

# **Derive 6.1, česká verze**

## **ŘEŠENÍ VYBRANÝCH ÚLOH Z MATEMATIKY**

### **Úlohy pro ZŠ**

**Roman Hašek**

**Jihočeská Univerzita v Č. Budějovicích**

**2004**

*„Existuje tisíce cest jak používat CAS (systémy počítačové algebry) pro vyučování, dobré i špatné. Špatné nebo, lépe, nevhodné přístupy často přicházejí od technických nadšenců z řad učitelů, kteří používají CAS už proto, že existují a že je možné je používat konkrétní cestou. Ale CAS by nikdy neměly řídit matematiku, kterou učíme, naše matematika (cíle vyučování) by měla řídit (a používat) CAS! Plně souhlasím s Helmutem Heuglem, ředitelem rakouského projektu Derive a TI-89/92 (které zahrnují téměř 6 000 studentů), který řekl: „Jestliže použití CAS není ospravedlněno pedagogicky, je pedagogicky ospravedlněno CAS nepoužívat.“ Tedy: CAS nás také nutí přemýšlet o tom JAK učit.“*

Bernhard Kutzler, ACDCA (Rakouské centrum pro didaktiku počítačové algebry),  
přednáška na konferenci „Užití počítačů ve výuce matematiky“,  
Č. Budějovice, 6. – 8. 11. 2003

## Úvod

Cílem tohoto materiálu je být impulsem k hledání odpovědí na otázky naznačené úvodním citátem, tj. zda je použití programů CAS při výuce matematiky na základních školách pedagogicky ospravedlněno, a jak je v takovém případě používat. Materiál obsahuje několik úloh řešených pomocí programu Derive, na nichž jsem se pokusil ukázat některé zajímavé schopnosti tohoto programu, které by mohly být přínosné i při hodinách matematiky na základní škole. Třeba tak, že umožní prožít žákovi svůj vlastní objevitelský pocit i když zrovna není matematickým géniem.

Úloh zatím není mnoho, doufám, že budou časem přibývat. Budu rád za jakékoliv podněty a připomínky ([hasek@pf.jcu.cz](mailto:hasek@pf.jcu.cz)). Úlohy jsou řešeny v české lokalizaci verze 6.1 programu Derive. Jedná se o aktuální verzi, která je rozšířením Derive 5. Materiál navazuje na již dříve publikovaný **“Derive 6.1, Řešení vybraných úloh z matematiky“**. Ten se podrobně věnuje řadě důležitých funkcí programu. 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.

# 1. Funkce

V programu Derive 6 je oproti starší verzi Derive 5 nově poskytnuta mimo jiné i funkce posuvníku (Slider bar). Tuto novou funkci programu oceníme při zkoumání závislosti průběhu grafu funkce na hodnotách nějakého parametru (třeba koeficientu u  $x$  v případě lineární funkce). Umožňuje nám pouhým pohybem myši měnit hodnoty tohoto parametru a okamžitě zobrazovat grafy příslušných funkcí.

## 1.1 Lineární funkce a její vlastnosti


**PŘÍKLAD 1.1:** Lineární funkce je každá funkce daná rovnicí  $y = ax + b$ , kde  $a, b$  jsou libovolná reálná čísla a definičním oborem je množina všech reálných čísel. Vyzkoumejte, jaký vliv mají hodnoty čísel  $a, b$  na podobu a průběh grafu lineární funkce.


*Řešení:*

Budeme zkoumat vliv každého z čísel  $a, b$  zvlášť. Tj. budeme pracovat s dílčími funkcemi, vždy s jedním parametrem. Například  $f_1: y = ax + 1$  s parametrem  $a$  a  $f_2: y = 2x + b$  s parametrem  $b$ .

