

Jihočeská univerzita v Českých Budějovicích

Pedagogická fakulta

Katedra matematiky



## **Využití programu Rhinoceros při výuce Deskriptivní geometrie**

Diplomová práce

Čečková Veronika

vedoucí diplomové práce

Mgr. Roman Hašek, Ph.D.

České Budějovice 2007

Anotace

**Název:** Využití programu Rhinoceros při výuce Deskriptivní geometrie

**Vypracoval:** Veronika Čečková

**Vedoucí práce:** Mgr. Roman Hašek, Ph.D.

**Klíčová slova:** Program Rhinoceros, prostorová představivost, výuka pomocí počítačů, stereoskopie, deskriptivní geometrie, smysluplnost.

Obsahem této práce je seznámení s obsluhou a možnostmi CAD programu Rhinoceros a jeho využití při výuce. Práce je především zaměřena na deskriptivní geometrii a její představivost v prostoru. Posouzení kladů a záporů zvoleného programu při výuce.

**Title:** Usage of computer program Rhinoceros in the lessons of Descriptive geometry

**Author:** Veronika Čečková

**Supervisor:** Mgr. Roman Hašek, Ph.D.

**Key words:** Computer program Rhinoceros, spatial imagination, education with computers, descriptive geometry, sense.

This work is about the acquaintance with the possibilities and manipulation of CAD program Rhinoceros and its usage in lessons. First of all the work is focused on descriptive geometry and its imagination in space. It is also about the assessment of advantages and disadvantages of chosen program in lessons.

## Poděkování

Ráda bych poděkovala panu Romanu Haškovi za podporu při zpracování této práce a za zapůjčení materiálů. Dále bych ráda poděkovala svojí rodině za trpělivost v celém průběhu mého studia a Anně Šůsové za kontrolu matematického vyjádření. V neposlední řadě děkuji za odbornou pomoc při kontrole pravopisu paní učitelce Janě Hůlové a paní učitelce Janě Fríbertové za pomoc s překladem do anglického jazyka ze ZŠ Poděbradova ve Strakonících.

Prohlašuji, že svoji diplomovou práci jsem vypracovala samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své diplomové práce, a to v nezkrácené podobě elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách.

26. duben 2007

# Obsah

1. Rhinoceros 2.0.....	7
1.1 Systémové požadavky.....	7
1.2 Ceník.....	8
1.3 Instalace.....	8
2. Základy programu Rhinoceros.....	12
2.1 Základní obrazovka.....	12
2.2 Nastavení modelu.....	14
2.3 Vrstvy.....	15
2.4 Modelovací pomůcky.....	16
2.5 Navigace v pohledech.....	17
3. Základní modelování.....	18
3.1 Přesné modelování.....	18
3.2 Bod.....	19
3.3 Čára.....	20
3.4 Kružnice.....	21
3.5 Oblouk.....	23
3.6 Elipsa a pravidelné n-úhelníky.....	24
3.6.1 Elipsa.....	24
3.6.2 Obdélník.....	25
3.6.3 Pravidelný n-úhelník a čtverec.....	26
3.7 Tělesa.....	27
3.8 Stínování.....	29
3.9 Analytické příkazy.....	30
4. Deskriptivní geometrie.....	31
4.1 Kótované promítání.....	31
4.1.1 Sklápění roviny.....	32
4.1.2 Otáčení roviny.....	33
4.2 Mongeovo promítání.....	33
4.2.1 Bod.....	34
4.2.2 Přímka.....	34
4.2.3 Rovina.....	35

4.2.4 Otáčení roviny.....	35
4.3 Kosoúhlé promítání.....	36
4.3.1 Bod, přímka a rovina.....	37
4.4 Pravoúhlá axonometrie.....	38
4.4.1 Zobrazení bodu, přímky a roviny.....	39
5. Příklady.....	41
5.1 Rovinné obrazce.....	41
5.2 Trojrozměrné objekty.....	53
6. Závěr.....	60

# 1. Rhinoceros 2.0

S rychlým rozvojem počítačové techniky a vytvářením programů se otvírají nové možnosti i ve školství. I matematika je předmětem, kde lze s úspěchem využívat počítač jak při samostatné práci, procvičování, ale i při výkladu. Počítače mohou i výrazně přispět při výuce deskriptivní geometrie, neboť žáci mají často problémy s prostorovou představivostí. Při výuce si lze vybírat z mnoha programů. V bohaté nabídce je zapotřebí se orientovat a vybrat vhodný program s ohledem na možnosti žáků, školy.

Mým cílem je seznámení s programem Rhinoceros a zjištění kladů a záporů při jeho využití ve školství.

Rhinoceros je komerční, programový nástroj pro 3D modelování. K reprezentaci dokonale hladkých křivek a ploch využívá NURBS<sup>1</sup> geometrii. NURBS jsou matematickým vyjádřením geometrických objektů, které umožňují přesně definovat jakýkoliv tvar.

Díky velké oblibě programu jde vývoj velice dopředu. V dnešní době je již na trhu Rhinoceros 3.0 a stále se vyvíjí. Na konci loňského roku se objevila betaverze programu Rhinoceros 4.0.

## 1.1 Systémové požadavky

Program Rhinoceros není náročný na požadovaný hardware. Lze spustit na běžném i přenosném PC.

- Pentium, Celeron nebo vyšší procesor
- Windows 98/NT/ME/2000/XP
- 65 MB na pevném disku
- 64 MB RAM (doporučeno je 128 MB a více)

---

1 NURBS = Non-Uniform Rational B-Splines

## ***1.2 Ceník***

### Rhinoceros 3.0 Commercial

- ◆ komerční licence pro jednoho uživatele
- ◆ uživatelská příručka a CD
- ◆ 31.900 Kč
- ◆ upgrade 9.900 Kč

### Rhinoceros 3.0 Educational

- ◆ školní licence pro jednoho uživatele
- ◆ totožná s komerční softwarem
- ◆ nutno přiložit potvrzení o studiu nebo o zaměstnaneckém poměru ve školství
- ◆ 6.600 Kč
- ◆ upgrade 95 Kč

### Rhinoceros 3.0 Educational Lab Kit

- ◆ školní multilicence
- ◆ písemná potvrzená objednávka školou
- ◆ 3 uživatelské příručky, 1 učebnice, učební osnovy a CD
- ◆ 31.500 Kč
- ◆ upgrade 295 Kč

V ceně je započtena doprava od výrobce a veškeré související náklady a poplatky.  
Cena je uvedena bez DPH.

## ***1.3 Instalace***

Instalace programu se nám zahájí samostatně (pokud máme nakonfigurované automatické spouštění z cd-romky) nebo spuštěním souboru Setup.exe z cd-čka. Po celou dobu instalace jsme vedeni průvodcem instalací.



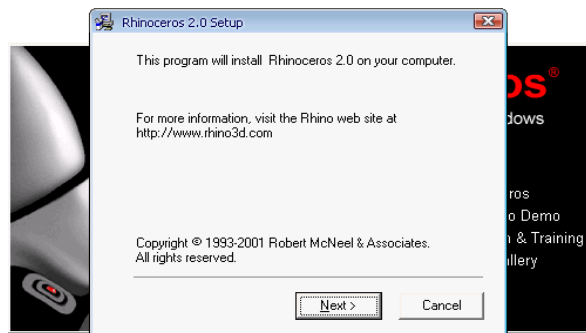
Kroky při instalaci:

Kliknout myší na položku Install Rhinoceros



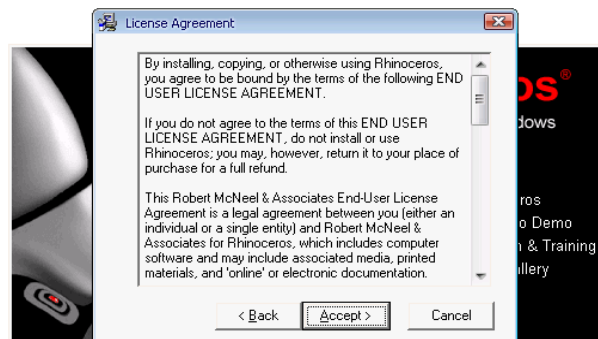
Obr. 1: Kroky instalace

Další okno průvodce informuje o instalované verzi programu a oznamuje webovou adresu, kde můžeme najít různé informace k programu.



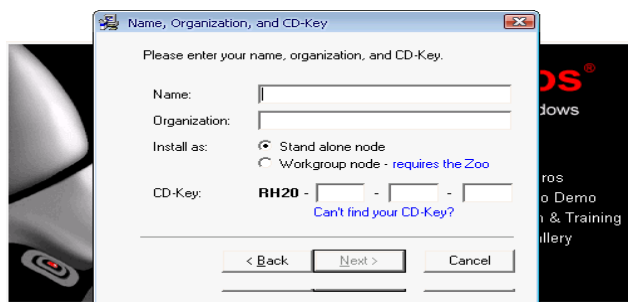
Obr. 2: Kroky instalace

Seznámení s licenčním ujednáním koncového uživatele. Pokud uživatel s podmínkami nesouhlasí, je instalace ukončena.



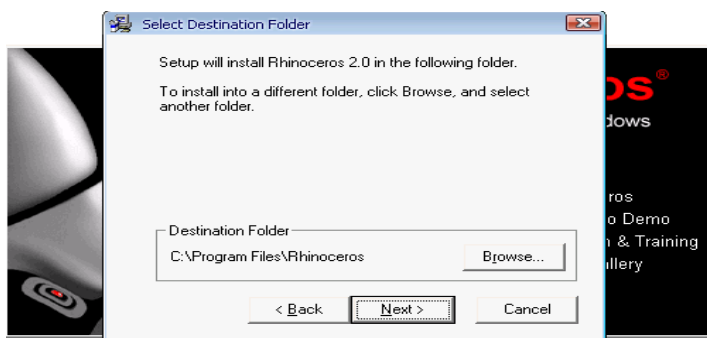
Obr. 3: Kroky instalace

Dále zadáváme organizaci a klíč k programu, který získáme po koupi s instalačním cd-čkem.



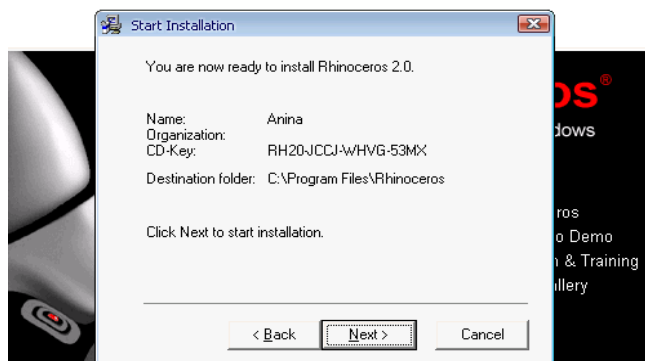
Obr. 4: Kroky instalace

Určení cílového adresáře, kam bude provedena instalace programu Rhinoceros (možno ponechat standardní nastavení).



Obr. 5: Kroky instalace

Průvodce je připraven k provedení instalace na pevný disk. V tomto okně můžete zkontrolovat její veškeré parametry a začít samotnou instalaci.



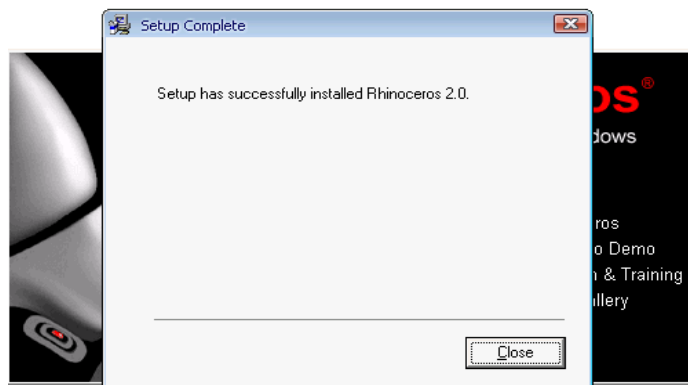
Obr. 6: Kroky instalace

V dalším kroku sledujeme průběh instalace.



*Obr. 7: Kroky instalace*

Poslední část je pouze informativní o tom, že instalace byla zdárně dokončena.



*Obr. 8: Kroky instalace*

Po instalaci se nám na ploše objeví nová ikona pro spuštění programu Rhinoceros 2.0.



Rhinoceros

*Obr. 9: Ikona*

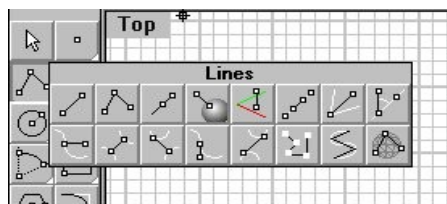
## 2. Základy programu Rhinoceros

V této kapitole se seznámíme s grafickým rozhraním programu Rhinoceros, jeho úpravou pro přizpůsobení a základními nástroji pro nastavení modelování.

### 2.1 Základní obrazovka

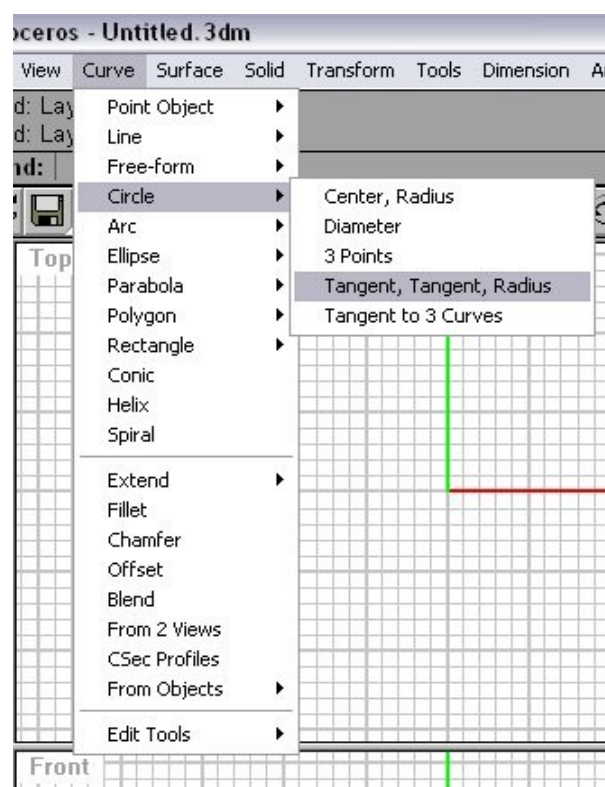
Základní obrazovka programu Rhinoceros je rozdělena na šest hlavních částí. Jedním z nich je roletové menu. Roletové menu slouží jako u každého programu pro veškerou činnost. Dají se zde vyvolávat příkazy, různé volby pro nastavení a v neposlední řadě i nápověda. Pod roletovým menu se nachází příkazový řádek, kde můžeme ručně vypisovat příkazy, nebo se dozvídáme informace a výzvy pro modelování. Při ručním vypisování je zde usnadnění s automatickým doplňováním slov pomocí klávesy „Tabulátor“. Další důležitou částí jsou palety nástrojů, které jsou rozděleny na dva druhy.

Jedny slouží pro práci se souborem a druhé jsou pro vykreslování modelu. Palety



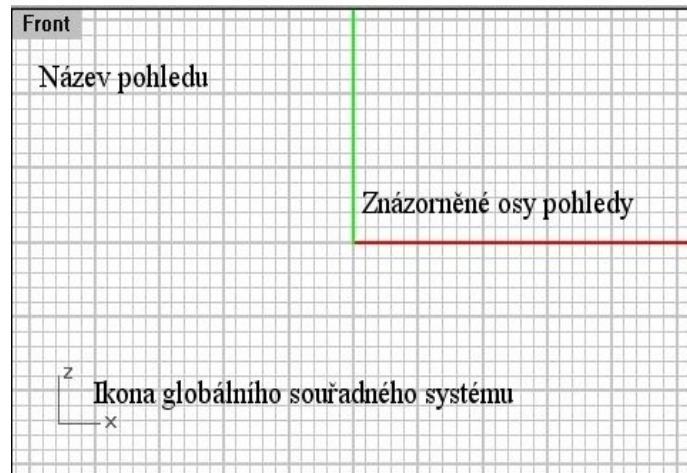
Obr. 11: Kaskádová paleta

obsahují tlačítka, které nám spouštějí různé příkazy. Většina tlačítek má dva různé příkazy. Ty se dozvíme po najetí myši, kde se nám zobrazí bublinková nápověda. Pokud je v pravém dolním rohu trojúhelníček, značí nám tzv. kaskádové tlačítko. Pod ním se nám ukrývá další paleta pouze s daným podtématem, která se nám zobrazí přidržetím levého tlačítka myši. Největší

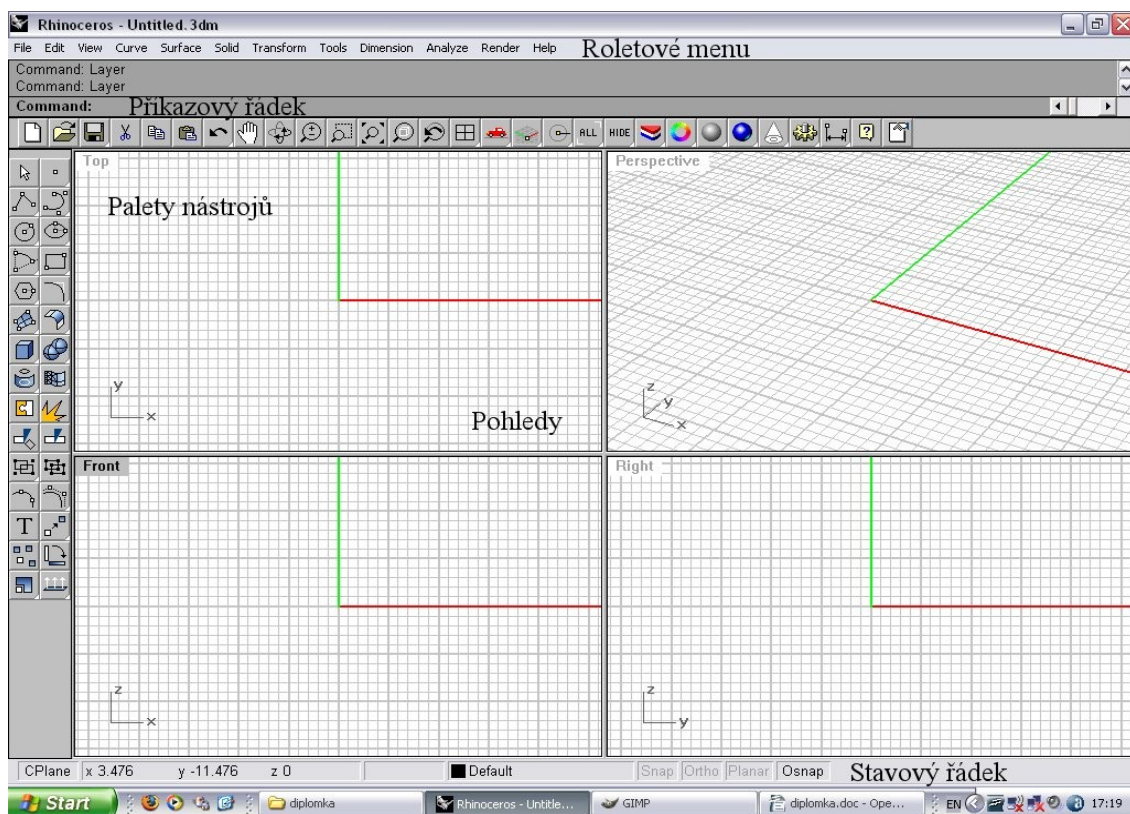


Obr. 10: Menu

část obrazovky nám zabírá grafická část. Ta je plně přizpůsobitelná podle požadavků a zobrazují se v ní pohledy na daný model. Pohledy lze libovolně přesunovat a měnit jejich velikost. Každý má svou vlastní konstrukční rovinu, lze s nimi rotovat, posunovat, přibližovat a oddalovat. Poslední částí obrazovky programu je stavový řádek. Ten nás informuje o souřadnicích kurzoru a jsou zde obsaženy volby pro vrstvy a různé přepínače pro modelování.



