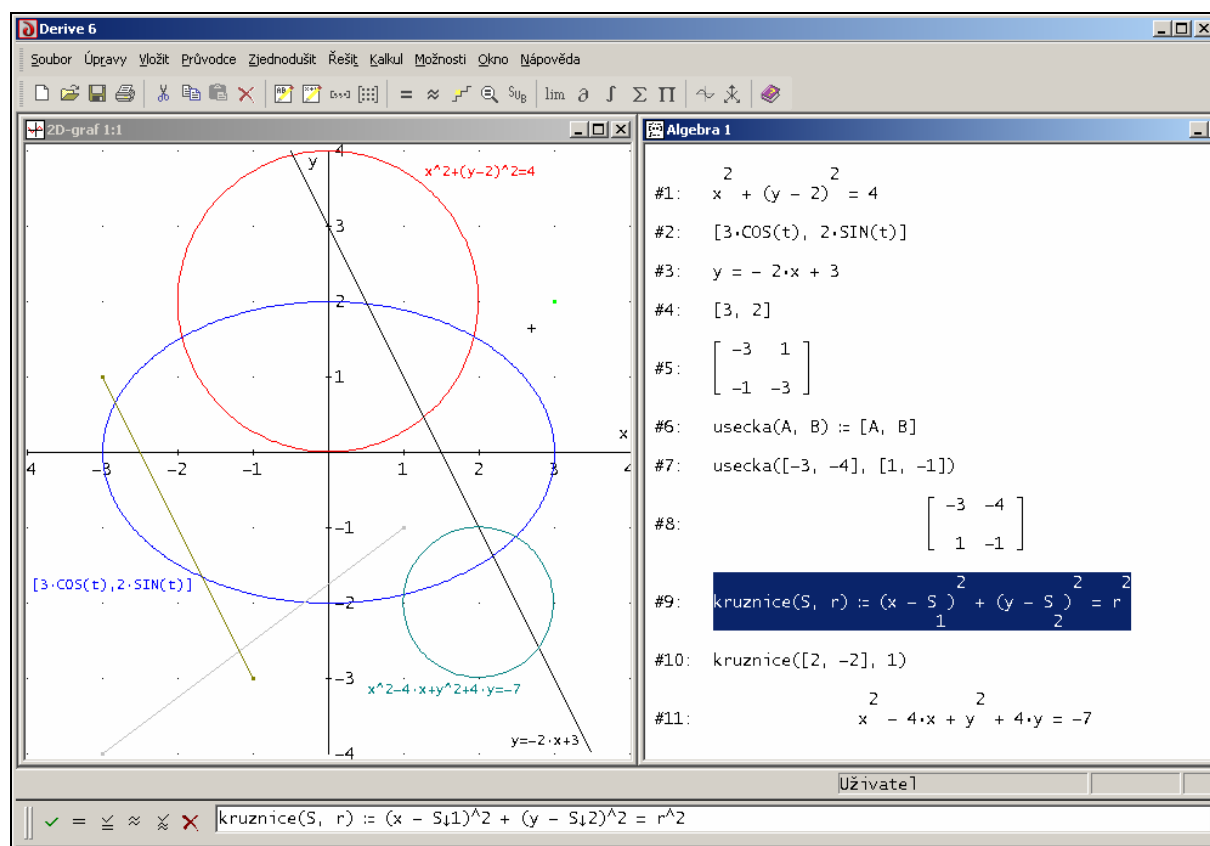
#5: VM(A, u, B, v) :=  $\frac{|(A - B) \cdot \text{CROSS}(u, v)|}{|\text{CROSS}(u, v)|}$

#6: VM([0, 0, 0], [1, 0, 0], [0, 5, 0], [0, 0, 1]) = 5

### 11.3. Kreslení geometrických útvarů

**Přímku, kružnici, křivku** nakreslíme prostě tak, že do okna Algebra napíšeme její rovnici (možno i parametrické vyjádření, viz **Obr. 27, #1, #2, #3**), tu zvýrazníme a zobrazíme v okně 2-D grafu. **Bod** zadáme jako vektor (**Obr. 27, #4**) a zase obvyklým způsobem zobrazíme. **Úsečku** zadáváme jako matici typu (2,2) s krajními body coby řádkovými vektory. Např. matici/úsečku na řádku **#5, Obr. 27** jsme na příkazový řádek zadali ve tvaru  $\begin{bmatrix} -3, 1 \\ -1, -3 \end{bmatrix}$ . Před vykreslením příslušné úsečky **nesmíme zapomenout** na volbu (při aktivním okně 2-D grafu): **Možnosti** → **Zobrazení...** → **Body** → **Spojovat** → **Ano**, jinak se nám zobrazí jenom krajní body úsečky. Pokud je grafický objekt spojen s rovnicí, můžeme ho v 2-D grafickém okně touto rovnicí popsat užitím volby: **Možnosti** → **Popsat nové grafy**

Analytická geometrie nám poskytuje široké pole působnosti při **tvorbě vlastních příkazů (funkcí)**. Na **Obr. 27** vidíme příklady definice příkazů **usecka(A,B)** a **kružnice(S,r)** pro snazší kreslení těchto útvarů. V definici funkce **kružnice** se objevují **dolní indexy** u proměnné S pro rozlišení první a druhé souřadnice středu. Jak je patrné z **Obr. 27**, k jejich zadání na příkazovém řádku použijeme šipku ↓ z panelu matematických symbolů. Abychom dostali rovnici kružnice, musíme příkaz po zadání na plochu obvyklým způsobem zjednodušit.



Obr. 27



## 11.4. Polární souřadnice

**PŘÍKLAD 11.3:** Křivka zvaná *strofoida* má v polárních souřadnicích rovnici

$$\rho = a \frac{\cos(2\varphi)}{\cos \varphi},$$

kde  $a$  je parametr křivky. Zobrazte tuto křivku.

*Pozn.:* Uvažujme svazek kružnic jejichž společnou tečnou je osa  $x$  a společným bodem dotyku je počátek. Zvolíme-li na ose  $x$  bod  $A[a,0]$  a vedeme-li jím průměry ke všem kružnicím svazku, potom krajní body těchto průměrů jsou body *strofoidy*.