*Připomeňme si, jak dostaneme **grafické znázornění (2D-graf)** nějakého výrazu (výsledek rovnice či nerovnice, předpis funkce, podmínky definující relaci, ...):*

1) výraz zvýrazníme,

2) klikneme na , objeví se okno 2-D grafu,


3) klikneme ještě jednou na , nakreslí se graf,

4) pro větší názornost můžeme rozdělit pracovní plochu na dvě části, jednu pro okno 2-D grafu, druhé pro okno Algebra : **Okno** → **Vertikální dlaždice**

*Program nám umožňuje výše uvedeným postupem zobrazit i jenom část výrazu (když jí zvýrazníme).*

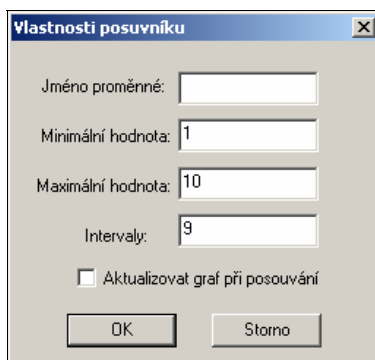
Pokud bychom chtěli výše uvedeným způsobem zobrazit i funkce  $f_1, f_2$ , odměnou by nám bylo hlášení **Příliš mnoho proměnných pro toto grafické okno!** V případě grafu funkce s parametrem musíme totiž zvolit tento postup:

1) výraz zvýrazníme,

2) klikneme na , objeví se okno 2-D grafu,

3) Při aktivním grafickém okně volíme příkaz **Vložit** → **Posuvník...**,

4) Objeví se dialogové okno

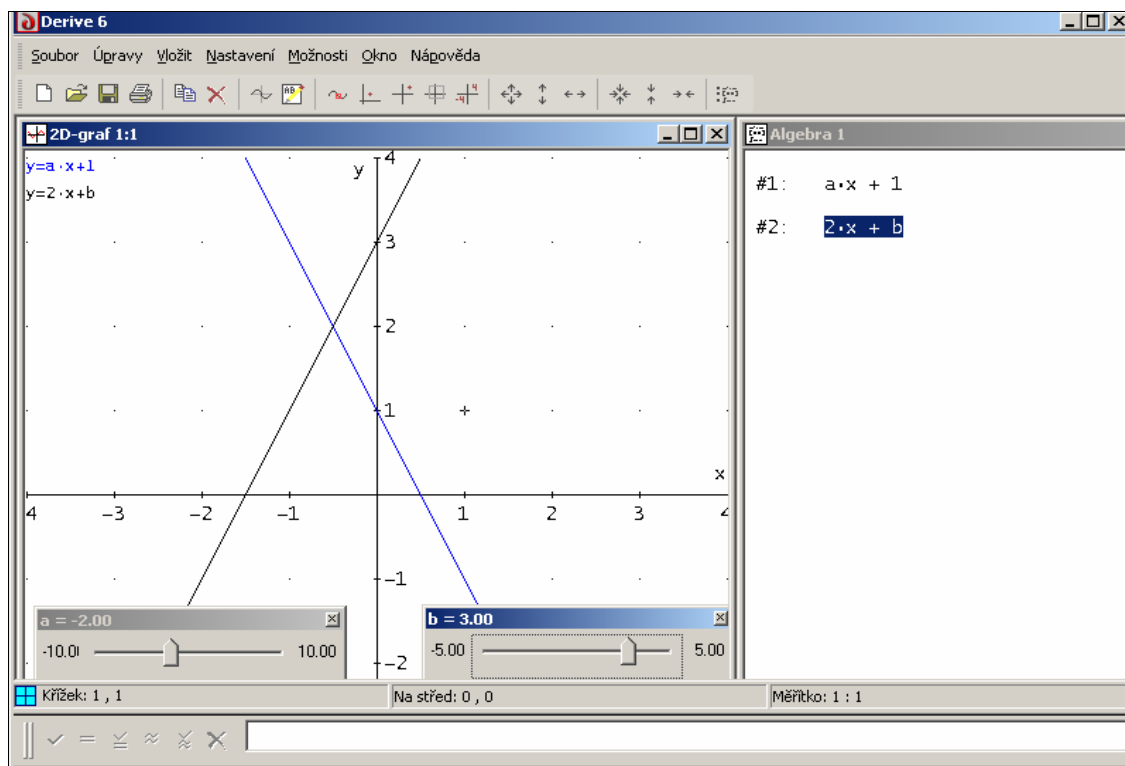


5) Vyplníme potřebné údaje a potvrdíme **OK**

*Pozn.:* Počtem intervalů ovlivníme jemnost změn hodnot parametru. Maximální hodnota je 50.