Obr. 12: Okno pohledu

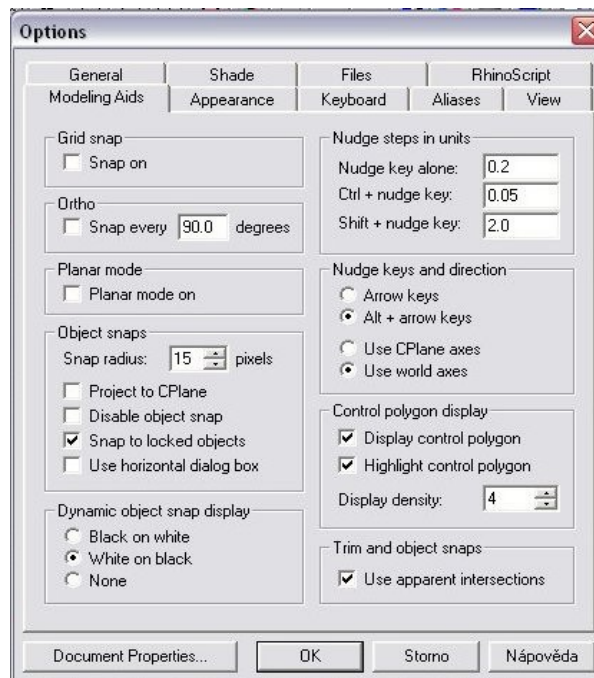


Obr. 13: Základní obrazovka

## 2.2 Nastavení modelu

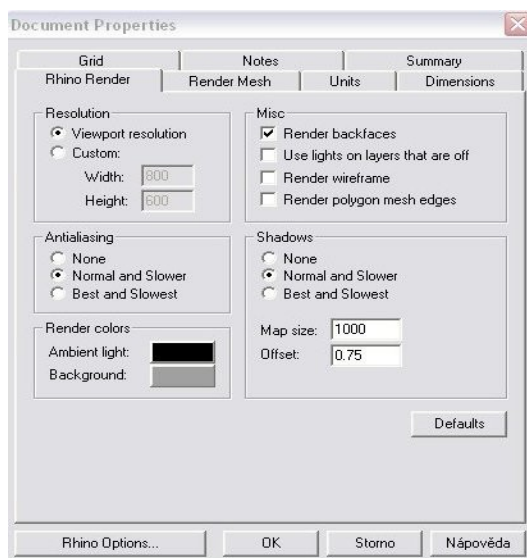
Při různorodosti modelování v programu si můžeme měnit modelovací prostředí. Pro změnu si zvolíme položku v roletovém menu Tools/Options. Zde se nám zobrazí panel s mnoha záložkami.

- General
- Shade – nastavení pro vystínování modelu
- Files – cesta k základním souborům
- RhinoScript



Obr. 14: Nastavení modelu

- Modeling Aids – umožňuje měnit nastavení přepínačů pro modelování
- Appearance – barevné vyjádření programu
- Key – základní klávesové zkratky
- Aliases – možnost samostatného navolení zkratk funkcí
- View



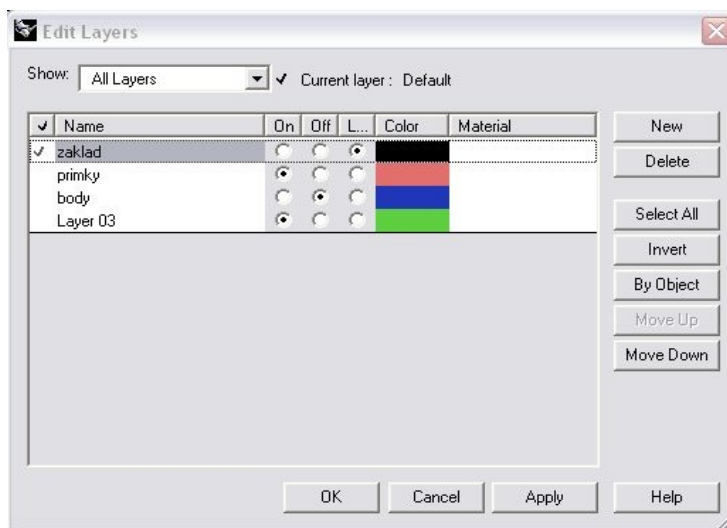
Obr. 15: Nastavení dokumentu

Další možností je nastavení dokumentu. Na něj se dostaneme klepnutím na tlačítko Document Properties ... v nastavení. Ta obsahuje další záložky pro modelování.

- Rhino Render – nastavení pro renderování
- Render Mesh
- Units – zobrazování v jednotkách
- Dimension - text
- Grid – přizpůsobení pohledů
- Notes
- Summary

## 2.3 Vrstvy

Vrstvy v programu slouží pro zpřehlednění modelu. Každá vrstva může zobrazovat určitou část a nezávisle na zbytku lze editovat. Počet vrstev není omezen. Každá vrstva má přiřazenou svoji barvu a název, které se dají měnit a lze je libovolně zviditelňovat. Do editace vrstev se



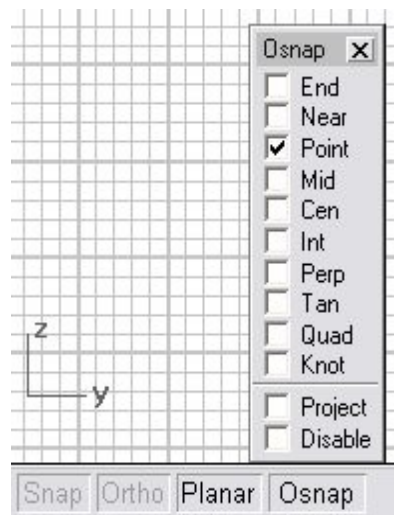
Obr. 16: Vrstvy

dostaneme pomocí Edit/Layers nebo dvojitým poklepáním ve stavovém řádku na aktivní vrstvě.

## 2.4 Modelovací pomůcky

Přepínače slouží pro lepší modelování. Lze je zapínat pomocí zkratk, kláves nebo tlačítek.

- Ortho – omezení pohybu kurzoru pod určitým úhlem. Výchozí nastavení je 90 stupňů.
- Grid – skrývá a zobrazuje mřížku v pohledech.
- Snap – kurzor přiskakuje k uzlovým bodům mřížky.
- Planar – podobná jako Ortho. Pomáhá modelovat rovinné obrazce.
- Osnap – sada pomůcek pro uchopování definovaných míst na objektech. Slouží pro přiskočení kurzoru do nejbližšího bodu.
  - ✓ End – konec křivek
  - ✓ Near – nejbližší bod nebo existující křivka
  - ✓ Point – bod nebo řídicí bod
  - ✓ Midpoint – bod uprostřed křivky
  - ✓ Center – střed křivky
  - ✓ Intersection – průsečík dvou křivek
  - ✓ Perpendicular To – bod na křivce, daný kolmicí z naposledy zadaného bodu
  - ✓ Tangent To – bod na křivce, daný tečnou z naposledy zadaného bodu
  - ✓ Quad – kvadranty kružnic, oblouků a elips
  - ✓ Knot
  - ✓ Project – promítá úchopný bod do konstrukční roviny
  - ✓ Disable – dočasně vypíná zapnuté režimy



Obr. 17: Modelovací režimy



## ***2.5 Navigace v pohledech***

Při různých činnostech modelování je potřeba se pohybovat v pohledech. Za těchto okolností můžeme ztratit orientaci. Obnovení původního pohledu provedeme klávesou Home. Posunutí pohledu se provádí pomocí klávesy Shift a pravého tlačítka myši při vlečení. Pokud bychom chtěli model přiblížit nebo oddálit, použijeme klávesu Ctrl a opět pravé tlačítko myši, tentokrát s pohybem nahoru a dolů, nebo u myši s rotačním kolečkem pomocí tohoto kolečka. Tuto navigaci lze také vyvolat příkazem View/Zoom/. Dále můžeme objekty různě přemísťovat. Po klepnutí daný objekt zežlutne a při uchopení myši a vlečení jej přesunujeme.

### 3. Základní modelování

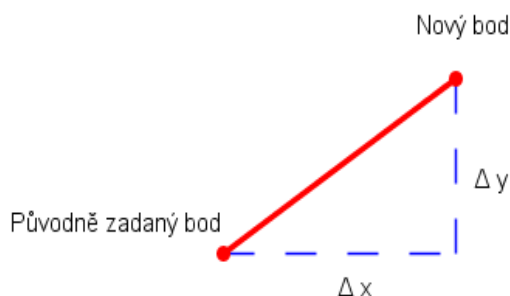
Další kapitola nás provede již funkcemi pro modelování. Ukážeme si základní postupy pro tvorbu modelu

#### 3.1 Přesné modelování

Přesné modelování slouží pro kreslení čar na přesně zadané souřadnice. K tomu využívá kartézský souřadný systém, kde se zobrazují pomocí os  $x$ ,  $y$  a  $z$ . To se dá provádět více možnostmi.

První možností je zadávání pomocí absolutních souřadnic. Zde udáváme přesné umístění bodu v osách  $x$ ,  $y$  a  $z$ . Místo umístění bodu pomocí myši napíšeme do příkazového řádku na výzvu bodů dané souřadnice oddělené čárkami bez mezer.

Další způsob, o něco rychlejší pro kreslení pravidelných  $n$ -úhelníků než za pomoci absolutních souřadnic, jsou relativní souřadnice. V tomto případě si program pamatuje poslední zadanou souřadnici a od ní se orientuje pro nový bod. Pro relativní souřadnice stačí napsat před souřadnice bodu písmeno  $R$  bez mezery.



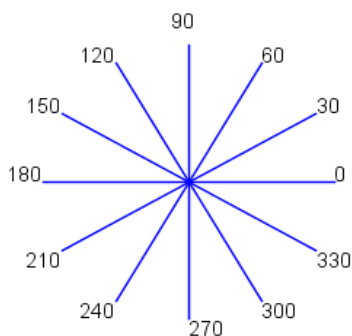
Obr. 18: Relativní souřadnice

Posledním možným způsobem je využití polárních souřadnic. Bod zadáváme pomocí vzdálenosti a směru od počátku. Nový bod udáváme ve formě vzdálenost<úhel. V tomto případě můžeme i nakombinovat s relativními souřadnicemi. Stačí, když před daný vstup napíšeme znak  $R$  opět bez mezery, ale nyní si bereme údaje z naposledy zadaného bodu.

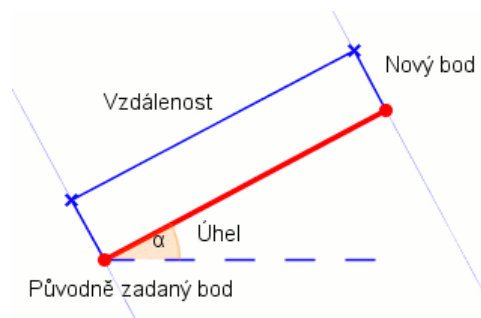
Další možnost přesného modelování je omezení vzdálenosti a úhlu. Tím lze rychle zadávat přesnou délku a úhel čar. Při omezení vzdálenosti stačí udat danou hodnotu a

stisknout klávesu Enter. Pro určení úhlu již zadáváme <úhel se stiskem Enter. Obě možnosti bere od naposledy určeného bodu.

Pokud chceme efektivně kreslit kolmé čáry, používáme režim Ortho. Pro dočasné zapnutí tohoto režimu držíme při zadávání klávesu Shift.



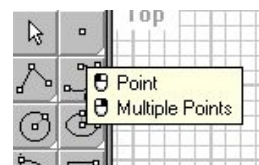
Obr. 19: Polární souřadnice - úhel



Obr. 20: Polární souřadnice - vzdálenost

### 3.2 Bod

Body jsou jedním ze základních stavebních kamenů při rýsování. V programu Rhinoceros se dají zadávat více způsoby. Jedním z nich je nabídka z roletového menu Curve/Point Object a další způsob máme pomocí tlačítek na panelu nástrojů.



Obr. 21: Nápověda bodu

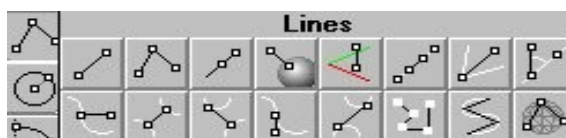
Pokud bychom chtěli zadávat přímo přes příkazový řádek, stačí znát příkaz pro jeden bod (Point) nebo více bodů (Points). Po vyvolání příkazu jakýmkoliv způsobem stačí udat místo v pohledu nebo přesné zadání pomocí souřadnic.



Obr. 22: Kaskádový panel bodu

### 3.3 Čára

Dalším základním kamenem je propojování bodů mezi sebou čarami. Ty lze opět rýsovat více možnostmi. Záleží jaká čára je zrovna potřeba.



Obr. 23: Celková paleta pro čáry

Single Line – samostatný čárový segment. Je zadán pomocí dvou bodů.



Po nakreslení segmentu se příkaz automaticky ukončí. Dá se spustit příkazem Line. Zobrazí se výzva Start of line (BothSides), kde zadáme první bod čáry a přeneseme se na výzvu End of line (BothSides) a výběrem bodu příkaz ukončíme. Výraz v závorce (BothSides) znamená volbu pro vykreslení čáry na obě strany od prvního zadaného bodu. Tuto volbu zaktivujeme kdykoliv v průběhu příkazu zapsáním písmena B potvrzené klávesou Enter.

Line Segments – jsou čárové segmenty, které na sebe navazují, nejsou však spojeny



v jednu čáru. V tomto případě můžeme zadávat více bodů a mezi nimi vzniká čára. Můžeme ji spustit příkazem Lines nebo pravým tlačítkem myši na daném tlačítku. Zobrazují se postupně dané výzvy, při kterých se zadávají body. Start of first line, End of line segment (Undo) a další, která se stále opakuje až do ukončení příkazu, End of line segment, press Enter when done (Undo). I u tohoto příkazu máme možnosti voleb. Celkem jsou tři. První je písmeno C, které ukončí příkaz a spojí poslední zadaný bod s prvním bodem (uzavře obrys). Další je možnost U pro zrušení posledního zadaného bodu. Poslední volba je Enter, která ukončí příkaz, ale neuzavře obrys.

Polyline – je lomená čára. Značí jednu čáru, skládající se interně z více segmentů.

Tlačítko pro spuštění má stejné jako lines, ale kliká se levým tlačítkem myši. Z příkazového řádku se spouští slovem Polyline. Má následující výzvy. Start of polyline, Next point of polyline (Undo), Next point of polyline, press Enter when done (Undo) a poslední stále se opakující do ukončení Next point of polyline, press Enter when done (Close, Undo). Tato funkce má stejné volby jako u příkazu Lines.

Tangent to 2 Curves – tato čára propojuje dvě kružnice. Spustí se příkazem LineTT

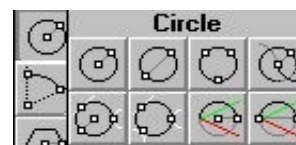


a má následující výzvy. Select first curve near tangent point, kde označíme bod

dotyku na první křivce a Select second curve near tangent point, což je druhý bod dotyku na další křivce.

### 3.4 Kružnice

Další roli v rýsování hrají kružnice. I zde nabízí program Rhinoceros více možností. Všechny možnosti najdeme v roletovém menu pod položkou Curve/Circle.




Obr. 24: Kružnice


Center, Radius – kružnice zadaná středem a poloměrem


nebo průměrem. Ten si můžeme přepnout v průběhu modelování na zadanou výzvu. Funkce se mimo jiné spouští příkazem Circle. V příkazovém řádku se nám objeví první výzva Center of circle (Deformable Vertical Around Curve), kde zadáme střed kružnice. Pokračujeme s výzvou Radius <x> (Diameter). Číslo x v závorce značí poloměr naposledy zadané kružnice. Pokud bychom chtěli jiný číselný údaj, stačí napsat novou hodnotu a potvrdit Entrem. Slovo Diameter je právě již zmíněná volba pro kružnici zadanou středem a průměrem. Dostaneme se na ni, když místo hodnoty poloměru vypíšeme písmeno D. Tím se nám změní výzva na Diameter <x> (Radius). Kdybychom teď zadali číselnou hodnotu stejnou jako v předchozím případě, vykreslí se nám kružnice menší z toho důvodu, že hodnota již nebude poloměr ale průměr. Pokud bychom si rozmysleli rýsování s průměrem, můžeme se zadáním písmena R vrátit zpět na funkci kreslení kružnice pomocí středu a poloměru.

3 Points – kružnice rýsovaná za pomoci tří bodů na obvodu. Z příkazového řádku

se spustí pomocí Circle3Pt. Na následující výzvy vždy zadáváme bod. First point on circle, Second point on circle a poslední Third point on circle. V tomto případě nejsou na výběr jiné volby.