*Řešení:*

1) Přes příkazový řádek vložíme uvedenou rovnici na pracovní plochu (okno Algebra). Místo parametru  $a$  volíme nějaké číslo (třeba 4) nebo  $a$  ponecháme a použijeme **posuvník** pro změnu jeho hodnot (viz PŘÍKLAD 9.4, str. 31).. Při vkládání řeckých písmen využíváme příslušné panely nástrojů.

2) Rovnici na pracovní ploše zvýrazníme.

3) 

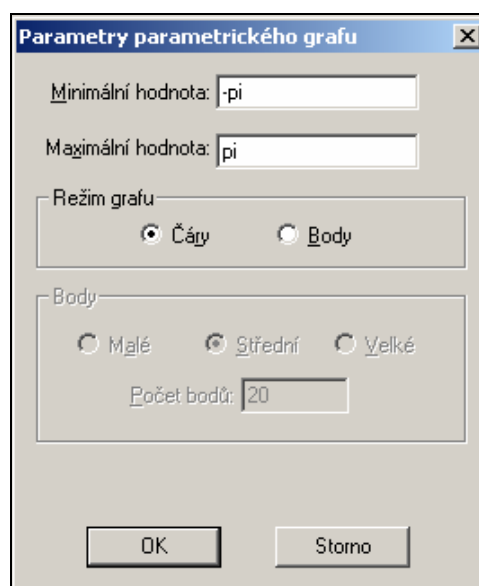
4) Při aktivním okně 2D-plot provedeme nastavení polárních souřadnic

**Nastavení** → **Souřadnicový systém...** → **Polární**

5) Aby křivka nebyla zobrazena zkresleně (s rozdílným vertikálním a horizontálním měřítkem), je dobré provést nastavení **Nastavení** → **Poměr stran...** → **Obnovit** → **OK**

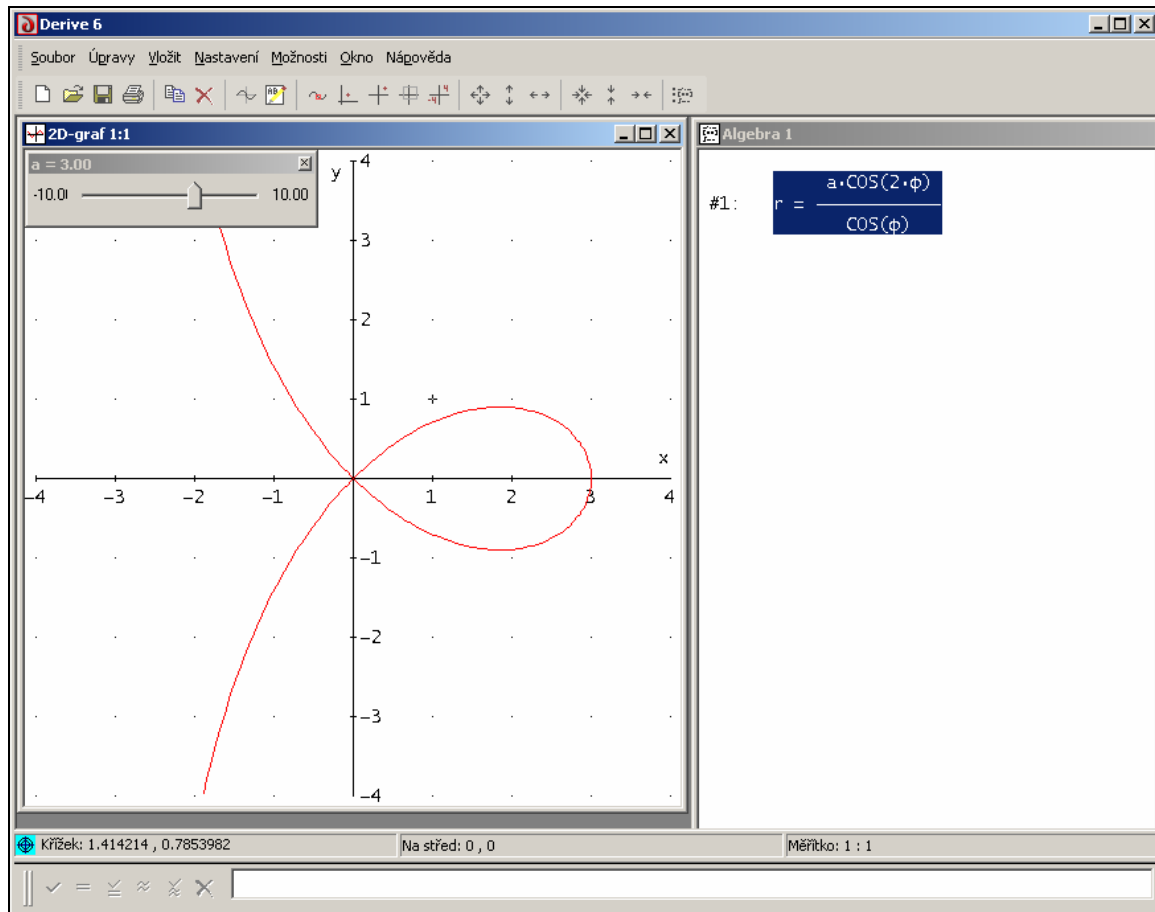
6) 

Objeví se okno na upřesnění parametrů grafu, viz **Obr. 28**.



**Obr. 28:** Parametry grafu

Meze parametru (úhlu) můžeme zadat číselně nebo jako násobky  $\pi$  (píšeme pi, viz **Obr.28**). V nabídce **Režim grafu** zvolíme, zda chceme křivku zobrazit spojitou (**Čáry**) čarou nebo bodově (**Body**). V případě druhé volby si ještě můžeme zvolit velikost a počet bodů. Po kliknutí na **OK** se již objeví graf.



**Obr. 29: Strofoida**

Při zápisu výrazu v polárních souřadnicích můžeme pro označení těchto souřadnic používat libovolnou dvojici symbolů. Tedy místo dvojice  $(\rho, \varphi)$  jsem klidně mohl použít dvojici  $(r, \alpha)$  apod.

**PŘÍKLAD 11.4:** Necht'  $P$  je pevný bod na kružnici  $K$ . Uvažujme množinu všech kruhů  $D_i$ , jejichž střed leží na  $K$  a hraniční kružnice prochází bodem  $P$ . Jaký je tvar plochy, kterou všechny tyto disky pokrývají? Jinak řečeno, jaká je jejich obálka.