6) Objeví se příslušný posuvník (viz **Obr. 1.1**). Ten můžeme po uchopení myši přemístit na vhodnější místo. Změnu hodnoty parametru provádíme pohybem ukazatele posuvníku a to buď myší nebo, při aktivním okně ukazatele, pomocí kláves - šipek (doleva, doprava).

Na obrázku 1.1 si můžeme všimnout ještě dalšího nového prvku v Derive 6. Můžeme zde pomocí volby **Možnosti** → **Popsat nové grafy** nastavit režim popisu nových grafů. To je velice praktická funkce v okamžiku, kdy máme v grafickém okně více grafů. Ke každé funkci je v její barvě do grafického okna přiřazen příslušný předpis. Tyto popisky můžeme v rámci okna přemístit a tím ještě zvýšit názornost grafů.



**Obr. 1.1: Graf funkce s parametrem – využití posuvnému**

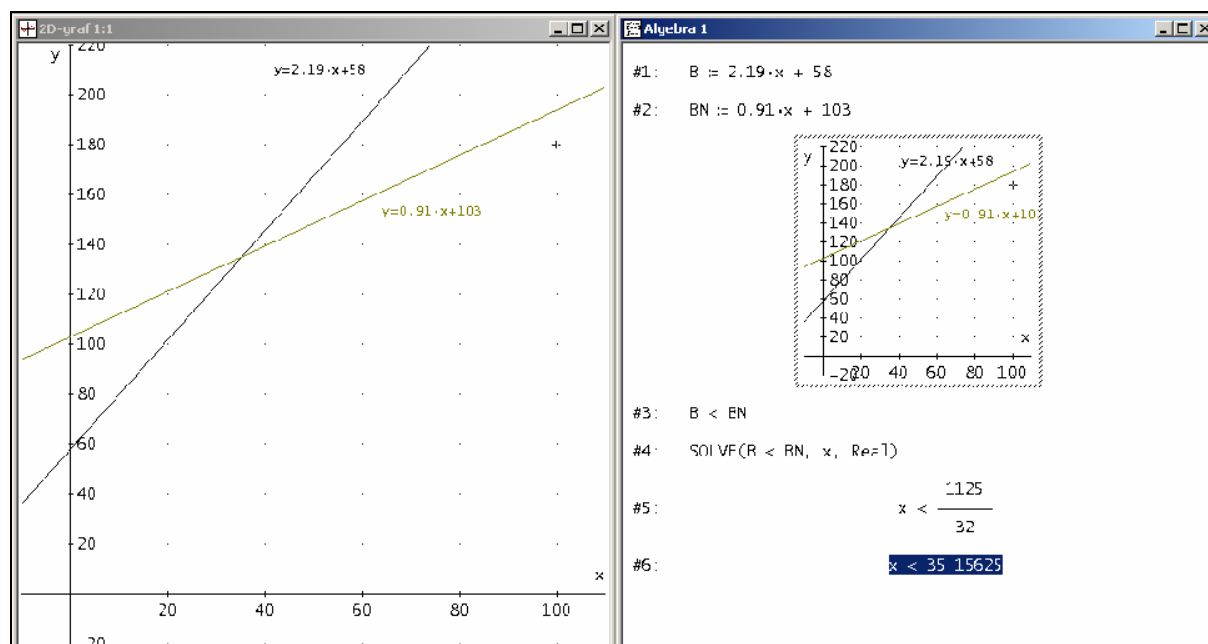
## 1.2. Lineární funkce při řešení slovních úloh

**PŘÍKLAD 1.2:** Existuje několik poplatkových sazeb za odběr elektrické energie. Nejčastěji se vyskytuje sazba B a sazba BN. V roce 1999 se v sazbě B platilo 2,19 Kč za 1 kWh a měsíční poplatek 58 Kč bez ohledu na spotřebu. V sazbě BN se platilo 0,91 Kč za 1 kWh a měsíční poplatek byl 103 Kč bez ohledu na spotřebu. Určete, při jaké měsíční spotřebě (vyjádřené s přesností na desetiny kWh) byla finančně vhodnější sazba B. Úlohu řešte graficky i početně.