Diameter – je speciální zadání kružnice. Využívá pouze dvou bodů náležící kružnici, které jsou určeny průměrem. Tyto body tedy leží na ose dané kružnice.  Vykoná se též pomocí příkazu CircleD, který postupně znázorní dané výzvy. Start of diameter (Deformable Vertical) pro první bod a End of diameter, kde můžeme určit přímo bod nebo nejdříve zadat hodnotu průměru potvrzenou Entrem a pak už jen nasměrovat kružnici do dané polohy.

Tangent, Tangent, Radius – další případ vytvoření kružnice za pomoci poloměru a  dvou tečných bodů. V tomto případě může na konci vzniknout i více odlišných kružnic najednou, které ovšem splňují zadané podmínky. Po spuštění příkazu CircleTTR máme první výzvu Select first curve for CircleTTR, kde vybíráme jeden z tečných objektů. Další výzva Select second curve for CircleTTR upozorňuje na výběr druhého tečného objektu a nakonec výzva Radius <x>, kde zadáme hodnotu poloměru a potvrdíme Entrem.

Tangent to 3 Curves – tato funkce zobrazuje kružnici, která se dotýká třech tečných  objektů. Tento případ má ovšem nevýhodu, protože nebere přímo zadaná místa jako tečný bod, nýbrž pouze určuje jakého objektu se má výsledná kružnice dotýkat. Spustí se příkazem CircleTTT s následujícími výzvami. First tangent curve (KeepMarks Point), Second tangent curve or radius <x> (KeepMarks Point) a poslední Third tangent curve. Press Enter to draw circle from first two points (KeepMarks Point). I v tomto případě mohou vzniknout nechtěné kružnice navíc.

Diameter – stejné zadání jako pro určení kružnice ze dvou bodů má i kružnice kolmá ke konstrukční rovině. Rovněž se při ní využívají jen dva body, které nám určují její průměr. Abychom vykreslily kružnici kolmou ke konstrukční rovině, stačí nám při první výzvě Start of diameter (Deformable Vertical) potvrdit písmeno V. Následují malinko odlišné výzvy, které zní. Start of diameter a End of diameter. Takto vykreslená kružnice bude viditelná v jiném pohledu.

Center, Radius – i tento příkaz má dvě různé možnosti zobrazení. První je již výše uvedena a druhá značí kružnici kolmou na jinou křivku. Na tuto možnost se dostaneme

při první výzvě Center of circle (Deformable Vertical Around Curve), kde zapíšeme znak A a potvrdíme. Následuje výzva Center of circle pro střed a Radius <x> (Diameter) pro poloměr (průměr) kružnice. I tuto kružnici uvidíme pouze v jiném pohledu. Nejlépe je znatelná v pohledu Perspective.

### 3.5 Oblouk

Oblouky nám většinou spojují dvě čáry nebo mohou prodloužit již stávající oblouk do jiného směru. V programu Rhinoceros je najdeme v roletovém menu pod položkou Curve/Arc a Curve/Extend.



Obr. 25: Oblouky

Center, Start, Angle – znázorňuje oblouk zadaný středem, počátečním bodem a koncovým bodem nebo úhlem. Z příkazového řádku se spouští pomocí slova Arc. Poté následují výzvy Center of arc, kde zadáváme střed oblouku. Další je začátek oblouku na výzvu Start of arc a poslední je End point of angle, kde můžeme umístit buď koncový bod nebo určit směr pohybem myši a vypsát velikost úhlu potvrzenou Entrem.



Start, End, Direction – oblouk vykreslený od počátečního do koncového bodu pod určitým směrem. Po spuštění příkazu ArcDir již máme následující výzvy. Start of arc, End of arc pro začátek a konec oblouku a nakonec Direction, kde určíme myší směr oblouku.



Tangent, Tangent, Radius – tentokrát oblouk omezují dva tečné objekty a je určen poloměrem. Spustí se příkazem ArcTTR a začneme prvním výběrem oblouku, čáry, kružnice nebo křivky na výzvu Select first curve for ArcTTR. Pokračujeme druhým výběrem z možností při výzvě Select second curve for ArcTTR. Poslední výzvou je Radius <x>, kterým zadáme poloměr oblouku a vytvarujeme myší do požadovaného tvaru.



3 Points – oblouk zadaný třemi body. Z příkazového řádku jej spustíme za pomoci



zkratky Arc3Pt. Na výzvy Start of arc, End of arc a Point on arc vybíráme začáteční bod, koncový bod a bod uvnitř oblouku.

By Arc – zobrazí prodloužení oblouku pod určitým úhlem. Získáme jej pouze příkazem ExtendByArc nebo nabídkou z roletového menu a má následující výzvy. Select curve to extend (Join=Yes) pro vybrání oblouku. Radius of arc (MatchRadius) můžeme určit číselnou hodnotou nebo kliknutím myši a nakonec máme End point or angle, který udává konec nového oblouku.

By Arc to Point – prodloužení oblouku do bodu. Opět pro tento příkaz nemáme tlačítko na paletě, proto se spouští z roletového menu nebo příkazem ExtendByArcToPt. Následují pouze dvě výzvy a to pro vybrání začátku Select curve to extend (Join=Yes) a konce oblouku End of arc.

### 3.6 Elipsa a pravidelné n-úhelníky

Elipsy a pravidelné n-úhelníky má program rozdělen do více kategorií. Z roletového menu to jsou hned tři možnosti výběru a to Curve/Ellipse, Curve/Polygon a Curve/Rectangle. Každý je již podle názvu určen do jaké skupiny patří.

#### 3.6.1 Elipsa



Obr. 26: Kaskáda elipsy

From Center – je elipsa zadaná ve svém původním stavu. To znamená středem a



koncovými body os. Spouští se příkazem Ellipse a následují tři výzvy. První je Center of ellipse (Deformable Vertical AroundCurve FromFoci), kde vybíráme střed elipsy. Následují další dvě navzájem podobné výzvy pro výběr konce první End of first axis a druhé osy End of second axis.

Diameter – zadaná dvěma koncovými body první osy a jedním koncovým bodem



druhé osy. Příkaz EllipseD nám zobrazí výzvu Start of first axis (Vertical)



pro začátek první osy. Poté se zobrazí End of first axis pro konec první osy a pak stačí na výzvu End of second axis zadat konec druhé osy.

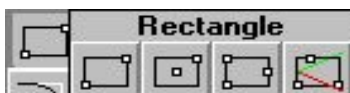
From Foci – elipsa zadaná dvěma ohnisky a bodem na elipse. První dvě výzvy jsou




pro zadání ohnisek Ellipse focus (MarkFoci=No Partial=No Repeat=No) a Other ellipse focus (MarkFoci=No Partial=No Repeat=No). Poslední výzva je pro libovolný bod na elipse Point on ellipse (MarkFoci=No Partial=No). Na tuto funkci se dostanu i pomocí písmena F zadané v elipse pomocí středu a koncových bodů obou os.


Další volby, které se vyskytly v průběhu modelování elips si vysvětlíme nyní. Pokud máme v závorce Vertical, značí nám po zadání písmena V, že vzniklá elipsa bude kolmá ke konstrukční rovině. Dále je možná volba ArounCurve s písmenem A pro elipsu, jejíž osa je kolmá k zadané křivce. A jak již bylo výše naznačeno písmeno F (FromFoci) nás přesměruje na modelování elipsy pomocí ohnisek.

### 3.6.2 Obdélník





Obr. 27: Obdélník

Corner to Corner – obdélník zadaný úhlopříčnými rohy. Z příkazového řádku se  spouští pomocí slova Rectangle. Po vyhodnocení příkazu následuje první výzva First corner of rectangle (Rounded) pro zadání prvního rohového bodu. Následuje Other corner or length, kde si můžeme vybrat buď přímo zadaný bod a tím nám vznikne obdélník nebo zadáme délku a přesuneme se na další výzvu Width, kde zadáme šířku obdélníku. Můžeme zadávat i záporné hodnoty. Záleží pouze na tom, jak má vypadat výsledný obdélník.

Center, Corner – obdélník vykreslený za pomoci středu, výšky a šířky. Po zadání  příkazu RectangleCen se zobrazí výzva na zadání středu obdélníku Center of rectangle (Rounded). Pokračujeme výzvou Corner of length, kde máme opět dvě

možnosti pro pokračování. První je za pomoci myši vytvořit daný obdélník nebo druhý případ, podobný z předchozího vysvětlení zadat číselnou hodnotu pro šířku a pokračovat výzvou Width pro zadání výšky obdélníku.

3 Points – tento obdélník vznikne pomocí dvou sousedních rohů a bodu na protější straně. Z příkazového řádku se spustí pomocí Rectangle3Pt. Nejdříve se  postupně zadají dva body na jedné straně, které jsou uvozeny výzvami Start of edge a End of edge. Poslední výzva Width značí výšku obdélníku.


Vertical – obdélník kolmý na konstrukční rovinu zadany třemi body. Popis konstrukce je uveden již výše. Jediný rozdíl je v tom, že nebude vidět  v rýsovacím pohledu nýbrž v pohledu jiném.

Pro funkci obdélník máme pouze jednu rozšiřující nabídku. Značí ji slovo Rounded uvedené u výzev v závorce. Značí ji slovo Rounded uvedené u výzev v závorce. Spustí se písmenem R a značí zaoblení rohů. Při tom se nám nakonec přidá výzva právě pro určení poloměru zaoblení rohu, která zní Radius or point for rounded corner to pass through (Corner=Arc).

### 3.6.3 Pravidelný n-úhelník a čtverec



Obr. 28: N-úhelníky a čtverec

Center, Radius – vytvoří n-úhelník zadany středem a poloměrem. Po spuštění příkazu Polygon v příkazovém řádku se objeví následující výzvy. Center of  inscribed polygon (NumSides=x Circumscribed Vertical AroundCurve), kde můžeme změnit číslem počet úhlů n-úhelníka (NumSides). Pokud souhlasíme s úhly, zadáváme střed. Další výzva je Corner of polygon. Ta určuje vrchol n-úhelníka. Buď si ji můžeme rovnou určit myší nebo si zadáme velikost poloměru a výzva se nám zopakuje pro natočení a přesné umístění bodu v dané vzdálenosti.

By Edge – n-úhelník zadaný koncovými body jedné strany. Zkratka pro spuštění z příkazového řádku je PolygonEdge. Začíná výzvou Start of edge (NumSides=x Vertical) pro zadání vrcholu n-úhleníka. Opět si můžeme změnit počet úhlů. Pokračujeme výzvou End of edge (Flip), kde vybíráme další vrchol na téže straně n-úhleníka nebo opět můžeme zadat tentokrát velikost hrany a umístit jeho pozici.

Čtverce vykreslujeme stejným způsobem jako u n-úhleníků. Můžeme mít zadaný střed a vrchol nebo dva body na stejné straně. Stačí když budeme mít položku NumSides rovnu čtyřem.

Většinu voleb pro n-úhleníky již známe z dřívějšíka. Vertical (V) označuje objekt kolmý na konstrukční rovinu. ArounCurve (A) je pro změnu kolmý na zadanou křivku. Poprvé se nám zde ale objevuje Circumscribed. Zkratkou je písmeno C a význam spočívá v tom, že n-úhleník je opsaný danému poloměru. Výchozí nastavení znázorňuje n-úhleník vepsaný.


### 3.7 Tělesa


Tělesa jsou jedním ze stavebních kamenů trojrozměrného pole. Je pro ně typická špatná viditelnost hran při různých polohách. Proto se vytváří spousta programů, kde jedním z nich je i program Rhinoceros. Tělesa najdeme pod položkou Solid v roletovém menu.





Obr. 29: Tělesa


Corner to Corner, Height – pro zobrazení čtyřbokého hranolu pomocí úhlopříčných bodů podstavy a výšky. Spouští se příkazem Box a má následující výzvy. First corner of box, kde zadáváme první bod podstavy. Other corner or length znamená zadání úhlopříčného bodu podstavy nebo délku pro podstavu. Pokud jsme zadali délku, zobrazí se další výzva pro určení šířky podstavy Width. Poté již stačí na výzvu Height určit výšku pro vznik nového hranolu.


3 Points, Height – další zobrazení čtyřbokého hranolu tentokrát za pomoci 3 bodů podstavy (2 koncové body jedné hrany a bod na protější hraně) a výšky.  Z příkazového řádku se spustí po zadání Box3Pt. První dvě výzvy jsou pro zadání koncových bodů hrany Start of edge a End of edge. Následuje Width pro určení bodu na protější hraně a poslední výzva je Height pro znázornění výšky hranolu.

Center, Radius – pro znázornění koule za pomoci středu a poloměru. Po napsání příkazu Sphere se zobrazí první výzva Center of sphere pro zadání středu koule.  Následuje Radius <x> (Diameter) pro určení poloměru. Pokud zadáme písmeno D, změní se nám poloměr na průměr koule.

Diameter – zadání koule pouze dvěma body, které zároveň určují průměr dané koule. Spustí se příkazem SphereD. Má pouze dvě výzvy, kde první Start of diameter určuje jeden bod na plášti koule a druhá End of diameter je pro určení druhého bodu též na plášti koule a zároveň leží na ose. 

3 Points – další možný způsob pro znázornění koule je pomocí 3 bodů na plášti.  Z příkazového řádku jej spustíme Sphere3Pt a výzvy jsou pro postupné zadání bodů. Zní First point on sphere, Second point on sphere a Third point on sphere.

Cylinder – určení válce za pomoci středu, průměru a výšky. Ke spuštění slouží slovo Cylinder a začíná výzvou Base of cylinder (Vertical) pro určení středu podstavy. Pokračujeme Diameter <x> (Radius), kde zadáváme průměr podstavy a ukončíme si funkci výzvou End of cylinder. Ta se nás dotazuje na výšku válce. Pokud bychom chtěli podstavnu plochu zadat pomocí průměru, stačí na výzvu Diameter <x> (Radius) potvrdit písmeno R. 

Cone – jehlan určený středem podstavy, poloměrem a výškou. Po zadání příkazu Cone se zobrazí výzva Base of cone (Vertical) pro určení středu. Pokračujeme Radius <x> (Diameter), kde si vybereme poloměr podstavy. A celý příkaz se ukončí výzvou Height, která určuje výšku celého jehlanu. V průběhu se nám objevily 

dvě volby. První je při zadání středu jehlanu, kde máme určeno Vertical. Po potvrzení písmena V bychom rýsovali jehlan kolmý ke konstrukční rovině. Další volbou je Diameter při zadávání poloměru podstavy. Pokud napíšeme D změní se poloměr na průměr.

### 3.8 Stínování

Pro lepší přehlednost jsou zde připraveny dvě funkce pro vystínování. Jak již bylo řečeno, dá se stínování přizpůsobit našim požadavkům v nastavení.

Shade – je monochromaticky vystínováno. Tím se získá představa o objemu. Model



Obr. 30: Kaskáda Shade

je vystínován přímo v okně pohledu a dá se s ním manipulovat jako s drátovým modelem. Pomocí různých tlačítek můžeme vystínovat pouze jeden pohled nebo všechny najednou. Též si můžeme vytvořit stínovaný model pouze jednoho objektu.

Render – je též stínování jako v Shade. Rozdíl je v tom, že zde jsou již použity pro

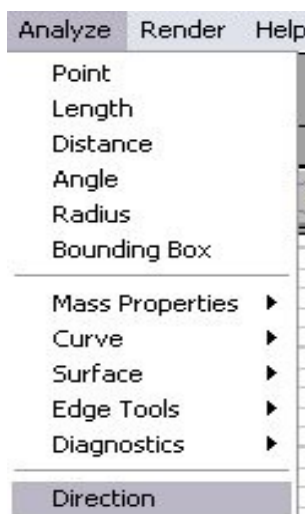


Obr. 31: Kaskáda Render

stínování barvy, které jsme si přidělili a vystínované objekty se otevřou v novém okně. S vyrenderovaným obrázkem již nemůžeme manipulovat jako s předchozím stínováním, ale můžeme si jej uložit do souboru.

### 3.9 Analytické příkazy

V určitých případech potřebujeme získat různé údaje. I v programu Rhinoceros na toto téma mysleli a dodali pár užitečných funkcí. V roletovém menu je umístili pod záložkou Analyze.



Distance – změří vzdálenost mezi dvěma body. Spustí se příkazem Distance a má dvě výzvy. První zní First point for distance, pro určení jednoho bodu a následuje Second point for distance pro druhý bod. Poté se v příkazovém řádku objeví informace o zadaných bodech a nakonec vypíše vzdálenost.

Obr. 32: Analytické příkazy

Length – určí délku zadaného objektu. Dá se spustit příkazem Length a je pouze jedna výzva pro určení objektu Select curve to measure. Po označení opět vypíše hodnotu do příkazového řádku.

Angle – slouží pro zjištění úhlu svírajícího dvěma čarami. Po napsání příkazu Angle se objeví první výzva Start of first line, kde určíme začátek první čáry. Pokračujeme koncem první čáry výzvou End of first line. Dále se objeví výzvy pro začátek a konec druhé čáry Start of second line a End of second line. Pak již čekáme na zpracování výsledku v příkazovém řádku.

Radius – určuje poloměr křivosti křivky, kružnice nebo oblouku. Stačí zadat funkci Radius a objeví se jediná výzva Select point on curve for radius measurement. Daný poloměr se zobrazí v kolonce příkazového řádku.

Point – zjišťuje souřadnice bodu. Z příkazového řádku se spouští pomocí příkazu EvaluatePt. Je zde jediná výzva Point to measure, a to na určení požadovaného bodu. Po určení bodu se dané souřadnice vypíší do příkazového řádku.

## 4. Deskriptivní geometrie

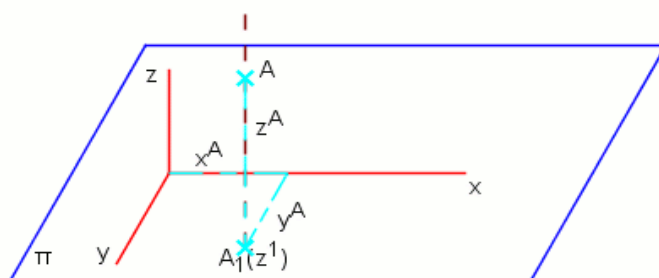
Jak již název napovídá, deskriptivní geometrie je částí širšího vědního oboru geometrie a tím se zařazuje do matematických věd. Matematika ve svém celku, kromě bezprostředního úkolu hledat řešení předložených praktických problémů, svou celou výstavbou a svými metodami neobyčejnou měrou přispívá k rozvoji logického myšlení a prostorové představivosti.

Deskriptivní geometrie je věda o zobrazení prostorových útvarů do roviny. Její podstatou je jednoznačný vztah mezi zobrazovaným objektem a jeho průmětem, proto je uplatnění metod deskriptivní geometrie v technické praxi široké.