*Řešení:*

Pro reprezentaci problému zvolíme **polární souřadnice**. Rovnice kružnice v obecné poloze má v polárních souřadnicích tvar

$$r^2 - 2rr_0 \cos(\alpha - \alpha_0) + r_0^2 - R^2 = 0, \quad (1)$$

kde  $(r_0, \alpha_0)$  jsou souřadnice středu kružnice a  $R$  je její poloměr (je to vlastně kosinová věta pro trojúhelník složený z uvedených prvků). Pokud pro kružnici  $K$  zvolíme polohu, ve které bod  $P$  splývá s pólem soustavy souřadnic a pro souřadnice středu platí  $r_0^K = R^K = \frac{1}{2}$ ,  $\alpha_0^K = \frac{\pi}{2}$ , její rovnice je oproti uvedenému obecnému tvaru značně jednodušší a vypadá takto

$$r^K = \sin \alpha^K.$$

To je ale, dle zadání, rovnice množiny poloh středů všech kruhů  $D_i$ , jejichž obálku hledáme. Protože jejich hranice procházejí bodem  $P$  (pólem), pro poloměr každého z nich platí  $R_i = r_{0_i} = r^K = \sin \alpha^K$ . Po dosazení do rovnice (1) dostaneme

$$r - 2 \sin \alpha^K \cos(\alpha - \alpha^K) = 0. \quad (2)$$

Po dosazení konkrétní hodnoty za  $\alpha^K$  do (2) tak dostaneme rovnici hraniční kružnice jednoho z uvažovaných kruhů  $D_i$ . Úlohu vyřešíme tak, že do jedné soustavy souřadnic zobrazíme dostatečné množství těchto kružnic.

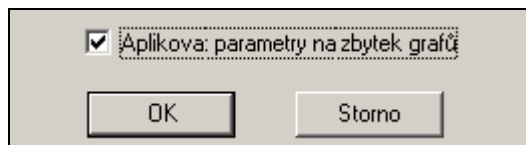
Využijeme příkaz **VECTOR(f(k),k,d,h,krok)** který je obdobou cyklu **FOR** známého z většiny programovacích jazyků. Příkaz postupně dosazuje do výrazu **f(k)** za proměnnou **k** hodnoty od **d** do **h** s krokem **krok** (parametr **krok** může být ve tvaru zlomku, může být i záporný a když ho neuvedeme, použije se implicitní hodnota 1). Výsledky dosazení (číselné hodnoty, výrazy apod.) se ukládají jako složky do vektoru, jehož dimenze je tedy rovna počtu dosazení. Pokud jsou složkami onoho vektoru předpisy funkcí (což je náš případ), můžeme všechny tyto funkce najednou zobrazit obvyklým způsobem pokud předtím celý vektor zvýrazníme. Jenom jednu složku (funkci) zobrazíme poté, co z celého vektoru zvýrazníme právě jenom ji.

1) Do okna Algebra vložíme příkaz

**VECTOR(r=2sin(k)cos(α-k), k, -π, π, 1/10)**

2) Příkaz provedeme (pokud je zvýrazněn) stisknutím **=** (Zjednodužit → Základní zjednodušení...)

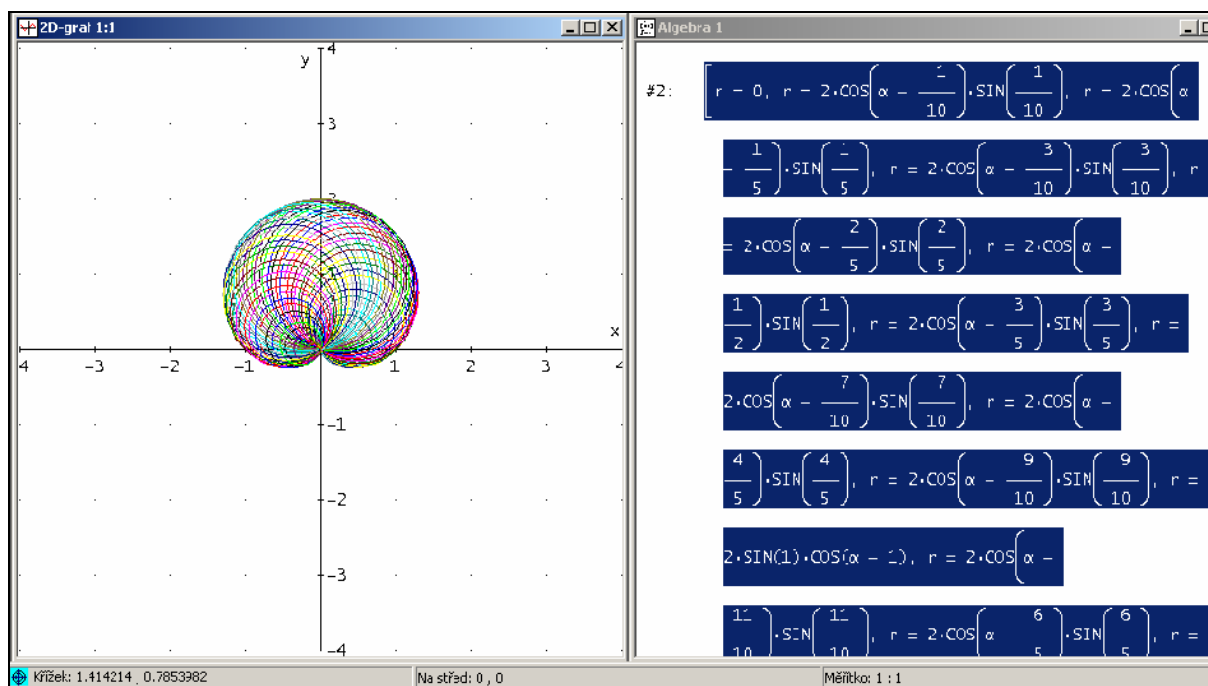
3) Na následujícím řádku se objeví vektor rovnic jednotlivých kružnic. Necháme ho zvýrazněný a zobrazíme ho v polárních souřadnicích postupem uvedeným v předchozím příkladě. Jediným rozdílem je snad podoba okna pro **zadání parametrů grafu (Obr. 28)**. Oproti situaci na **Obr. 28** se v něm navíc objeví volba **Aplikovat parametry na zbytek grafů** (viz **Obr. 30**),



**Obr. 30**

jejímž zatrhnutím zabráníme tomu, aby se toto okno zobrazilo pro každou kružnici zvlášť.