(Šarounová, A. a kol.: Matematika 9, II. díl, Prométheus Praha, 2000, str. 135, Cv. 1)

*Řešení:*

Definujeme funkce B a BN (viz **Obr. 1.2**, řádky #1, #2 okna Algebra). Program Derive je po instalaci nastaven tak, že nepracuje s víceznakovými jmény proměnných. Chceme-li je používat, musíme nejprve změnit toto nastavení. To provedeme příkazem **Možnosti** → **Nastavení...** → **Vstup** → **Režim pro jména proměnných** → **Slovo**, nebo na příkazový řádek napíšeme přiřazovací příkaz `InputMode:=Word`. Tato opatření provádět **nemusíme**, pokud se víceznaková proměnná vyskytuje v levé části přiřazovacího příkazu (viz **Obr. 1.2**, #2).



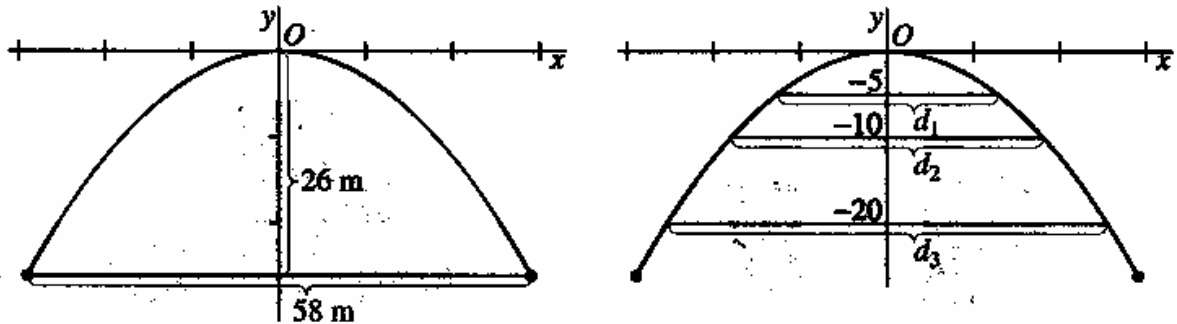
**Obr. 1.2: Lineární funkce**

Z grafu žák snadno pozná úskalí jednotlivých sazeb, pochopí souvislost grafu s předpisem lineární funkce, o které se učí a najde řešení úlohy  $B < BN$ . K tomu zřejmě užije režim Trasování grafů, aktivovaný ikonou  grafického okna. Takovýto výzkum je dobré podpořit i výpočtem. Řešíme tedy nerovnici  $B < BN$  (viz **Obr. 1.2**, řádek #3).

### 1.3. Nelineární funkce

U některých úloh, které třeba vypadají obtížně, můžeme program použít k ukázce toho, že řešení existuje a není nijak složité. Žák potom nechť přemýšlí, jak ho najít.

**PŘÍKLAD 1.3:** Architekt navrhuje lávku pro pěší. Stavitel lávky požaduje podrobnější informace o nosných obloucích. Architekt předkládá nákres nosného oblouku a doplňuje údaje: „Oblouk má rozpětí 58 metrů a výšku 26 metrů. V soustavě souřadnic na obrázku vlevo jde o část paraboly, která je grafem kvadratické funkce  $y = ax^2$ .”