Některé prvky deskriptivní geometrie, pojaté jako nauka o zobrazovacích metodách, byly známy již před naším letopočtem. Ovšem za zakladatele deskriptivní geometrie je považován Gaspard Monge (1746-1818), který byl francouzským matematikem, fyzikem a geometrem. V roce 1799 popsal v knize „Géométrie descriptive“ známé pravoúhlé promítání na dvě k sobě kolmé průmětny. Tímto dílem byl dán podklad k rozvoji deskriptivní geometrie.

### 4.1 Kótované promítání

Kótované promítání se využívá především ve stavebním inženýrství a pozemním stavitelství. Základem tohoto promítání je pravoúhlé promítání na jednu průmětnu. Zobrazující metoda je promítnutí z třírozměrného prostoru do dvourozměrné roviny s připojením kót.

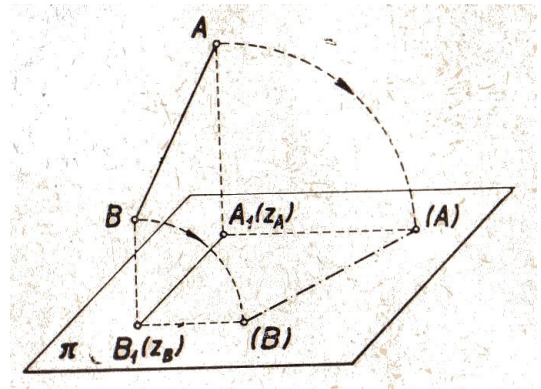


Obr. 33: Kótované promítání

### 4.1.1 Sklápění roviny

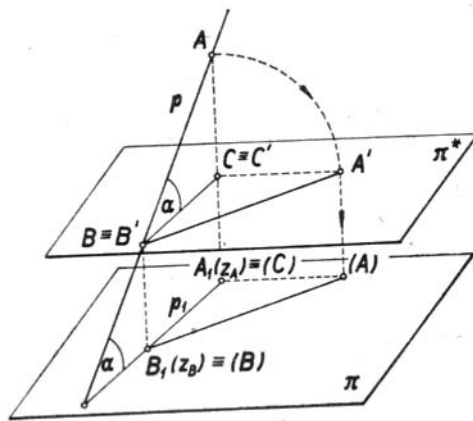
Často potřebujeme řešit úlohy, kde je nejdříve potřeba vytvořit daný obraz ve skutečné velikosti. K tomu slouží řešení v dané rovině. Pokud je rovina rovnoběžná s průmětnou nebo dokonce splývá, pak máme dané útvary ve skutečné velikosti. Jestliže však rovina není hlavní, máme jejich průměty zkreslené. Proto nejdříve musíme danou rovinu otočit do průmětny nebo do hlavní roviny. Teprve potom můžeme úlohu dořešit.

Otáčení promítací roviny do průmětny se nazývá sklápění roviny do průmětny. Jako při každém otáčení či sklápění musíme nejdřív určit osu otáčení (sklápění). Osou je průsečnice dané roviny s průmětnou, tj. její stopa. Poté již stačí sklápět libovolné body, které neleží v průmětně.



Obr. 34: Sklápění do průmětny

Místo do průmětny lze také sklápět do hlavní roviny. Promítací rovinu úsečky AB nesklopíme do průmětny, ale do hlavní roviny, která prochází jedním z bodů a je rovnoběžná s průmětnou.



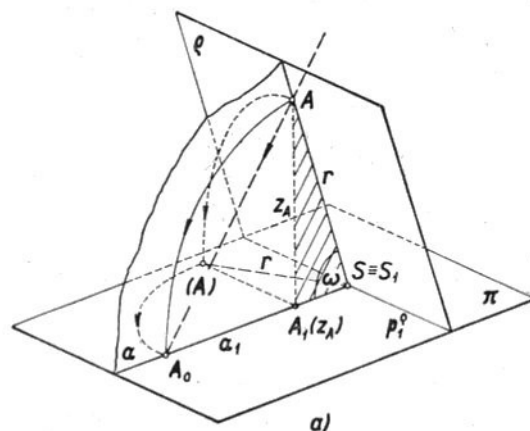
Obr. 35: Sklápění do hlavní roviny

Užitím sklápění můžeme též nalézt úhel přímky s průmětnou. Jeho velikost se nazývá odchylka přímky od průmětny.



### 4.1.2 Otáčení roviny

O dané rovině předpokládáme, že není promítací ani hlavní. Otáčení roviny znamená najít osu otáčení a pak stačí otočit její libovolný bod, který neleží na ose otáčení. Osou otáčení je podobně jako osou sklápění průsečnice roviny s průmětnou. Otáčíme libovolný bod, který neleží na ose tím, že postupně určíme rovinu otáčení, střed otáčení a nakonec poloměr otáčení.

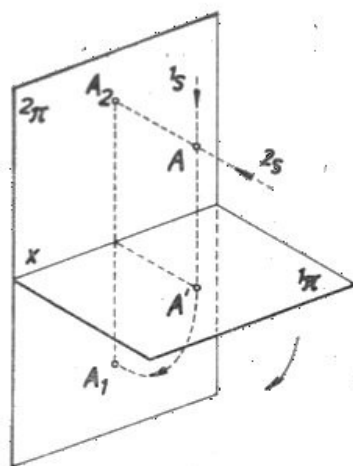


Obr. 36: Otáčení roviny

Rovina otáčení bodu je kolmá k ose otáčení. Střed otáčení bodu je průsečík roviny otáčení bodu s osou otáčení. Poloměr otáčení bodu je jeho vzdálenost od středu otáčení.

### 4.2 Mongeovo promítání

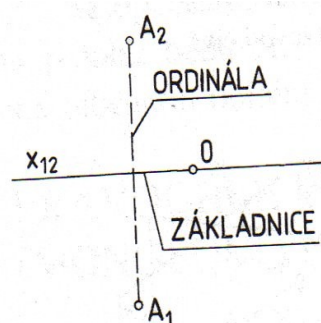
Nejjednodušší a nejběžnější zobrazení v deskriptivní geometrii je pravoúhlé promítání na dvě k sobě kolmé průmětny, které se nazývá Mongeovo promítání.



Obr. 37: Mongeovo promítání

### 4.2.1 Bod

Základním principem zobrazení je dvojice sdružených průmětů, které leží na kolmici k ose  $x$ . Sdružené průměty tvoří uspořádané dvojice, kde první bod je otočený první průmět a druhý bod je druhý průmět bodu. Osu  $x$  nazýváme základnice a kolmici k základnici říkáme ordinála. Sdružené průměty bodu tudíž leží na ordinále.

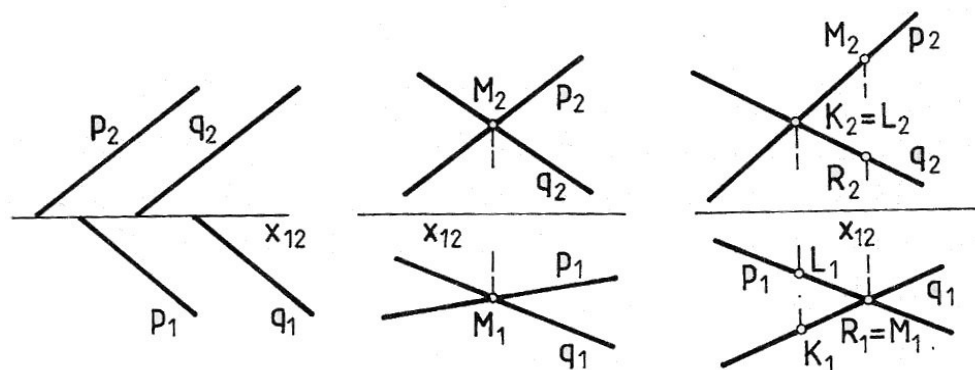


Obr. 38: Základní zobrazení bodu

### 4.2.2 Přímka

Přímka je v prostoru určena dvěma body. Sdruženými průměty těchto bodů jsou určeny sdružené průměty přímky, která těmito body prochází. Při zobrazování přímek se rozlišují dva případy a to přímka kolmá k základnici nebo přímka, která není kolmá k základnici. Stopník přímky je průsečík přímky s průmětnou. Rozlišujeme půdorysný stopník pro průsečík přímky s první průmětnou a nárysný stopník je průsečík s druhou průmětnou.

Dvě přímky v prostoru mohou být vzájemně rovnoběžné, různoběžné nebo mimoběžné. Vše se pozná podle jejich průmětů. Rovnoběžné přímky mají souhlasné průměty rovnoběžné. Průsečíky souhlasných průmětů různoběžných přímek leží na ordinále a průsečíky souhlasných průmětů mimoběžných přímek na ordinále neleží.

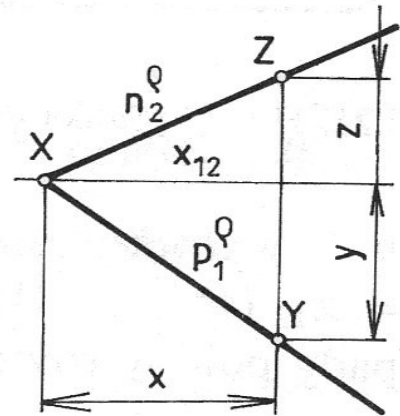


Obr. 39: Vzájemná poloha přímek

### 4.2.3 Rovina

V Mongeově promítání může být rovina určena více druhy. Jednou z možností je pomocí tří nekolineárních bodů, přímkou a bodem, který přímkce nenáleží, nebo dvěma různoběžkami či rovnoběžkami. Průmětem roviny je buď přímka nebo celá průmětna.

Stopa roviny je průsečnice roviny s průmětnou. Opět jako u přímek rozlišujeme půdorysnou a nárysnu stopu.

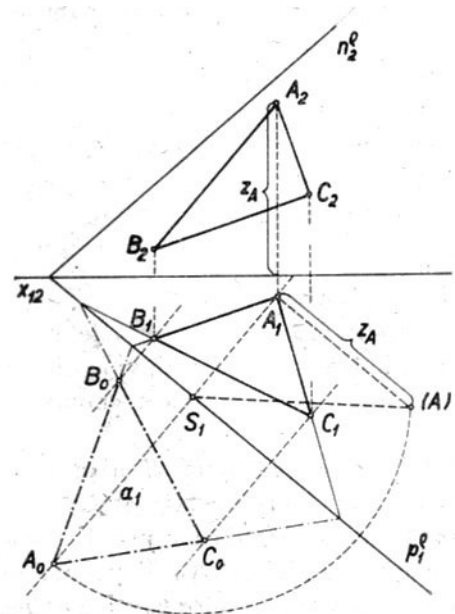


Obr. 40: Stopa roviny

U vzájemné polohy rovin rozlišujeme rovnoběžné různé nebo totožné a různoběžné. Dvě různé rovnoběžné roviny nemají žádný společný bod. Jejich průměty jsou navzájem rovnoběžné. Různoběžné roviny mají společný bod a zároveň i společnou přímku, která prochází tímto bodem. Společná přímka se nazývá průsečnice.

### 4.2.4 Otáčení roviny

Cílem při otáčení roviny je buď najít skutečnou velikost rovinného útvaru, nebo najít sdružené průměty daného rovinného útvaru. Rovinu můžeme otáčet přímo do průmětny, nebo do roviny rovnoběžné s nějakou průmětnou. Při otáčení v Mongeově promítání vycházíme z již zmiňovaného kótovaného promítání. Dřívější číselnou kótu nyní nahrazuje grafická.

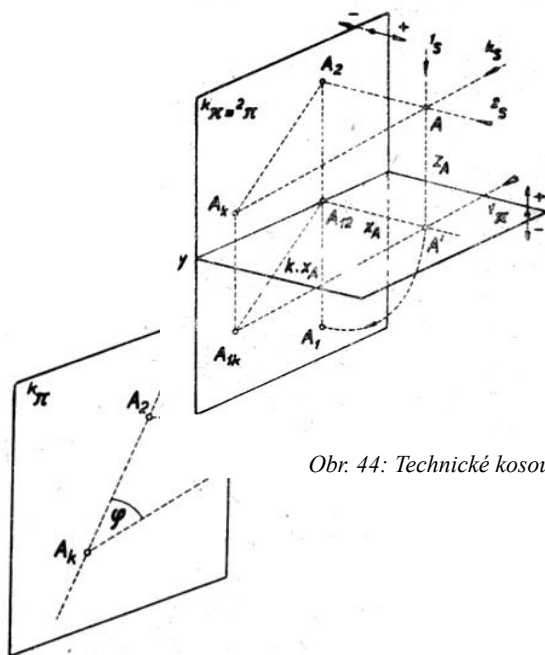


Obr. 41: Otáčení roviny

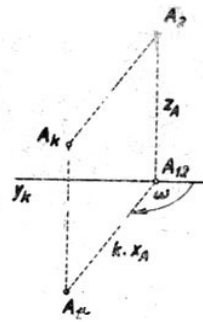
### 4.3 Kosoúhlé promítání

Kosoúhlé promítání se užívá k názornému zobrazování technických předmětů. Základem kosoúhlého zobrazení je promítání na jednu průmětnu. Pokud bychom jej chtěli využít jako vzájemně jednoznačné zobrazení mezi vzorem a obrazem, pak průmět doplňujeme dalším průmětem. Může to být pravoúhlý průmět do téže průmětny nebo pravoúhlý průmět do pomocné průmětny. V praxi se většinou obě možnosti kombinují a vzniká tak technické kosoúhlé promítání, které je určeno úhlem a poměrem zkrácení.

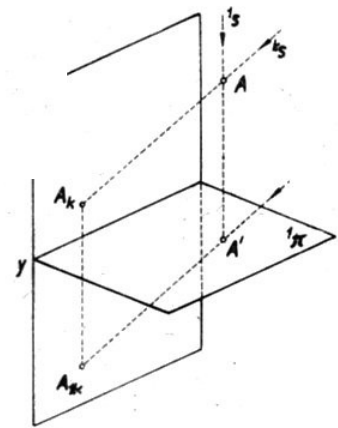
V kosoúhlém promítání zůstávají vždy dvě souřadnice bodu nezměněné a třetí se mění v závislosti poměru zkrácení.



Obr. 42: Kosoúhlé promítání na jednu průmětnu



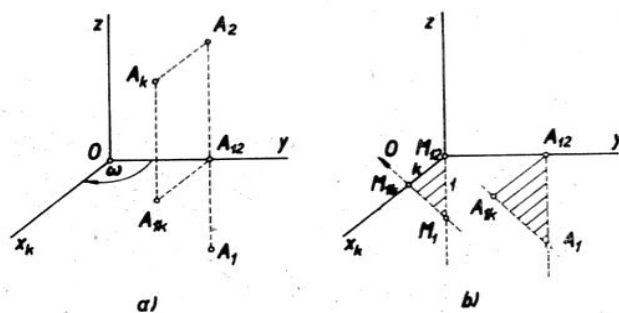
Obr. 44: Technické kosoúhlé promítání



Obr. 43: Kosoúhlý průmět do pomocné průmětny

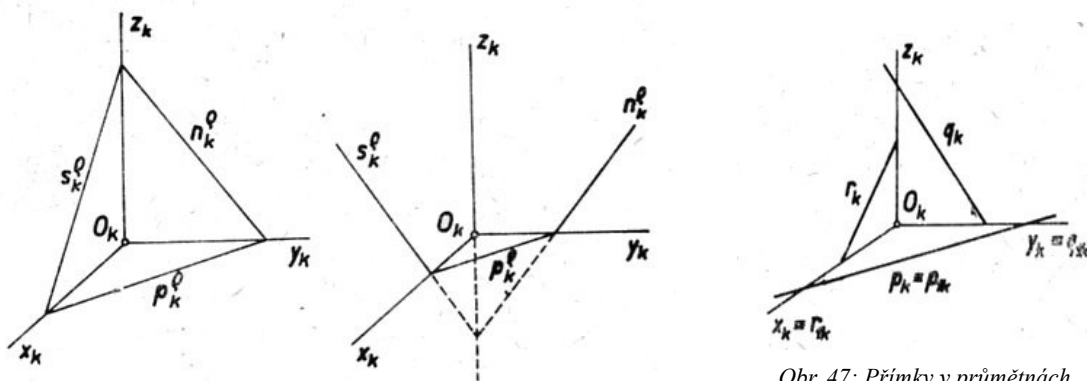
### 4.3.1 Bod, přímka a rovina

Kosoúhlý průmět nějakého bodu určíme tak, že zobrazíme jeho souřadnicový kvádr. Délky úseček od počátku směrem na osy  $x$  a  $z$  se promítáním nemění, jde o úsečky v průmětně. Délka úsečky od počátku směrem k ose  $y$  se mění daným koeficientem. Koeficientem je myšleno poměr zkrácení.



Obr. 45: Kosoúhlý průmět bodu

Pokud víme, jak zobrazit v tomto zobrazení bod, můžeme nyní přistoupit k zobrazování přímky a roviny. Stačí si pouze uvědomit, čím je přímka a rovina zadána. Přímku můžeme sestavit tehdy, pokud známe, nebo si zvolíme dva libovolné body ležící na přímce. Potom postupujeme v konstrukci stejně jako u bodu. Podobné je to i u roviny, která je zadána buď bodem a přímkou nebo třemi body.



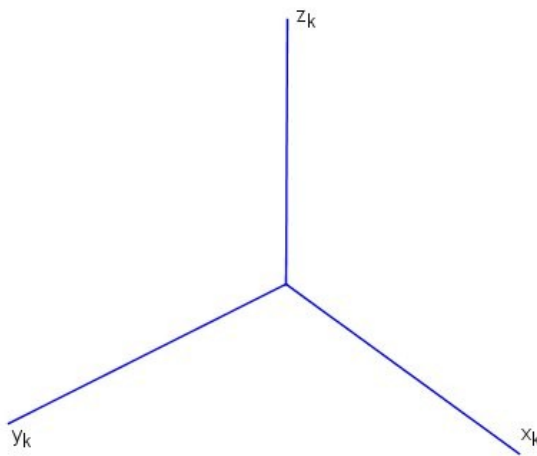
Obr. 46: Zobrazení rovin

Obr. 47: Přímky v průmětnách

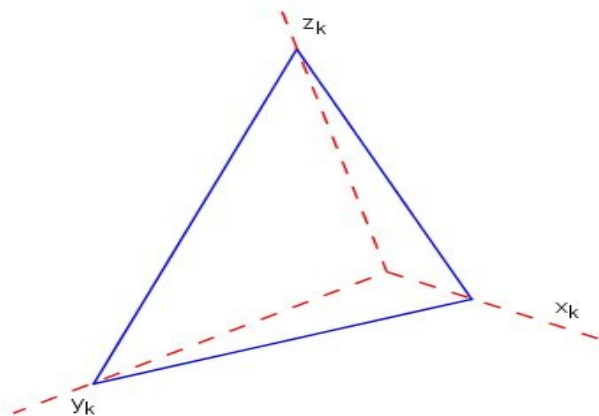
## 4.4 Pravoúhlá axonometrie

Toto zobrazení používáme především k názornému zobrazení malých strojních součástí. V pravoúhlé axonometrie lze sestavit snadno názorné obrazy bez velkých obtíží.