**Výsledek** vidíme na **Obr. 31**. Hraniční křivkou zkoumané plochy je křivka zvaná *kardioida* (srdcovka).



**Obr. 31: Kardioida**

## 12. Množinové operace

$M = \{a, b, c\}$

$M^c$  ... doplněk  $M$

$A \cup B$  ... sjednocení  $A, B$

$A \cap B$  ... průnik  $A, B$

$A \setminus B$  ... rozdíl  $A, B$

$\text{POWER\_SET}(A)$  ... všechny podmnožiny  $A$

$\text{POWER\_SET}(A, n)$  ... všechny podmnožiny mající  $n$  členů


$A * B$  ... kartézský součin množin  $A, B$



Množinové operace můžeme provádět na množinách s konečným počtem prvků. Zadávání množin lze zjednodušit třemi tečkami (#1, #3). Kromě toho program umožňuje symbolické úpravy množinových výrazů. Operaci doplněk lze použít právě jenom při těchto symbolických úpravách.

```
#1: M := {5, 6, ..., 10}
#2: M := {5, 6, 7, 8, 9, 10}
#3: N := {5, 7, ..., 20}
#4: N := {5, 7, 9, 11, 13, 15, 17, 19}
#5: M ∩ N = {5, 7, 9}
#6: M ∪ N = {5, 6, 7, 8, 9, 10, 11, 13, 15, 17, 19}
#7: M \ N = {6, 8, 10}
#8: POWER_SET(M ∩ N) = {{}, {5}, {5, 7}, {5, 7, 9}, {5, 9}, {7}, {7, 9}, {9}}
#9: a ∩ (b ∪ a) = a
#10: (a ∩ b)' = a' ∪ b'
#11: (a ∪ b)' = a' ∩ b'
#12: (a ∪ b) \ b = a ∩ b'
#13: (a ∩ c) ∪ (b ∩ c) = (a ∪ b) ∩ c
#14: a \ {} = a
```

Obr. 32: Množinové operace

### 13. 3D grafy

Pro tvorbu 3D grafu platí stejný postup, jaký jsme uplatňovali v případě 2D grafu. Akorát použijeme ikonu , vyhrazenou pro 3D graf:

- 1) výraz zvýrazníme,
- 2) klikneme na , objeví se okno 3D grafu,
- 3) klikneme ještě jednou na , nakreslí se graf,
- 4) pro větší názornost můžeme rozdělit pracovní plochu na dvě části, jednu pro okno 3D grafu, druhé pro okno Algebra : **Okno** → **Vertikální dlaždice**

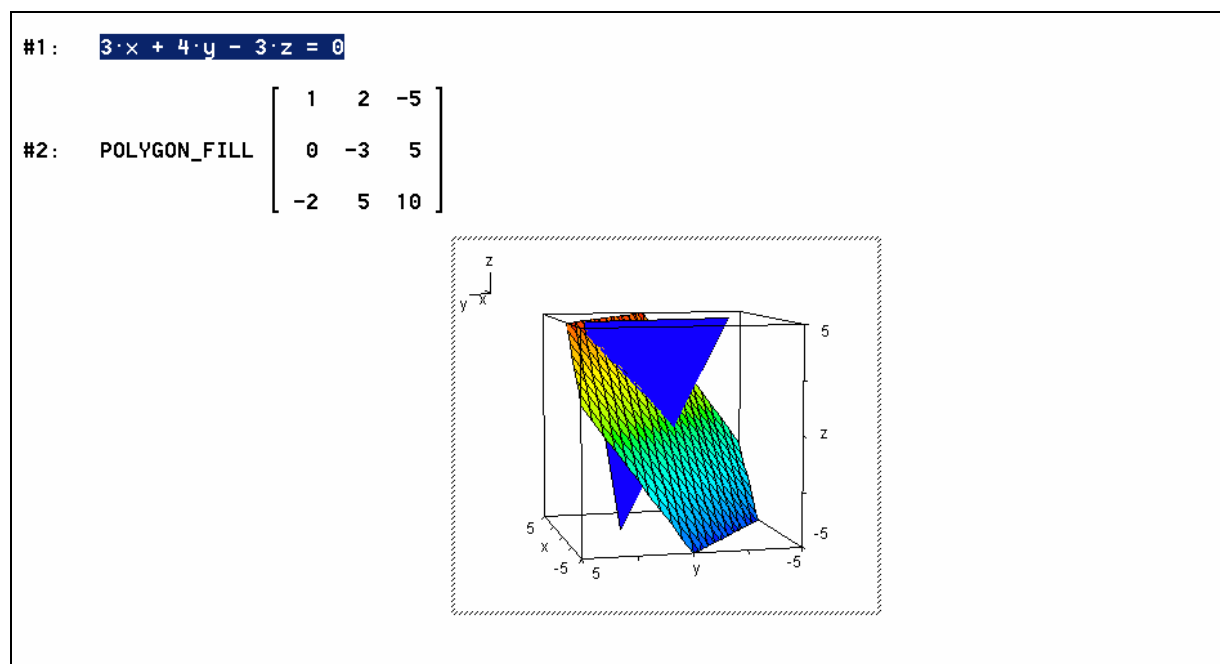
Tak jako v rovině můžeme i v 3D okně zobrazit grafy dané explicitně i parametricky a můžeme si rovněž vybrat souřadnicový systém (pravoúhlé, sférické nebo cylindrické souřadnice). Zobrazení implicitně zadaných prostorových útvarů nelze zaručit, ale třeba **rovinu zadanou obecnou rovnicí** program vykreslí.

**PŘÍKLAD 13.1:** Zobrazte rovinu danou obecnou rovnicí  $3x + 4y - 3z = 0$  spolu s trojúhelníkem, který je zadán vrcholy A[1; 2; -5], B[0; -3; 5], C[-2; 5; 10].

*Řešení:*

Obecnou rovnicí roviny napíšeme a po zvýraznění zobrazíme výše uvedeným způsobem.

K zobrazení trojúhelníka (mnohoúhelníka) použijeme příkaz **POLYGON\_PLOT(M)**, kde **M** je matice vrcholů trojúhelníka (mnohoúhelníka) (viz. **Obr. 33**).