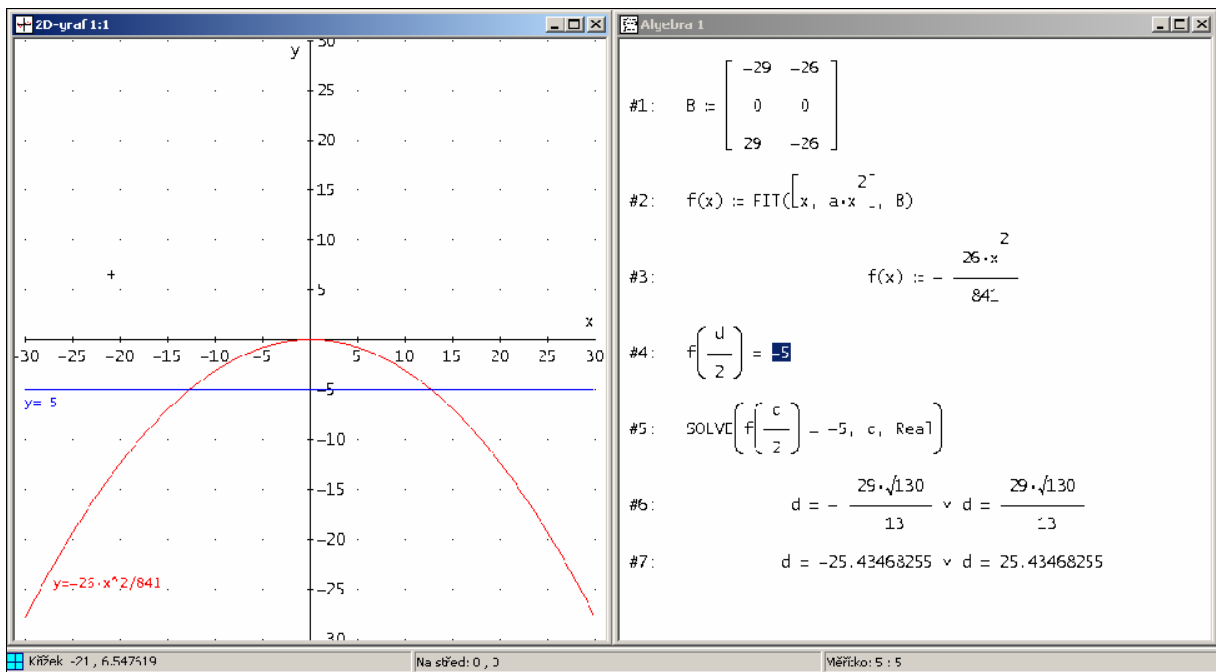
a) Urči ve vzorci  $y = ax^2$ , kterým je tato kvadratická funkce vyjádřena, hodnotu  $a$ .

b) Vypočítejte pro stavitele délky  $d_1, d_2, d_3$ , které vyznačil v obrázku vpravo.

(Odvárko, O., Kadlecěk, J.: Matematika pro 9. ročník z.š., 2. díl, Prométheus Praha, 2000, str. 34, Cv. 9)

*Řešení:*

Danými body máme proložit parabolou. K tomu použijeme příkaz **FIT**, jehož výstupem je rovnice hledané křivky.



Obr. 1.3: Kvadratická funkce – příkaz FIT

Těžko asi budeme po žácích chtít, aby tuto úlohu řešili v Derive sami. I když věřím, že syntaxi příkazu FIT by pochopili rychle. Přínosná by mohla být spíše diskuse nad řešením. Kolik bodů stačí, co když budeme chtít proložit jinou funkcí než kvadratickou, zda řešení závisí na umístění v soustavě souřadnic apod.

## 2. Rovnice

Při řešení rovnic se uplatní především grafické možnosti programu, tj. snadné a rychlé zobrazení hodnot nějakého výrazu v soustavě souřadnic. Určitě oceníme i možnost provádět jednotlivé úpravy rovnice krok za krokem

### 2.1. Úpravy rovnic

Program Derive umožňuje provádět úpravy rovnic krok za krokem, tak, jak postupujeme při počítání na papíře či tabuli. Velice jednoduše můžeme provádět různé početní operace s oběma stranami rovnice současně a neustále sledovat vliv těchto úprav na dílčí rovnice, od zadání až po výsledek. Pomocí příkazu pro substituci můžeme potom snadno vykonat zkoušku.

#### **PŘÍKLAD 2.1:** Řešte rovnici $2x + 5 = 27$

*Řešení:*

Jak bylo řečeno výše program Derive umožňuje provádět **ekvivalentní úpravy rovnic** přímo na pracovní ploše. Chceme-li například od obou stran rovnice

**#1:**  $2x + 5 = 27$

odečíst 5, napíšeme na příkazový řádek jednoduše **#1+5** (tj. na rovnici se odkazujeme prostřednictvím proměnné **#1**) nebo rovnici zkopírujeme do závorek pomocí **F4** a odečteme od ní onu pětku. Na následujícím řádku potom dostaneme (viz též **Obr. 2.1**)