Pravoúhlá axonometrie je zobrazení na dvě vzájemně kosé průmětny. Je zadána osovým křížem nebo axonometrickým trojúhelníkem, který musí být ostroúhlý. Axonometrický průmět osového kříže je tvořen výškami axonometrického trojúhelníka.



Obr. 48: Osový kříž pravoúhlé axonometrie

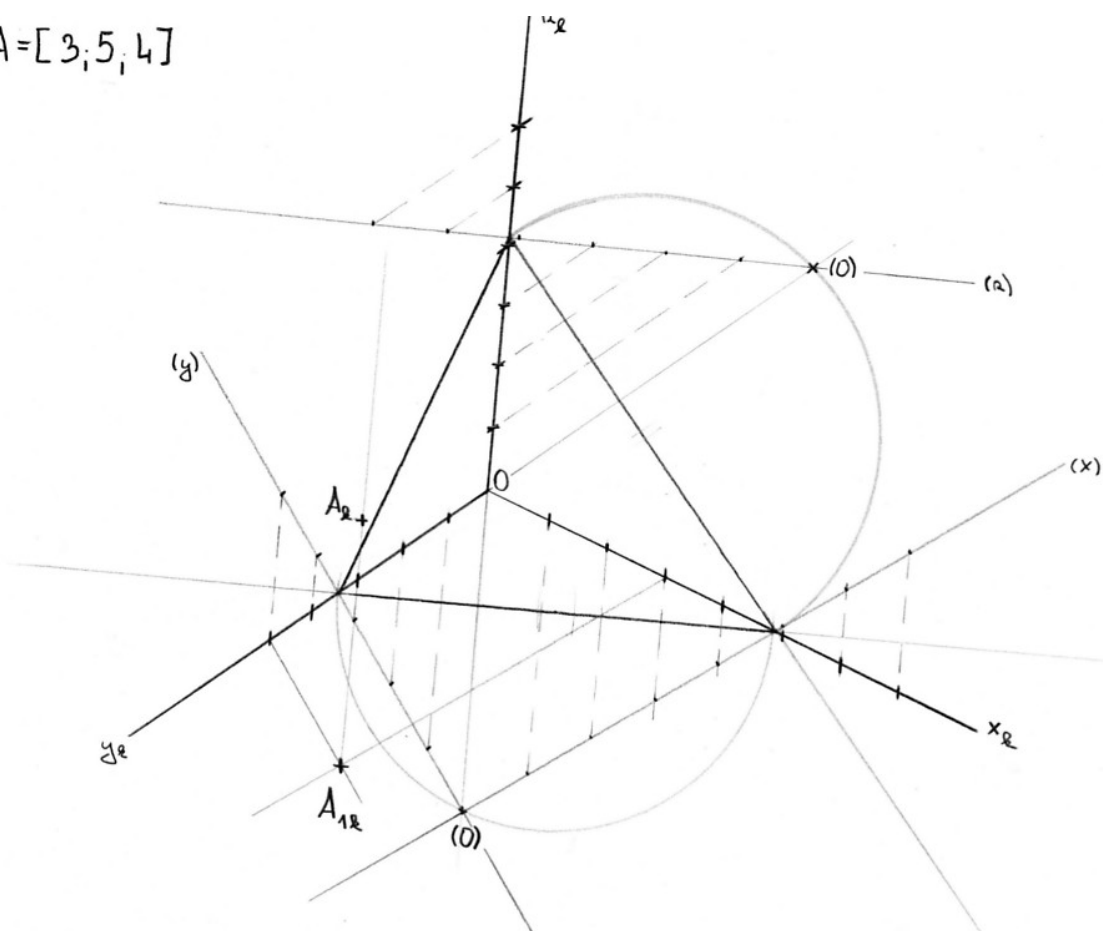


Obr. 49: Axonometrický trojúhelník

#### 4.4.1 Zobrazení bodu, přímky a roviny

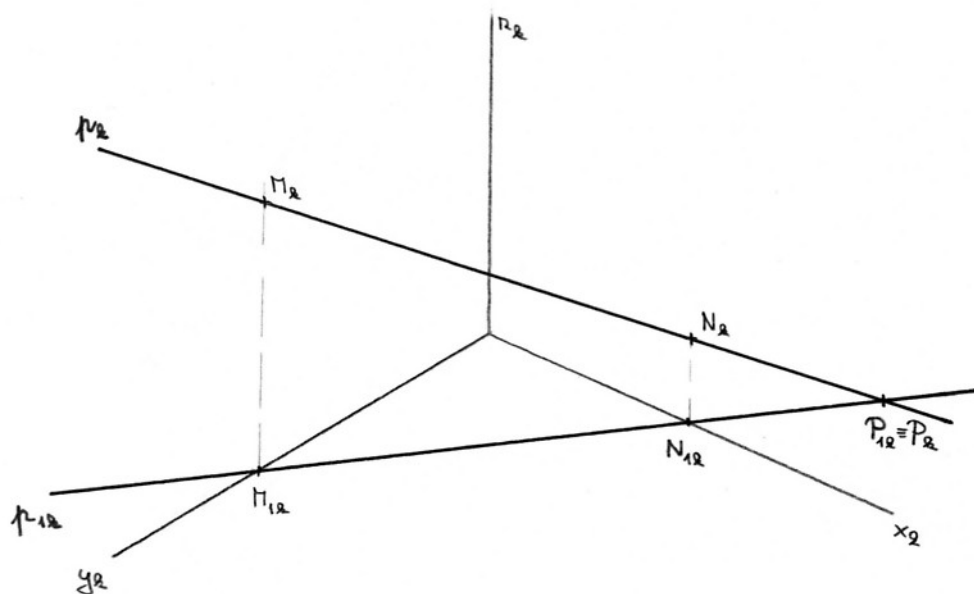
Pravouhlá axonometrie při zobrazování bodu je velice podobná s kosoúhlým promítáním. Nejdříve si ovšem musíme určit axonometrické jednotky na osách, které se nám mohou lišit na všech osách. Pokud je máme vyřešený, můžeme zadávat body.

$$A = [3; 5; 4]$$



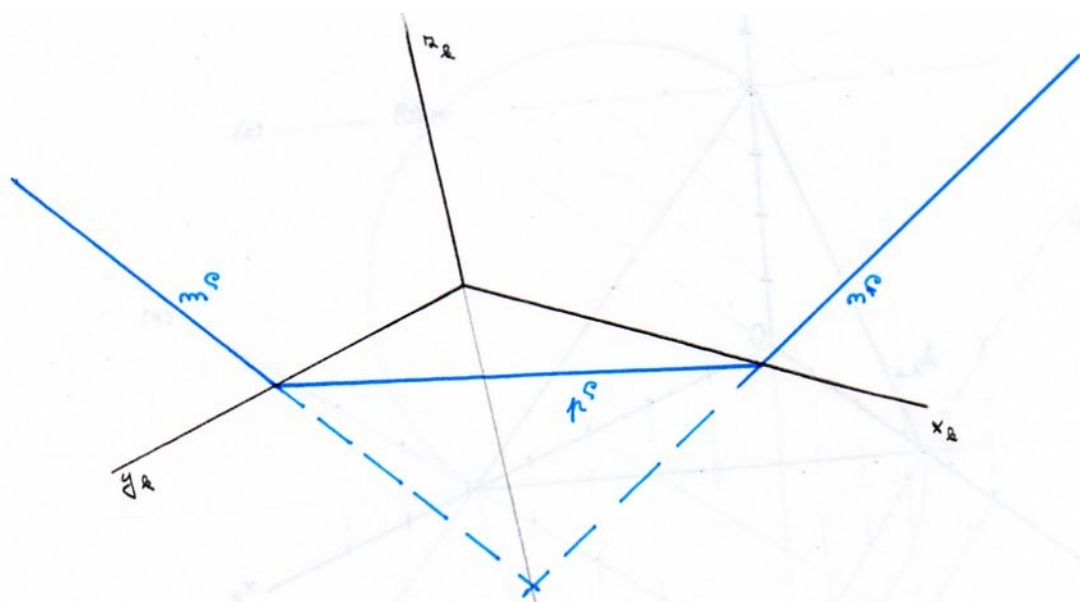
Obr. 50: Určení axonometrických jednotek a zobrazení bodu A

Jestliže již umíme určovat body v pravouhlé axonometrii, nic nám nebrání v zobrazování přímek. Ty jsou vždy určeny dvěma body. U přímek můžeme určovat stopníky. Rozlišujeme půdorysný (P), nárysný (N) a bokorysný (M) stopník.



Obr. 51: Přímka a její stopníky

Pokud máme pravoúhlu axonometrii zadanou pomocí axonometrického trojúhelníka, tak již máme určenou i rovinu. Strany trojúhelníka tvoří rovinu. Dalším způsobem zadání roviny je určení průsečíků s osovým křížem pravoúhlé axonometrie.



Obr. 52: Rovina zadaná body na osách



## 5. Příklady

Nyní se zaměříme na vypracování některých příkladů pomocí programu Rhinoceros. Zjistíme, zda nám daný program při konstrukčních úlohách usnadní práci nebo naopak nebude užitečný.

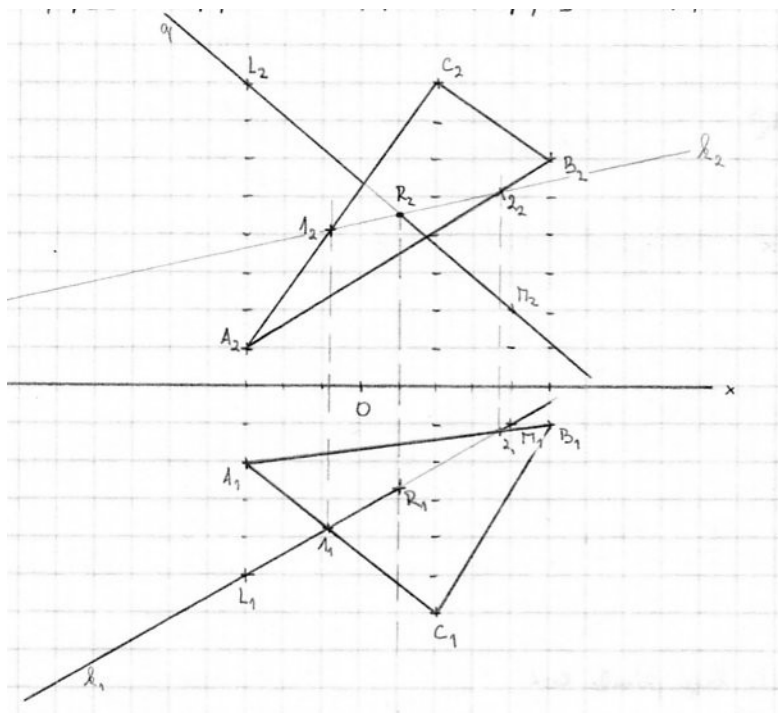
U každého příkladu rozeberu postup klasického rýsování na papír. Dále se budu zabývat rýsováním v programu Rhinoceros, o kterém pojednává má diplomová práce. Na závěr každého příkladu vyhodnotím klady a zápory obou těchto technik.

### 5.1 Rovinné obrazce

**Příklad 1:** Zobrazte průsečík přímky  $LM$  s rovinou  $\rho$ , která je dána body  $ABC$ .

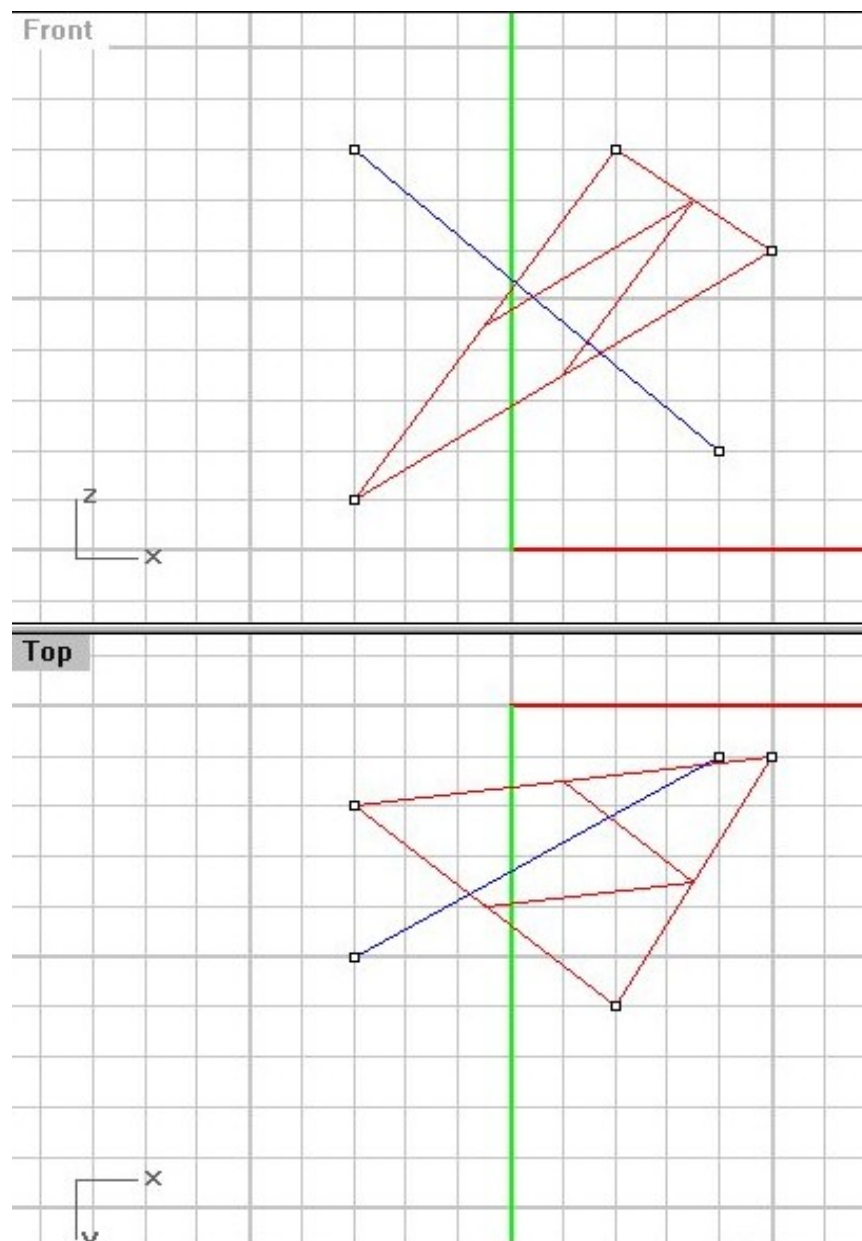
$$L=[-3;5;8] \quad M=[4;1;2] \quad A=[-3;2;1] \quad B=[5;1;6] \quad C=[2;6;8]$$

Nejprve si narýsujeme základnici a vyneseme všechny x-ové, y-ové a z-ové souřadnice všech zadaných bodů. Dalším bodem v postupu rýsování je spojení bodů roviny, které tvoří skupina bodů  $A_1B_1C_1$ , které leží v půdorysně a body  $A_2B_2C_2$ , které leží v nárysně. Současně musím vymežit přímku, která je dána body  $LM$ , tudíž spojím body  $L_1M_1$ , což je přímka  $g_1$  a leží v půdorysně. Body  $L_2M_2$ , tedy přímka  $g_2$ , která leží v nárysně. Nyní následuje nalezení průsečíků roviny s přímkou, které jsem si označila jako  $1_1$  a  $2_1$  v půdorysně, které se mi zobrazí přes ordinálu do náryсны, tedy body  $1_2$  a  $2_2$ . Tyto body náležejí



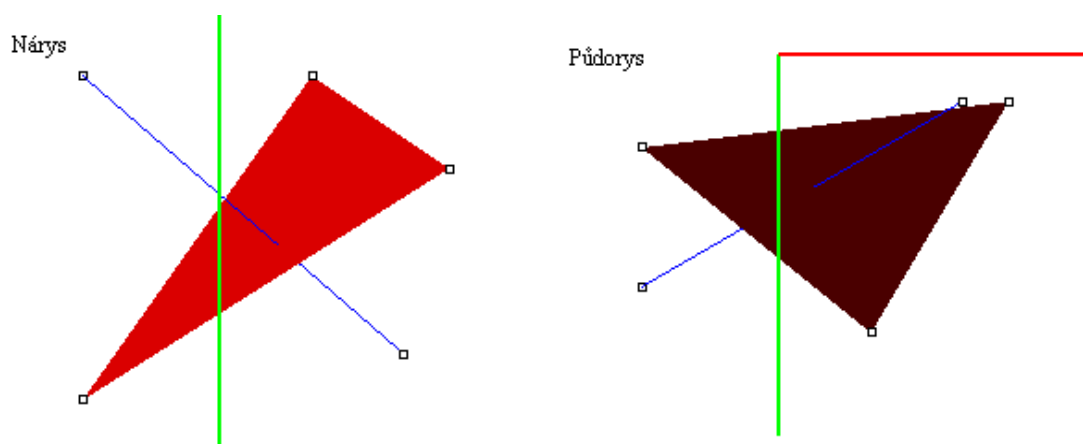
Obr. 53: Klasické rýsování - přímka a rovina

rovině dané body  $A_2B_2C_2$  v nárysně. Průsečíky jsem volila v půdorysně, tedy kde se mi přímka  $g_1$  protíná s rovinou danou body  $A_1B_1C_1$ . Přímka  $g_1$  se mi zobrazuje do pomocné přímky  $k_1$ , tedy jsou totožné. V nárysně spojíme body  $1_2$  a  $2_2$ , které nám vznikly přenesením přes ordinálu, tudíž tvoří přímku  $k_2$ . Kde se nám přímka  $k_2$  protne s přímku  $g_2$ , vznikne průsečík roviny s přímku, nazveme ho  $R_2$ . Tento bod přeneseme do půdorysny pomocí ordinály. Bod  $R_1$  náleží přímce  $g_1$  a zároveň  $k_1$  a rovině  $A_1B_1C_1$ . Poté vytáhneme viditelnost přímky a roviny v obou průmětech.