**Obr. 33:** Průnik roviny s trojúhelníkem

### 13.1. Rotační plochy

**PŘÍKLAD 13.2:** Je dána funkce  $f: y = x^2 - 4$ . Sestrojte rotační plochu, která vznikne otáčením grafu této funkce kolem osy  $x$  ( $y$ ).

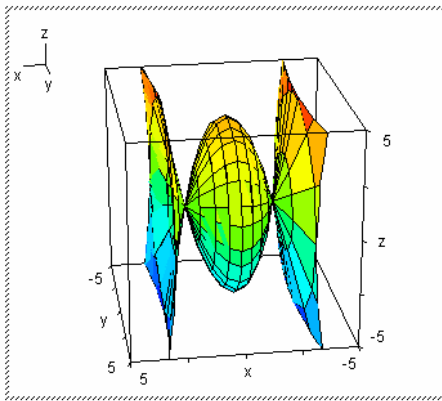
*Řešení:*

Nejprve definujeme funkci  $f(x)$ . Potom si představíme graf této funkce přenesený do 3D okna a zapíšeme parametrická vyjádření příslušných ploch.

#1:  $f(x) := x^2 - 4$

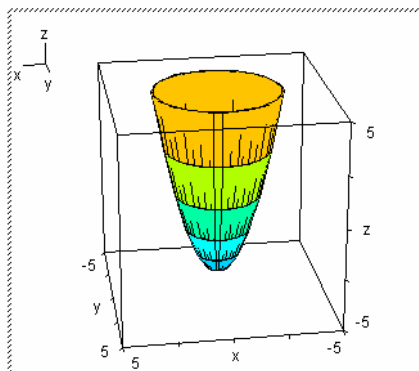
Rotaci kolem osy  $x$  odpovídá po přenesení do 3D okna opět rotace kolem osy  $x$ . Plocha závisí na dvou parametrech, symboly pro jejich označení (zde  $u, v$ ) nejsou předepsány. Parametrické rovnice pro jednotlivé souřadnice napíšeme do vektoru, ten zvýrazníme a jeho zobrazením do 3D okna dostaneme očekávanou plochu.

#2:  $[u, f(u) \cdot \cos(v), f(u) \cdot \sin(v)]$



Rotaci kolem osy  $y$  po přenesení do 3D okna odpovídá rotace kolem osy  $z$ .

#3:  $[u \cdot \cos(v), u \cdot \sin(v), f(u)]$




## 14. Proložení daných bodů grafem funkce

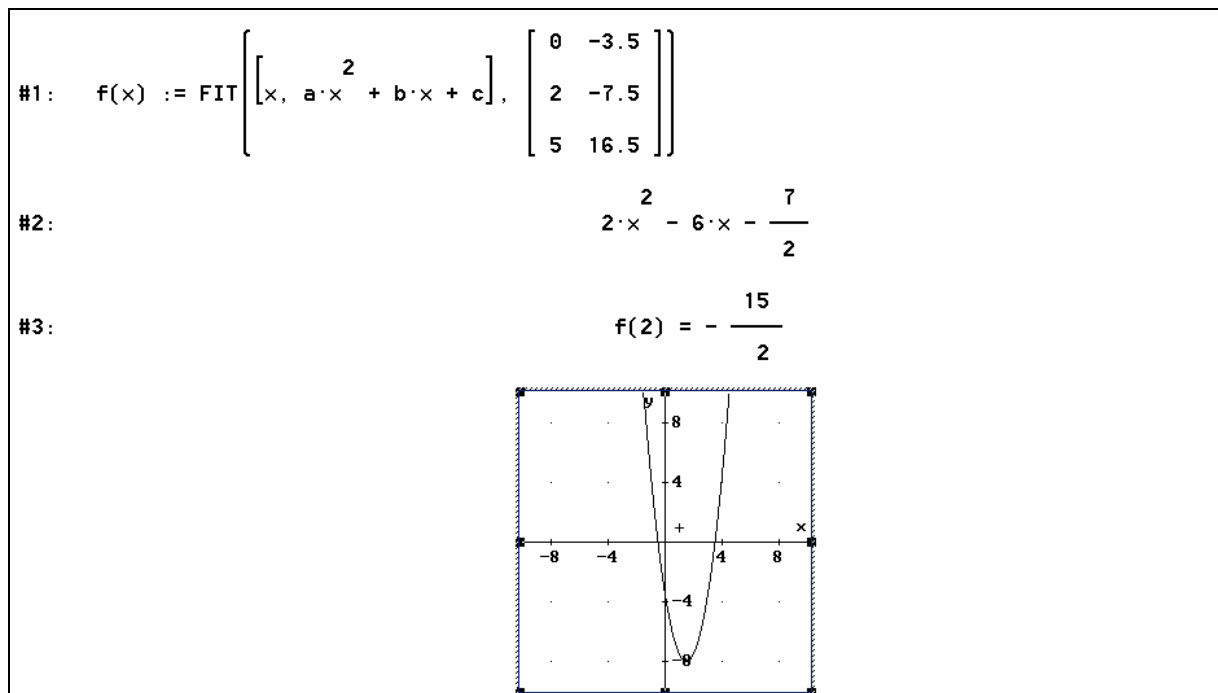
Proložit funkci zvoleného typu danými body v rovině i v prostoru nám umožňuje příkaz  $\text{FIT}([x,f(x)],M)$ , resp.  $\text{FIT}([x,y,f(x,y)],M)$ , kde  $[x,f(x)]$ , resp.  $[x,y,f(x,y)]$  je vektor, jehož složkami jsou postupně nejprve proměnné a nakonec obecné zadání předpisu funkce, parametrem  $M$  je matice s řádky tvořenými souřadnicemi daných bodů (viz příklady). Použití příkazu si ukážeme na příkladu hledání předpisu kvadratické funkce určené třemi body v rovině a na příkladu zobrazení roviny v 3D okně. Příkaz je založen na metodě nejmenších čtverců a možnosti jeho využití jsou samozřejmě širší. Zájemce o podrobnější informace odkazují na nápovědu.

**PŘÍKLAD 14.1:** Určete rovnici kvadratické funkce  $f$ , jejíž graf prochází body  $A[0; -3.5]$ ,  $B[2; -7.5]$ ,  $C[5; 16.5]$ .