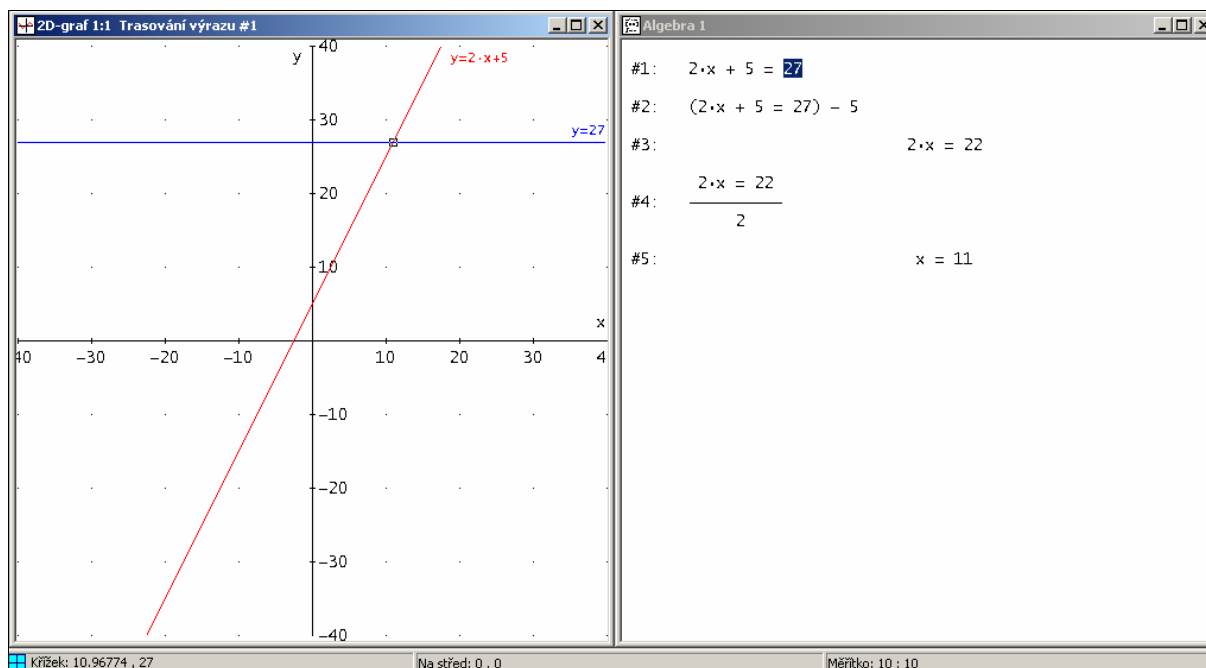
**#2:**  $(2x + 5 = 27) - 5$

Po zjednodušení příkazem **Zjednodušit** → **Základní zjednodušení** (**=**) dostaneme výsledek

**#3:**  $2x = 22$

V úpravách pokračujeme ve stejném duchu (viz **Obr. 2.1**)

Výsledek úprav můžeme potom konfrontovat s grafickým řešením rovnice. To dostaneme tak, že stejným způsobem, jakým jsme zobrazovali grafy funkcí, zobrazíme zvláště levou a pravou stranu rovnice (viz **Obr. 2.1**, grafické okno).



Obr. 2.1: Úpravy rovnice a grafické řešení.

## 2.2. Diofantická rovnice

Počítačový program může žákům pomoci dotáhnout do konce řešení úlohy, jejíž zadání je snadno pochopitelné, matematické vyjádření nepřesahuje rámec učiva, ale úpravy vedoucí k řešení jsou natolik komplikované nebo speciální, že není nijak přínosné s nimi všechny žáky zatěžovat. Jak ukazuje následující příklad, program v takovém případě rozhodně nemusí hrát roli nějaké černé skříňky, do níž strčíme zadání a vypadne správné řešení. Naopak, postup řešení takovéto úlohy se zápisem v programu stane zřejmý a nijak obtížný.

**PŘÍKLAD 2.2:** Chytali spolu čtyři poctiví rybáři. Skončili za tmy, úlovek nepočítali a šli spát. V noci jeden vstal, rozpočítal chycené ryby na čtyři stejné díly, jedna ryba mu zbyla a tak ji hodil do moře. Vzal si svou čtvrtinu úlovku a odešel. Totéž udělal po chvíli druhý rybář, potom třetí a čtvrtý. Jaký je nejmenší počet ryb, které rybáři ulovili?