Obr. 54: Pohled Front a Top programu Rhinoceros

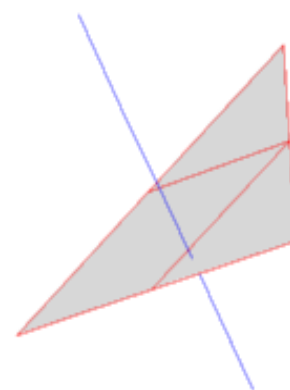
V programu Rhinoceros jsem rýsovala v perspektivním pohledu, kde jsem si v první řadě zadala body pomocí příkazu Point. Poté jsem si vyznačila roviny pomocí tří bodů ABC příkazem Surface. Dále jsem narýsovala přímku g příkazem Line za pomoci bodů L a M. Během mého rýsování v perspektivě se okamžitě zobrazuje narýsovaná rovina a přímka v pohledu Front (nárýsna) a Top (půdorys). Průsečík roviny a přímky se



*Obr. 55: Vystínovaný pohled příkladu programem se zobrazením os*  
zde řeší pouze vystínováním dané roviny, tudíž se nijak pracně nerýsuje. Zároveň i vystínováním je určena viditelnost daných objektů.

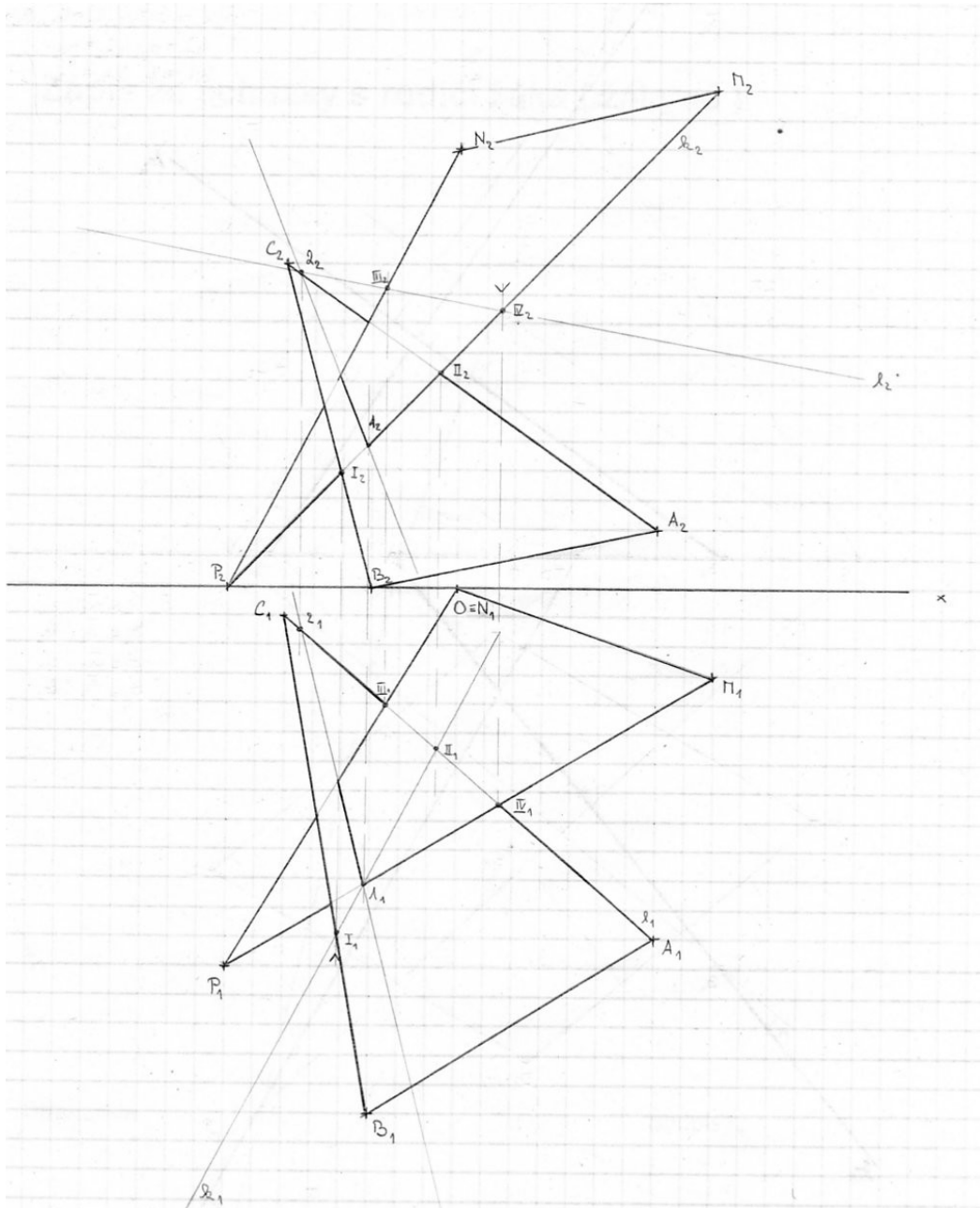
Kladem programu Rhinoceros je jednoduché rýsování v perspektivě a následné zobrazení v obou průmětnách. Dále si myslím, že je velikou výhodou názorná simulace vystínované roviny, tudíž zde nemusí být velká představivost, kde by se měl daný průsečík nalézat.

Záporem v tomto příkladu je nepřesné zobrazení průsečíku přímky a roviny v programu Rhinoceros. Myslím si, že v tomto případě průsečíku je klasické rýsování na papír přesnější.



*Obr. 56: Objekty v pohledu Perspective*

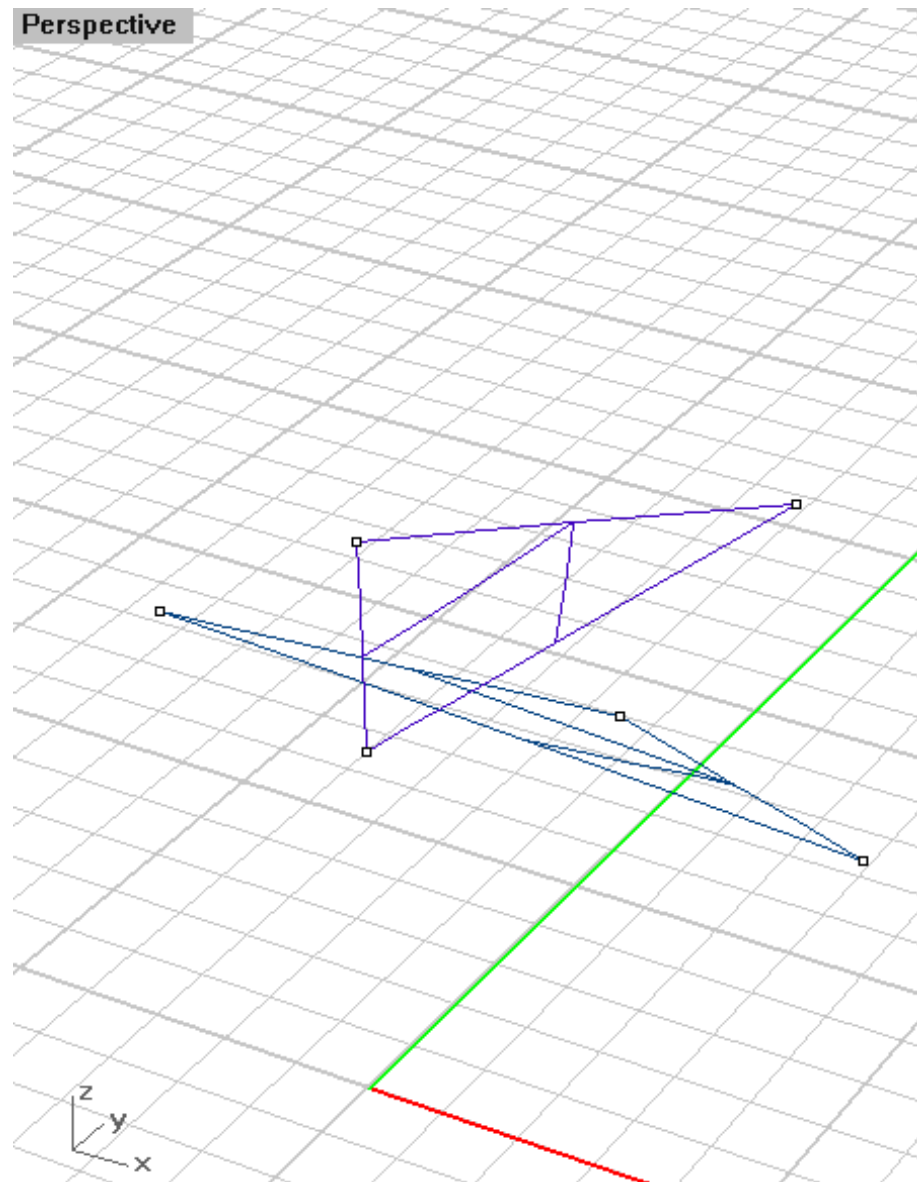
Příklad 2: Zobrazte průsek  $\triangle ABC$  s  $\triangle MNP$ .  $A=[3,5;6;1]$   $B=[-1,5;9;0]$   
 $C=[-3;0,5;5;5]$   $M=[4,5;1,5;8,5]$   $N=[0;0;7,5]$   $P=[-4;6,5;0]$



Obr. 57: Mongeovo promítání v klasickém podání

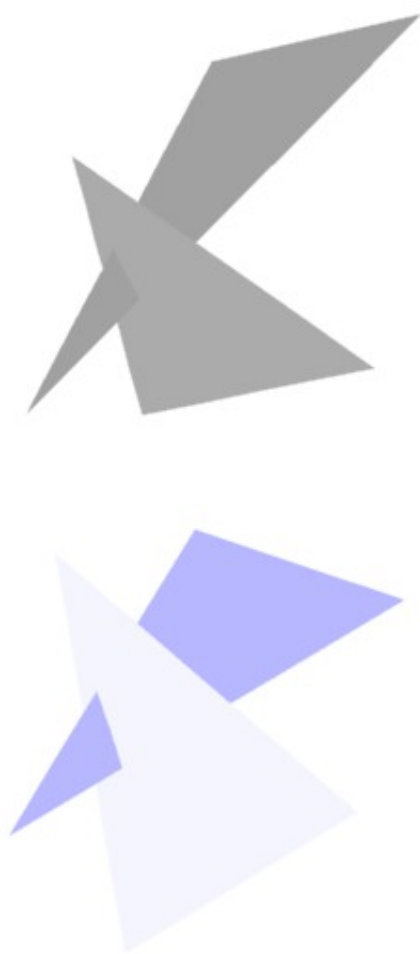
Narýsuji si základnici a zobrazím v nárysně i půdorysně zadané body. Poté spojím body odpovídající daným trojúhelníkům, tedy trojúhelník ABC a trojúhelník MNP. Tyto trojúhelníky vyznačím jak v nárysně tak i půdorysně s danými indexy. V nárysně jsem si našla body, které jsou průsečíky obou trojúhelníků a přenesla jsem je do půdorysny.

To samé jsem opakovala v půdorysně a převedla do nárýsky. Poté jsem si označila úseky, které jsou skutečnými průsečíky obou trojúhelníků a vytáhla jsem viditelnost těchto obrazců.



Obr. 58: Průsek dvou trojúhelníků

Pomocí programu jsem si opět zobrazila dané body pomocí příkazu Point v perspektivním pohledu. Dále jsem spojila body náležící daným trojúhelníkům pomocí příkazu Surface. A tímto bylo rýsování dokončeno. Pro určení průsečíku a viditelnosti stačí dané trojúhelníky vystínovat.



*Obr. 59: Vystínování v pohledech Front a Top*

Kladem je opět rychlé narýsování daného problému v perspektivním pohledu a jeho okamžitý přenos do nárysu a půdorysu. Není zde náročné hledání průsečíku jako v klasickém rýsování. Je sice pravda, že tento průsečík je nepřesný, ale pro názornou ukázkou viditelnosti postačující.

Příklad 3: Zobrazte průnik  $\Delta PQR$  s  $ABCD$ , známe-li:  $A=[2,5;4;2,5]$ ,  $B=[0;7,5;1]$ ,  $C=[-3;4;5,5]$ ,  $P=[2,5;6,5;6]$ ,  $Q=[-4;6;4,5]$  a  $R=[1;4,5;1]$

### Nárýsováním

základnice a zadaných bodů si zobrazíme zadání.

Dále si spojíme body trojúhelníku PQR

v nárýsně i půdorysně s danými indexi.

Pro rýsování

rovnoběžníku ABCD

nejprve potřebuji zjistit

souřadnice bodu D. Ty

získám tím, že spojím

body AB a BC v nárýsně a

přes rovnoběžky

zbývajícím třetím bodem

ze zadání. V průsečíku

rovnoběžek získám bod D

v nárýsně. V půdorysně

tento postup opakuji.

Ověření správnosti

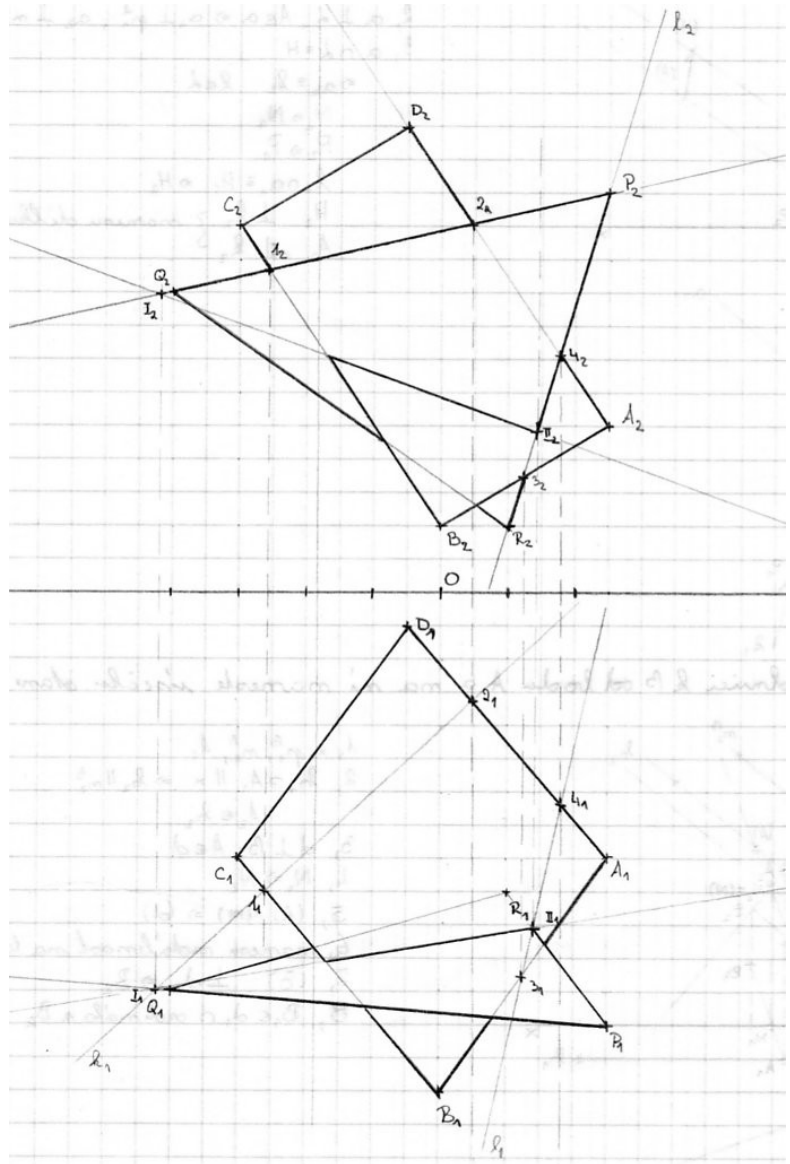
rýsování mi musí tento

bod v nárýsně i půdorysně

ležet na kolmici

k základnici, tedy

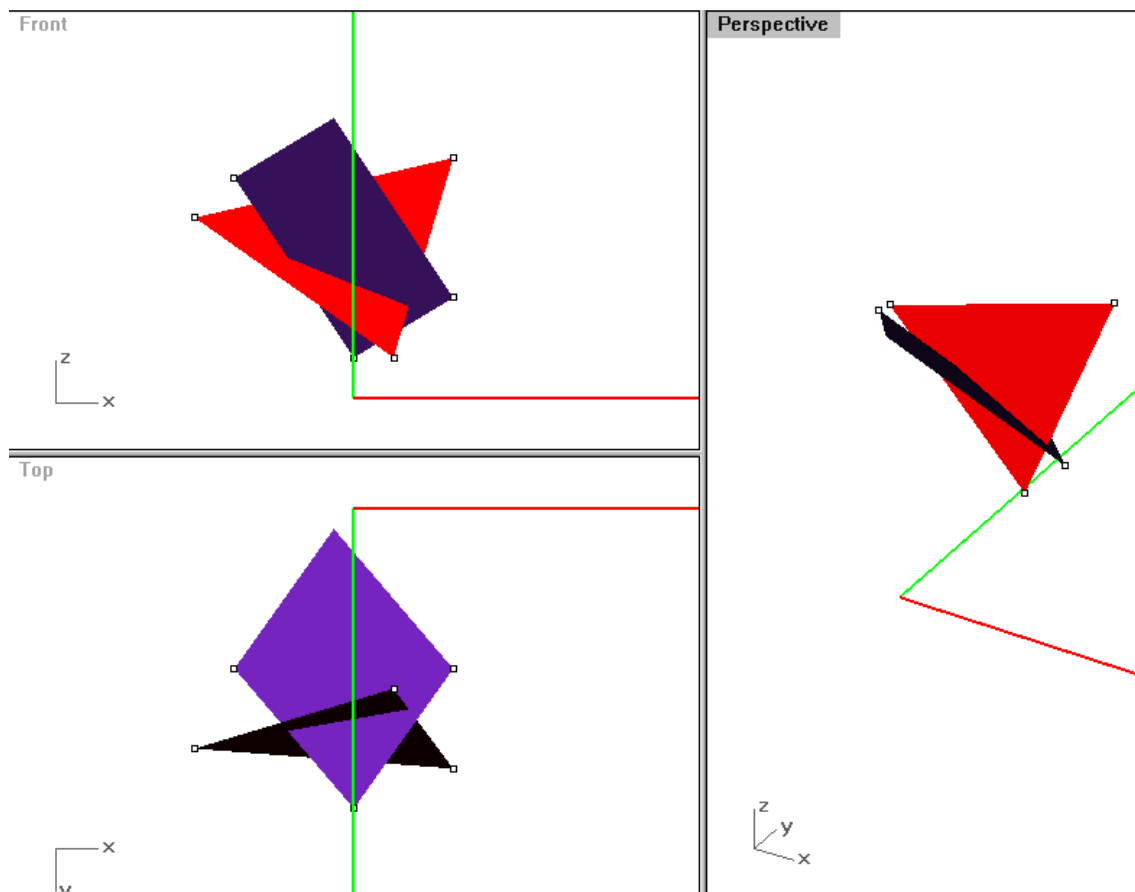
na ordinále. Poté si naleznou v nárýsně i půdorysně dané průsečíky a ty převedu přes ordinálu na opačnou průmětnu. Nyní vytáhnu viditelnost obou obrazců.