*Řešení:*

Zadání příkazu viz **Obr. 34**.

Příkaz provedeme (pokud je zvýrazněn) stisknutím  (Zjednodušit → Základní zjednodušení...). Získanou kvadratickou funkci potom běžným způsobem zobrazíme. Pokud provedeme přiřazovací příkaz  $f(x):= \dots$  jako na řádku #1, můžeme se na funkci v dalších výpočtech odkazovat pomocí tohoto jména (viz ř.#3).



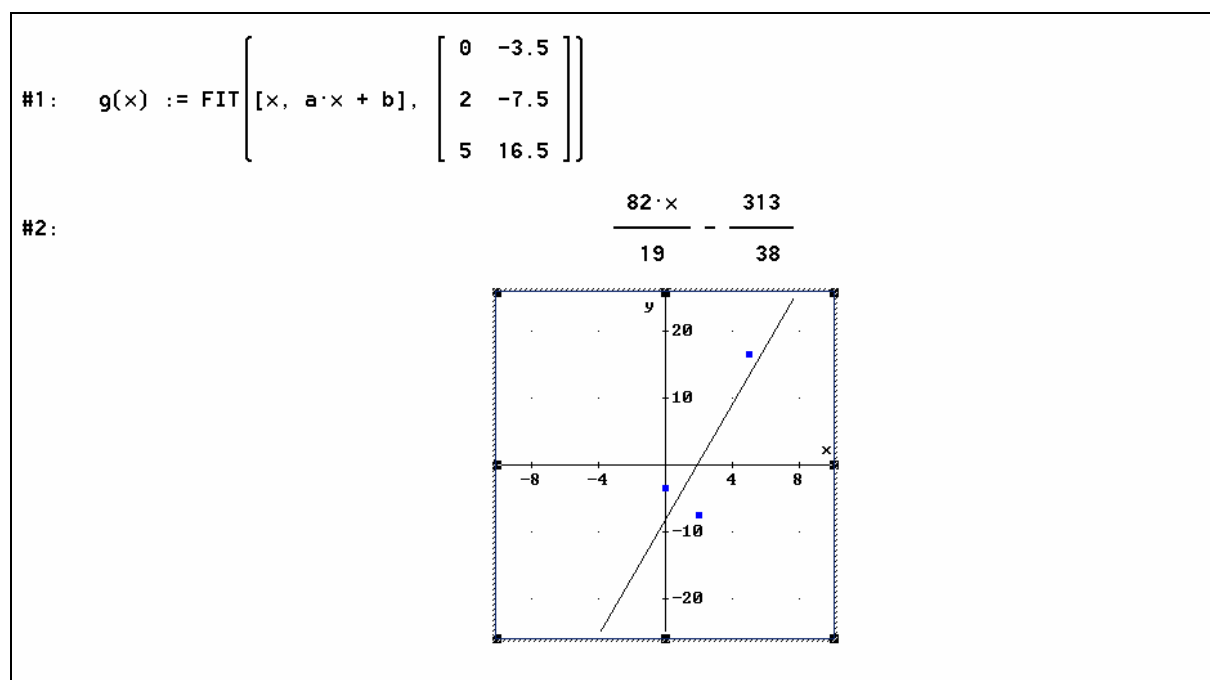
Obr. 34: Kvadratická funkce



Danými body můžeme nechat proložit i funkcí nižšího řádu, v tomto případě přímkou (viz **Obr. 35**). Samotné body zobrazíme jako každý jiný výraz poté, co zvýrazníme matici, v níž jsou zapsány jejich souřadnice.

Velikost bodů nastavíme při aktivním grafickém okně posloupností příkazů

Možnosti → Zobrazení → Body → Velikost (Malá, Střední, Velká).



**Obr. 35: Přímka proložená třemi body**

**PŘÍKLAD 14.2:** Rovina je určena body  $M[-1; 2; 5]$ ,  $N[0; 3; 9]$ ,  $P[5;-8; 1]$ . Napište její rovnici a zobrazte ji v 3D okně.

*Řešení:*

Řešení viz **Obr. 36**.

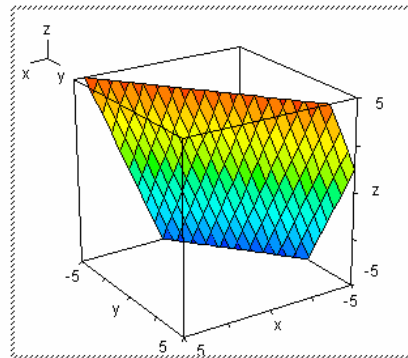
Rovinu vyjadřuji pomocí předpisu odpovídající funkce dvou proměnných  $z = f(x,y) = ax+by+c$ . Mohu si představit, že ji získám z obecné rovnice roviny  $Ax + By + Cz + D = 0$  vyjádřením  $z$ .

#1:  $M := \begin{bmatrix} -1 & 2 & 5 \\ 0 & 3 & 9 \\ 5 & -8 & 1 \end{bmatrix}$

#2:  $\text{FIT}([x, y, a \cdot x + b \cdot y + c], M)$

#3:

$$\frac{9 \cdot x}{4} + \frac{7 \cdot y}{4} + \frac{15}{4}$$



Obr. 36: Rovina

## 15. Fraktály

Běžné nastavení kreslení 2-D grafů, v němž se pro každý následující graf použije jiná barva, spolu s příkazy **ITERATE** a **VECTOR** nám dovoluje poměrně jednoduše zobrazit některé fraktální jevy v Gaussově rovině komplexních čísel. Samozřejmě, pokud nechceme v Derive programovat, nemůžeme čekat zázraky a velice rychle narazíme na meze těchto možností. Uvedené příklady rozhodně nijak nezastíní speciální programy, které najdeme na internetu nebo si je napíšeme sami. Můžeme je však brát jako impuls pro pochopení a případné další zkoumání krásného světa fraktálů.

### ITERATE, ITERATES

Výstupem příkazu (funkce) **ITERATE(f(x),x,p,n)** je výsledek **n** opakování přiřazení **f(x) → x** (tj. do výrazu **f(x)** dosazují za **x** opět výraz **f(x)**) s počáteční hodnotou **x = p**.