*Řešení:*

Hledaný nejmenší počet ryb označíme proměnnou  $n$ . Podíly jednotlivých rybářů potom proměnnými  $k, l, m, o$ . Pomocí přiřazovacích příkazů popíšeme zmiňovaná čtyři noční dělení (viz Obr. 2.2, #1, #2, #3, #4). Potom zadáme příkaz ke zjednodušení proměnné  $n$  ve tvaru:  $n = \text{Enter}$ . Výsledkem je výraz #5. Protože všechny uvedené proměnné mohou nabývat pouze



nezáporných celočíselných hodnot (diofantické rovnice = rovnice řešené v oboru celých čísel), využijeme příkaz VECTOR a do výrazu #5 dosadíme za  $o$  několik (27) prvních kladných celých čísel. Výsledkem je potom první celočíselná hodnota  $n$ , tj. 253.

```
#1:  n := 4·k + 1
#2:  k :=  $\frac{4}{3} \cdot l + \frac{1}{3}$ 
#3:  l :=  $\frac{4}{3} \cdot m + \frac{1}{3}$ 
#4:  m :=  $\frac{4}{3} \cdot o + \frac{1}{3}$ 
#5:  n =  $\frac{256 \cdot o + 175}{27}$ 
#6:  VECTOR(n, o, 1, 27)
#7:  [  $\frac{431}{27}, \frac{229}{9}, \frac{943}{27}, \frac{1199}{27}, \frac{485}{9}, \frac{1711}{27}, \frac{1967}{27}, \frac{247}{3}, \frac{2479}{27},$ 
       $\frac{2735}{27}, \frac{997}{9}, \frac{3247}{27}, \frac{3503}{27}, \frac{1253}{9}, \frac{4015}{27}, \frac{4271}{27}, \frac{503}{3}, \frac{4783}{27},$ 
       $\frac{5039}{27}, \frac{1765}{9}, \frac{5551}{27}, \frac{5807}{27}, \frac{2021}{9}, \frac{6319}{27}, \frac{6575}{27}, 253, \frac{7087}{27} ]$ 
```

Obr. 2.2: Rybáři – diofantická rovnice

### 3. Číselné soustavy

Následující úloha je z kategorie motivačních problémů, které s největší pravděpodobností vyřeší vyučující sám, v lepším případě k řešení žáky dovede. Smyslem úlohy je uvést jisté téma nebo metodu. V tomto případě se jedná o číselné soustavy o různých základech. Program Derive nám v tomto případě jenom poskytne svou schopnost provádět rychle a jednoduše převod mezi soustavami o různých základech.

**PŘÍKLAD 3.1:** Některá čísla zobrazená na displeji kalkulačky dávají smysl i když se na ně díváme obráceně, tj. když kalkulačku otočíme „nohama vzhůru“. Například z čísla 1995 se stane 5661. Páté číslo, které je čitelné i nohama vzhůru je 8 a patnácté číslo s touto vlastností je 21, které se otočením změní na 12. Určete milionté číslo, které je čitelné i nohama vzhůru.

*Řešení:*

Myšlenka řešení je založena na předpokladu, že číslice, které jsou čitelné i „nohama vzhůru“ (tj. 1, 2, 5, 6, 8, 9, 0) představují úplnou sadu číslic číselné soustavy o základu 7; 0=0, 1=1, 2=2, 3=5, 4=6, 5=8, 6=9. Naším úkolem je potom jenom najít zápis čísla 1 000 000 v této soustavě. Nastavíme-li výstup hodnot v programu Derive v sedmičkové soustavě, stačí napsat na příkazový řádek číslo 1000000 a v algebraickém okně se objeví příslušné číslo.

Nastavení provedeme příkazem (Derive 6)

**Možnosti** → **Nastavení...** → **Výstup** → **Soustava** → 7

Pozn.: Nenechme se splést výsledkem příkazu (viz řádek #1 níže). Výstup je přece v sedmičkové soustavě.

#1:	OutputBase := 10
#2:	11333311

**Obr. 3.1: Číselné soustavy**