Obr. 60: Klasické rýsování průniku

V programu Rhinoceros postupuji opět přes příkaz Point, kde si zobrazím všechny body ze zadání. Příkazem Surface zobrazím trojúhelník PQR. Rovnoběžník vytvořím

pomocí příkazu Rectangle, u kterého mi postačí znát krajní body na jedné straně a jeden bod na protější straně. Toto jsem rýsovala pouze v perspektivním pohledu a do půdorysné a nárysné průmětny se daný problém zobrazoval automaticky. Průsečík a viditelnost se opět zobrazí po vystínování problému.

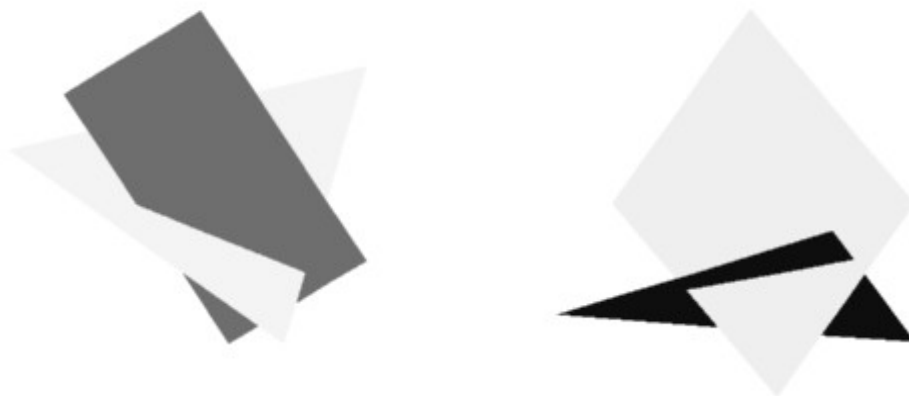


Obr. 61: Celkové vystínování všech pohledů

Kladem programu Rhinoceros je opět jednoduché rýsování v jednom pohledu a následné převedení do průměten. Tímto se nemusí pracně rýsovat zadání jako v klasickém rýsování, ale provádí se zde pomocí jednoduchého příkazu.

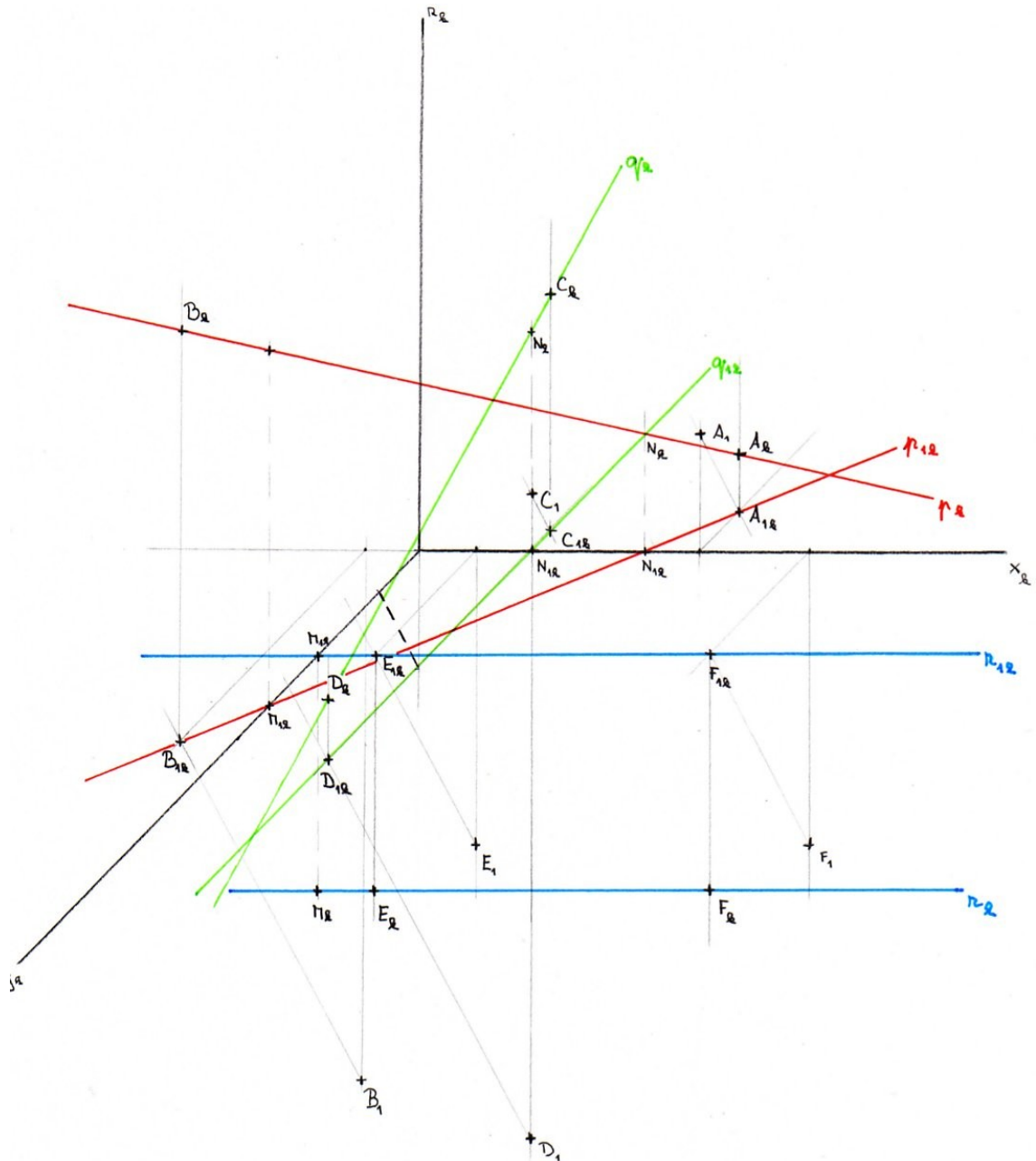
Záporem je opět nepřesné zobrazení daného průseku trojúhelníka a rovnoběžníka. Naproti tomu však pro názornou ukázkou ve výuce je plně dostačující. Jakož je i postačující ba naopak velikou výhodou vystínování a tím názorné předvedení viditelnosti obou obrazců.





*Obr. 62: Viditelnost*

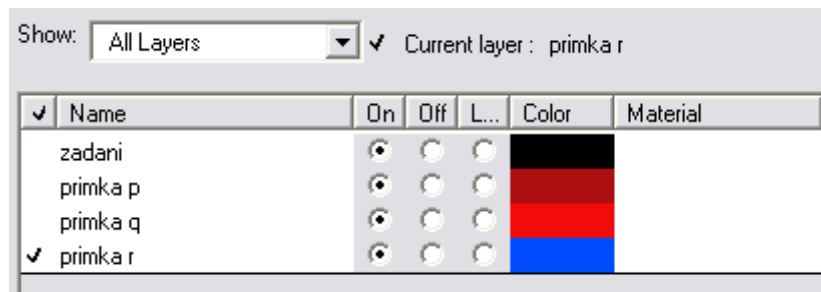
Příklad 4: Sestrojte průměty přímek  $p = AB$ ,  $q = CD$  a  $r = EF$ , určete jejich polohu vzhledem k průmětnám a jejich stopníky.  $A = [5;-2;1]$ ,  $B = [-1;9;7]$ ,  $C = [2;-1;4]$ ,  $D = [2;10;1]$ ,  $E = [1;5;-4]$  a  $F = [7;5;-4]$ ,  $\omega = 135^\circ$ ,  $q = \frac{1}{2}$ .



Obr. 63: Poloha přímek vzhledem k průmětnám

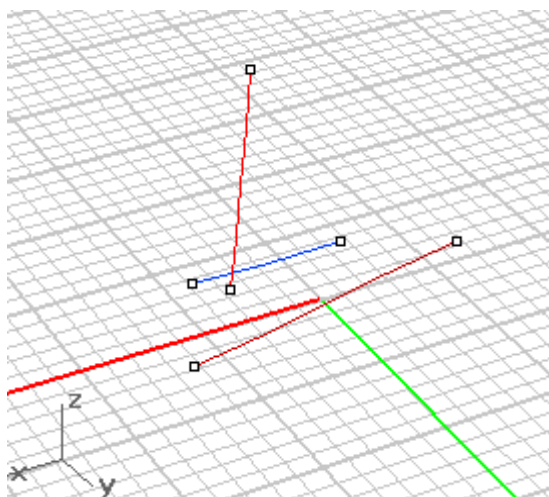
V klasickém rýsování jsem si nejprve určila kosouhlé promítání pomocí kříže a poměru zkrácení. Poté jsem vynesla všechny zadané body do průměten. Po vynesení bodů již stačilo spojit body ze zadaných přímek, tedy body A a B pro přímku  $p$ , body C a D pro určení přímky  $q$  a nakonec body E a F pro přímku  $r$ . Nakonec se určili stopníky

u každé přímky. Poté již stačí rozhodnout jejich polohy vzhledem k průmětnám. Přímka p je v obecné poloze, tudíž nelze zjistit vzájemnou polohu k průmětnám. Přímka q je rovnoběžná s bokorysnou a nakonec přímka r je taky rovnoběžná ale s nárysnou.

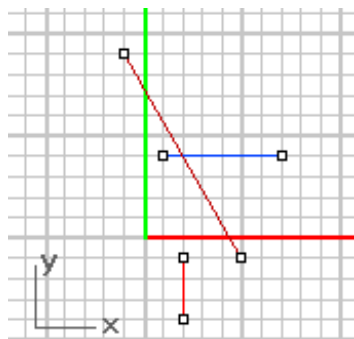


Obr. 64: Vrstvy v příkladu

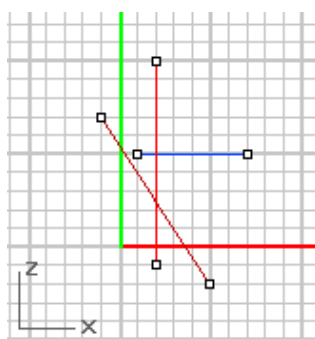
V programu Rhinoceros nastal první problém s nastavením os. Ty jsou pevně dané a nelze je měnit, tudíž lze použít pouze promítání s kolmými průmětny. Přesto jsem se pokusila problém zde vyřešit, i když výsledný efekt nebude stejný jako v klasickém rýsování. Nejprve jsem zanesla všechny body do soustavy a poté je pospojovala podle zadání do přímek. V tomto případě již práce v programu skončila a musím pouze posuzovat podle vlastního mínění, jak jsou dané přímky rozmístěny.



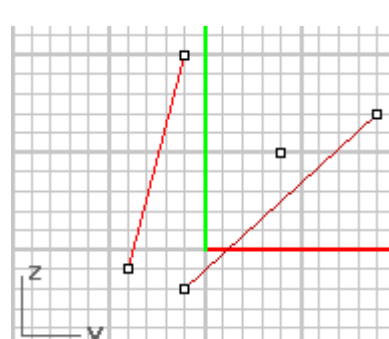
Obr. 65: Perspektivní pohled na přímky



Obr. 66: Rovina (xy)



Obr. 67: Rovina (xz)



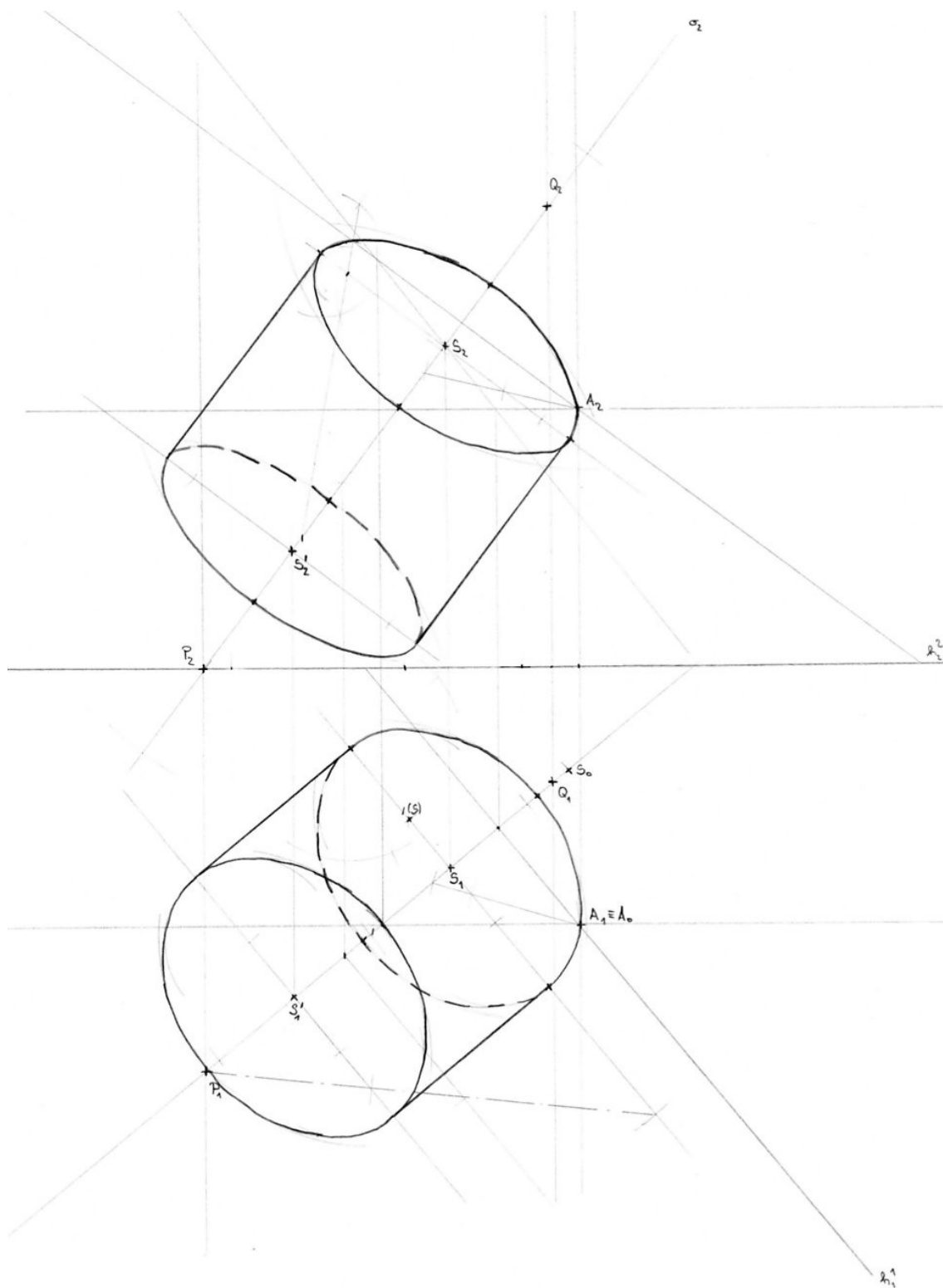
Obr. 68: Rovina (yz)

V tomto příkladu nevidím moc kladů pro program Rhinoceros. Již první problém nastal při určení os a tím je zkeslen i výsledek. Jinak zde bylo opět rychlé narýsování bodů a poté propojení těchto bodů v přímky.

## 5.2 Trojrozměrné objekty

Příklad 1: Zobrazte rotační válec výšky  $v = 5 \text{ cm}$ , je-li dána jeho osa  $o$  zadaná body  $P = [-3,5;7;0]$   $Q = [2,5;2;8]$  a bod  $A = [3;4,5;4,5]$  jeho podstavné hrany.

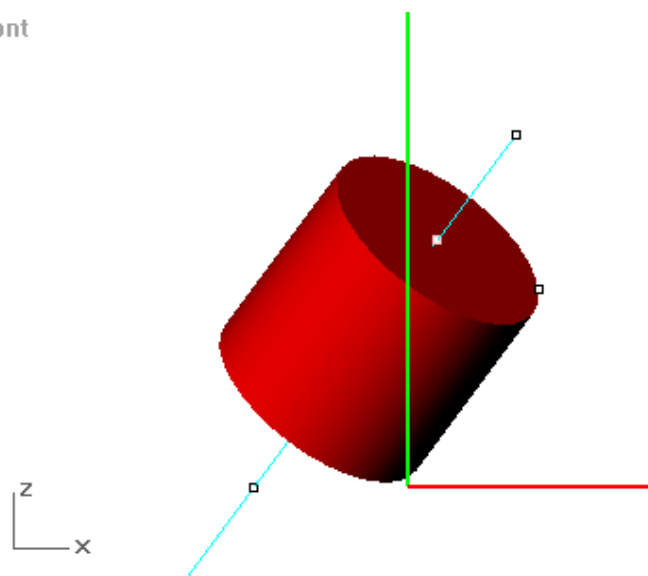
V klasickém rýsování na papír si nejdříve musíme představit, jak má výsledný objekt, v tomto případě válec, vypadat. Po narýsování zadání v půdorysně a nárysně bodu A a osy válce zadané body PQ, musíme dorýsovat hlavní rovinu, která je kolmá k ose válce a bod A náleží této rovině. Dále si v půdorysně zvolíme body K a L, které leží v průsečíku osy válce a na přímkách hlavní roviny. Tyto dva body následně převedeme pomocí ordinály do nárysné průmětny. Po spojení bodů KL v nárysně nám vznikne průsečík této přímky s osou válce. Tento průsečík nám označuje střed podstavy válce v nárysně. Opět přeneseme střed z nárysně do půdorysně pomocí ordinály. Střed se nám zobrazí opět na ose válce. Nyní potřebujeme zjistit hlavní osy elipsy, protože kružnice se nám zde zobrazuje právě do elipsy. To zajistíme sklopením středu podstavy do hlavní roviny. Do hlavní roviny též přeneseme i zadaný bod A. Tento bod je totožný jak ve skutečnosti tak i zobrazený v průmětnách. Díky tomu jsme zjistili skutečný poloměr podstavné kružnice. Nyní vedeme středem rovnoběžnou přímkou s hlavní přímkou v půdorysně a na ní nanášíme ze středu zjištěný poloměr. Tím jsme získali body M a N, které náleží kružnici. Tyto body přeneseme pomocí ordinály do nárysně. Tento krok s body M a N opakujeme i v nárysně, ovšem zde místo bodů MN, jsem zvolila body UV. Body M, N, U a V nám tvoří sdružené průměty kružnice. Proto abychom mohli narýsovat elipsu (ve skutečnosti kružnici), použijeme proužkovou konstrukci. Pro zjištění výšky válce si musíme nejdříve daný střed podstavy a pomocný bod ležící na ose otočit. Poté stačí od otočeného bodu S nanést zadanou vzdálenost výšky a přenést zpět do průmětu. Poté již stačí daný válec vykreslit a určit jeho polohu pro viditelnost.