Posloupnost všech **n+1** postupných výsledků (**i** s počáteční hodnotou) tohoto přiřazení (tedy iterací funkce **f(x)**) dostaneme příkazem **ITERATES(f(x),x,p,n)**.

### Několik jednoduchých příkladů použití příkazů ITERATE, ITERATES :

#1:	<code>ITERATE(1 + x, x, a, 5)</code>	
#2:		$a + 5$
#3:	<code>ITERATES(1 + x, x, a, 5)</code>	
#4:		$[a, a + 1, a + 2, a + 3, a + 4, a + 5]$
#5:	<code>ITERATES(<math>\frac{1}{x}</math>, x, a, 5)</code>	
#6:		$[a, \frac{1}{a}, a, \frac{1}{a}, a, \frac{1}{a}]$
#7:	<code>ITERATES(x·i, x, i, 5)</code>	
#8:		$[i, -1, -i, 1, i, -1]$

### VECTOR

Příkaz **VECTOR(f(k),k,d,h,krok)** je obdobou cyklu **FOR** známého z většiny programovacích jazyků. Příkaz postupně dosazuje do výrazu **f(k)** za proměnnou **k** hodnoty od **d** do **h** s krokem **krok** (parametr **krok** může být ve tvaru zlomku, může být i záporný a když

ho neuvedeme, použije se implicitní hodnota 1). Výsledky dosazení (číselné hodnoty, výrazy apod.) se ukládají jako složky do vektoru, jehož dimenze je tedy rovna počtu dosazení.

### Několik jednoduchých příkladů použití příkazu VECTOR:

```
#1: VECTOR  $\left( \frac{1}{2}, n, 1, 10, 2 \right)$ 
#2:  $\left[ 1, \frac{1}{9}, \frac{1}{25}, \frac{1}{49}, \frac{1}{81} \right]$ 
#3:  $f(x) := x + \frac{a}{x-1}$ 
#4: VECTOR(f(x), a, 1, 5)
#5:  $\left[ \frac{1}{x-1} + x, \frac{2}{x-1} + x, \frac{3}{x-1} + x, \frac{4}{x-1} + x, \frac{5}{x-1} + x \right]$ 
```

**Poznámka:** Možnosti použití uvedených příkazů jsou širší. Potřebné informace a příklady najdeme v nápovědě ( [Nápověda](#) → [Témata nápovědy](#) → [Index](#) → napíšeme hledaný příkaz).

### Fraktální jevy v rovině komplexních čísel

Jevy popíší jen na úrovni uvedených příkladů. Uvažujme funkci komplexní proměnné  $f(z)$ . Zvolme nějaký bod  $z = x + yi$  komplexní roviny a provádějme iterace funkce  $f(z)$  s počáteční hodnotou v tomto bodě. Zajímá nás jak rychle se vzdalují hodnoty těchto iterací od onoho výchozího bodu. Zvolme určitý počet iterací  $n$  (vzhledem k možnostem programu bude poměrně malý) a rozdělme body komplexní roviny (přesněji, nějaké její části) do skupin (lišících se obarvením) podle toho, jak daleko se "dostanou" po tomto počtu iterací od své výchozí polohy. Výsledkem budou fraktální obrazce v Gaussově rovině. Pro iterace funkce  $f(z) = z^2 + c$ , kde  $c$  je komplexní parametr, dostáváme tzv. **Juliovy množiny**. Pro konkrétní Juliovu množinu je tak charakteristická určitá hodnota parametru  $c$ , zatímco výchozí hodnota  $z = x + yi$  probíhá nějakou oblastí komplexní roviny. Pokud výchozí bod iterací necháme pevný, třeba  $z = 0$ , a určitou oblastí komplexní roviny necháme probíhat parametr naopak  $c$ , dostaneme tzv. **Mandelbrotovu množinu**. I jiné funkce komplexní proměnné mohou přinést pozoruhodné výsledky. Zajímavé je například zkoumání iterací

funkce  $f(z) = \frac{2z^3 + 1}{3z^2}$ . To je funkce, jejímiž iteracemi získáme řešení rovnice  $z^3 - 1 = 0$  tzv.

**Newtonovou iterační metodou**. Je zajímavé sledovat, s jakou ochotou se body roviny přibližují k některému z kořenů této rovnice. Tato "ochota" je vyjádřena hodnotou derivace iterační funkce  $f'(x)$ . Spádové oblasti Gaussovy roviny příslušející jednotlivým kořenům

rovnice a navíc rozdělené podle rychlosti konvergence tak dostaneme zobrazením oblasti roviny komplexních čísel, lišících se hodnotou této derivace.

**PŘÍKLAD 13.1: Juliova množina.**

Zobrazte Juliovu množinu pro  $c = 0.2 + 0.4i$  a oblast komplexní roviny se středem v počátku a poloměrem 2.

*Řešení:* (viz. Obr. 32)