Obr. 69: Klasické rýsování příkladu válce

V programu Rhinoceros bylo rýsování válce velice rychlé. Po původním zavedení bodů pomocí příkazu Point, jsme si zobrazili osu válce příkazem Line. Poté jsme si vytvořili kolmici z bodu A na osu válce, zadané body PQ. Průsečík osy válce a nově zobrazené kolmice je již potřebný střed podstavy válce. Pak již stačilo pomocí příkazu

Front

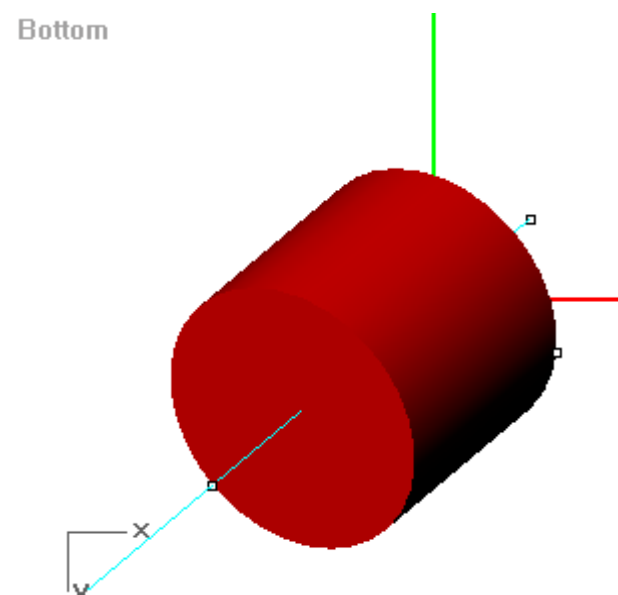


Obr. 70: Nárýs válce

Cylinder vykreslit válec, kde již známe střed podstavy, bod podstavy a výšku válce. Stačí jen určit směr polohy válce. Po vystínování pohledů zjistíme i viditelnost válce v průmětnách. Vše se opět rýsovalo v perspektivním pohledu.

Kladem zmiňovaného programu Rhinoceros je velice snadné a rychlé narýsování zadaného válce. Zároveň je zde i velice dobře patrná viditelnost po vystínování pohledu. Záporem výše uvedeného programu je problém nepochopení základních principů

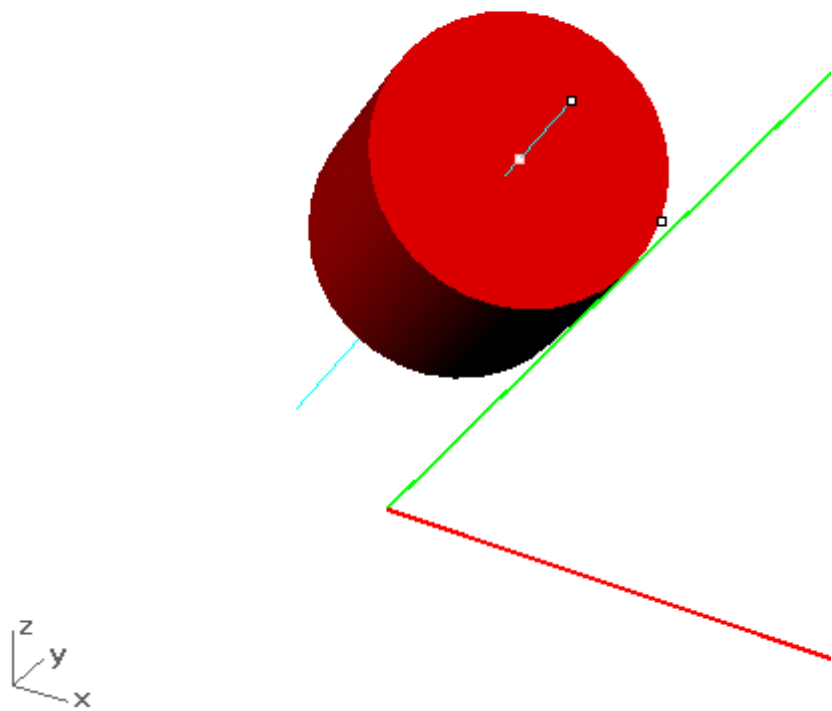
Bottom



Obr. 71: Půdorys válce

rýsování v deskriptivní geometrii, které zde není potřeba využít.

## Perspective

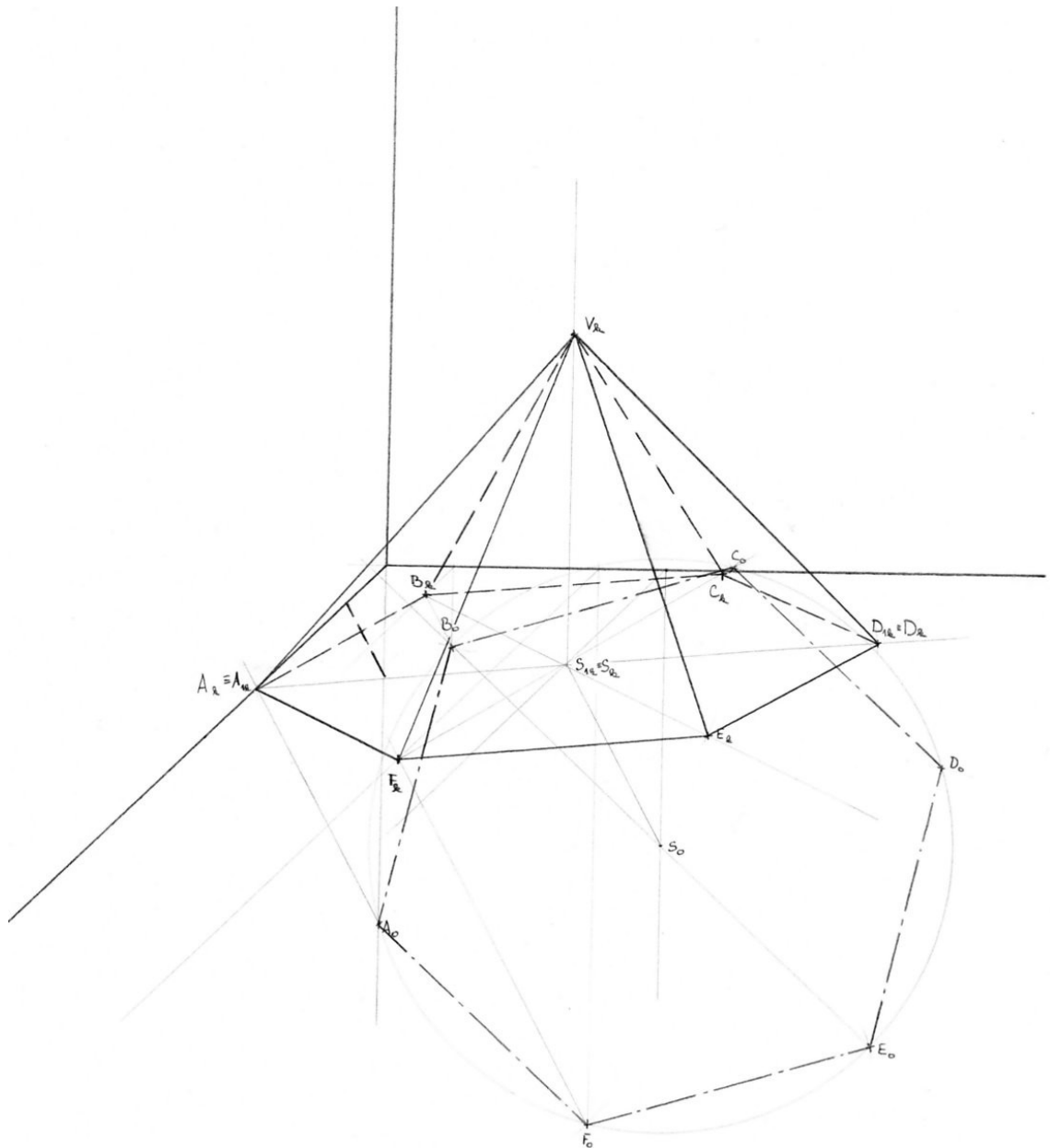


Obr. 72: Příklad z perspektivního pohledu



Příklad 2: V kosoúhlém promítání  $\omega = 135^\circ$ ,  $q = \frac{1}{2}$  zobrazte pravidelný šestiboký jehlan s podstavou v rovině (xy) o středu  $S=[5;5;0]$  a vrcholu  $A=[0;6;5]$ . Výška jehlanu je 6 cm.

V klasickém rýsování si nejdříve zobrazíme osy v daném úhlu a určíme si směr pro zkrácení bodů podle koeficientu. Poté si zobrazíme zadání v daném promítání. První, co musíme provést, je převedení podstavného šestiúhelníku do skutečné velikosti.

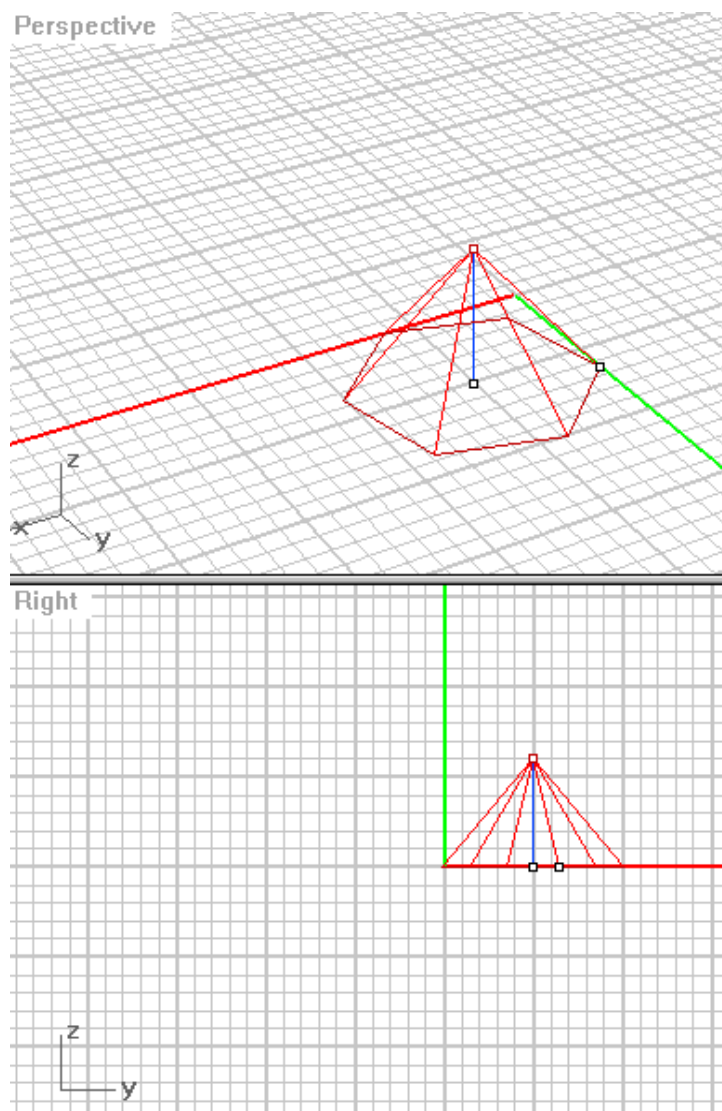


Obr. 73: Kosoúhlé promítání - jehlan v rovině (xy)

Poté již stačí každý bod podstavné hrany převést zpět do kosoúhlého průmětu.

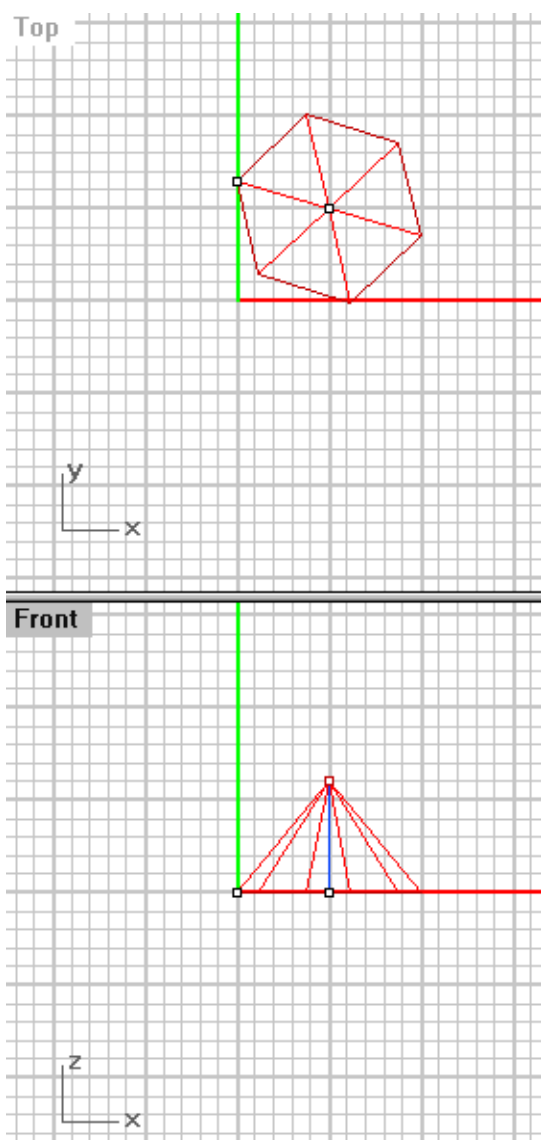
Po převedení bodů nám stačí pouze vynést výšku ze středu podstavy, která svoji délku nemění. Každý bod spojíme s vrcholem a vytáhneme viditelnost daného obrazce.

Ihned po spuštění programu Rhinoceros nastává problém, protože neumožňuje jiné nastavení os než k sobě kolmé. I přesto jsem se pokusila daný jehlan narýsovat s poznámkou, že výsledný obrázek nebude přesný jako při rýsování na papír. Zadala jsem si tedy středový bod a bod vrcholu podstavu v perspektivním okně. Dále jsem narazila na další problém, protože program neumí vykreslovat složitější tělása, právě jako je šestiboký jehlan. Proto jsem nejdříve pomocí příkazu Polygon vykreslila podstavu, v tomto případě pravidelný šestiúhelník za pomoci středu



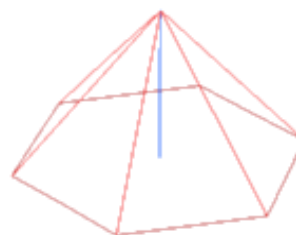
Obr. 74: Perspektivní pohled a pohled na rovinu (yz)

a vrcholu. Poté jsem potřebovala si znázornit výšku. Tu jsem vytvořila za pomoci příkazu Line v pohledu Right, který je kolmý k podstavné rovině. Pak už stačilo spojit každý vrchol s danou výškou. Jelikož je ale vše tvořeno pouze v dvourozměrném prostoru, není možné určit přesně viditelnost ani po vystínování pohledu.



Obr. 75: Pohledy na roviny  $(xy)$  a  $(xz)$

Kladem v tomto příkladu pro program Rhinoceros je pouze rychlé vykreslení šestiúhelníku již ve správné velikosti. Vše ostatní, jak již bylo zmiňováno výše bych připisovala k záporům. Tento příklad je tedy vhodné řešit přímo klasickou metodou rýsování na papír a program Rhinoceros pro jistotu nespouštět.



Obr. 76: Vystínovaný model v perspektivním pohledu

## 6. Závěr

Závěrem bych chtěla říci, že práce s tímto programem, díky perspektivnímu pohledu a rýsování v něm, je velice dobrá. Kladně hodnotím i automatický převod na nárysny a půdorysný pohled. Uživatelům (žákům) může posloužit hlavně pro představivost, která je dána stínováním, tudíž je zde velice dobrá názornost.

Jako negativní stránku programu hodnotím to, že celý program je pouze v anglickém jazyce, což by ale pro dnešní mládež neměl být velký problém. Dalším nedostatkem programu Rhinoceros je to, že si nemůžeme nastavit vlastní rozvržení os. Tudíž pro deskriptivní geometrii lze využít pouze Mongeovu projekci, která je zadána dvěma k sobě kolmými průmětnami.

Dále si myslím, že program je vhodný jak pro názornou ukázkou, tak i pro vypracování určitých problémů v Mongeově promítání. Ale přesto by se nemělo zapomínat na klasické rýsování na papír, kde je potřeba znát hlavní úkony pro vypracování rysů ve všech promítání deskriptivní geometrie.

Program Rhinoceros je velice vhodný do výuky kvůli již zmíněné názorné představivosti a snadnému ovládní. Školy jsou navíc v současné době již dobře technicky vybavené počítači, takže práce s programem nenaruší časové rozvržení vyučovací hodiny.

Myslím si, že moje diplomová práce může zároveň sloužit jako návod pro učitele i žáky při výkladu, procvičování i opakování úloh z deskriptivní geometrie.

## **Literatura:**

<http://www.rhino3d.cz/>

<http://www.cz.rhino3d.com/>

<http://web.quick.cz/Rhinoceros/>

Leinveber, J., Švercl, J. a kol. Technické kreslení a základy deskriptivní geometrie.

Praha: Scientia, spol. s r. o. pedagogické nakladatelství, 1998.

Urban, A. Deskriptivní geometrie I, II. Praha: SNTL, 1977.

Drs, L. Deskriptivní geometrie pro střední školy II. Praha: Prometheus, s. r. o., 1996.