1) Deklarujeme komplexní proměnné  $z, k$

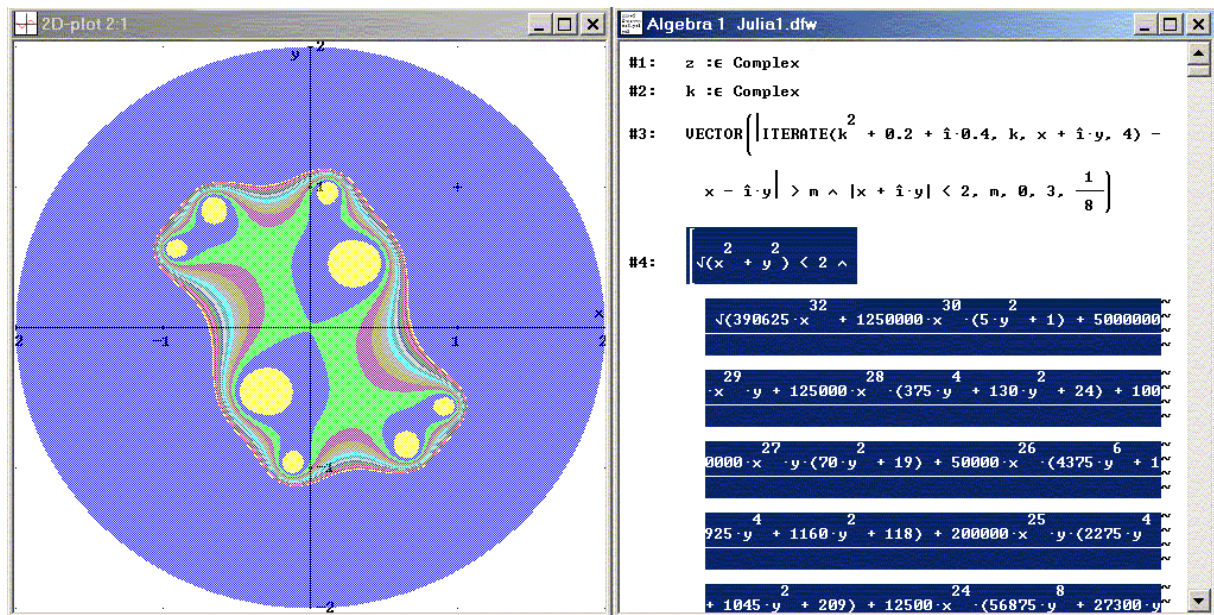
2) Zadáme příkaz

`VECTOR(ABS(ITERATE(k^2+0.2+0.4i, k, x+yi, 4)-x-yi)>m AND ABS(x+yi)<2, m, 0, 3, 1/8)`

3) Příkaz zjednodušíme (můžeme volit i numerický výpočet)

Výsledkem je vektor. Složkami vektoru jsou předpisy relací definujících vždy ty body uvedené oblasti komplexní roviny, které se po čtyřech iteracích funkce  $f(z) = z^2 + c$  dostaly od své výchozí polohy dále než je hodnota  $m$ .

4) Složky vektoru necháme zvýrazněny a obvyklým způsobem zobrazíme v okně 2-D grafu.



Obr. 32: Juliova množina

### PŘÍKLAD 13.2: Newtonova iterační metoda

Zobrazte oblasti přitažlivosti jednotlivých kořenů v Gaussově rovině komplexních čísel při řešení rovnice  $z^3 - 1 = 0$  Newtonovou iterační metodou.

Řešení: (viz Obr. 33)

Posloupnost řešení rovnice je dána rekurentním vztahem  $x_{n+1} = x_n - \frac{g(x_n)}{g'(x_n)}$ , kde

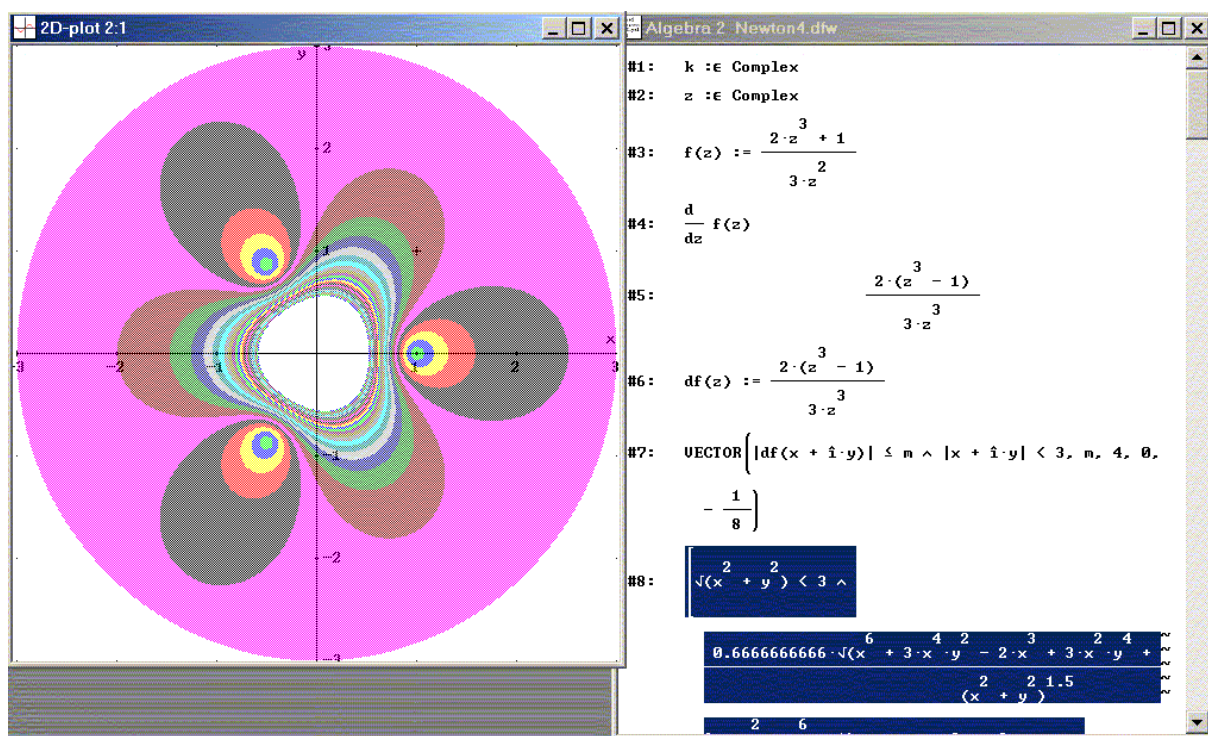
$g(x) = x^3 - 1$ . Ten můžeme přepsat do tvaru  $x_{n+1} = f(x_n) = \frac{2x_n^3 + 1}{3x_n^2}$ . Jak je uvedeno výše,

zajímá nás **derivace** této funkce, kterou získáme následujícím postupem (viz Obr. 33):

1) Na řádce #3 zvýrazníme jméno funkce **f(z)** (můžeme ale klidně zvýraznit celý přiřazovací příkaz, nebo jenom jeho pravou stranu).

2) **Kalkul** → **Derivace...** → Proměnná, Řád

3) Derivaci funkce **f(x)** definujeme jako funkci **df(x)** (viz #6), kterou následně použijeme v příkazu VECTOR (#7). V tomto příkazu porovnáváme hodnotu  $|f'(z)|$  s postupně se měnícím parametrem **m**. Přitom se omezujeme na hodnoty  $|z| < 3$ . Výsledkem jsou předpisy relací proměnných  $x, y$ , jimž na obrázku odpovídají oblasti vybarvené stejnou barvou (ne tak docela, když je oblastí více než barev, posloupnost barev se začne opakovat).



Obr. 33: Newtonova iterační metoda,  $z^3 - 1 = 0$