

Projekt Honza

Autor: Jan Kolář, DiS

Verze programu: 3

Obsah

0	Revize dokumentu	1
1	Obecné pojednání	2
2	Schémata uspořádání	2
2.1	Datová struktura	2
2.2	Primární datové soubory	3
2.2.1	Obecný základní	3
2.2.2	Slovosledný	4
2.2.3	Pohybový	5
2.3	Sekundární datové soubory	6
2.3.1	Lidé	6
2.3.2	Obrázky	6
2.3.3	Osobnost	6
2.3.4	Pohyby	8
2.3.5	Slovník	8
2.3.6	Struktury	9
2.3.7	Znalosti	9
2.3.8	Zvuky	9
2.4	Oddílové soubory	10
3	Programovací funkce	11
3.1	Stručný seznam dostupných funkcí	11
3.2	Práce s pamětí	13
3.3	Pohyby	16
3.4	Práce s historií	17
3.5	Speciální	18
3.6	Vyhledávací	20
3.7	Práce s obrázky	23
5	Podrobný popis jednotlivých funkcí	24
5.1	Práce s historií a její využívání	24
5.2	Práce s datem a časem + jeho využívání	25
5.3	Funkce vyhledávání	26
5.3.1	Vyhledávání znalostí s kombinací funkcí \$V a \$k	27
5.4	Sestavení příběhu	28
5.4.1	Potřebné soubory:	28
5.4.2	Princip fungování	29

0 Revize dokumentu

Od verze 3.6 vytvořen oddíl revize této dokumentace.

16. 2. 2014: Pro aktuální verzi programu jsou zde uvedeny datové soubory obsahující slovní spojení pomocí znaku „+“. V aktuální verzi programu je použit znak „&“.

14. 2. 2014: Upraven modul práce s obrázky (Vizualizace pomocí obrázků). Kapitola 2.3.2 - Obrázky

1 Obecné pojednání

Program slouží ke vzájemné konverzaci člověka s počítačem, přesněji řečeno – s počítačovým programem, tedy softwarem. Cílem projektu je vytvoření programu, který bude člověka simulovat do takové míry, že nebude možno rozpoznat, píše-li si uživatel s programem nebo se skutečným člověkem.

2 Schémata uspořádání

2.1 Datová struktura

Veškeré vědomostní prvky jsou uloženy ve složce „data“. Strom uspořádání souborů je na následujícím diagramu:

Primární	Sekundární
Název: 0 – nekonečno.txt (0.txt, 1.txt, 2.txt,...) Popis: Základní datové soubory obsahující vyhodnocení podnětu	Název oddílu: lide Popis: Obsahuje informace o uživateli a lidech, se kterými hovořil.
Název: N.txt Popis: Nulový datový soubor – odpověď je „ “	Název oddílu: obrázky Popis: Obsahuje data a obrázky pro obrázkovou vizualizaci.
Název: s.txt (<i>slovosled</i>) Popis: Určuje pořadí vypisování reakcí základních datových souborů v případě většího množství.	Název oddílu: osobnost Popis: Obsahuje informace o vlastní osobě – vlastní já. (jméno, místo narození, nálada, atd...)
Název: telo.ini (<i>vizualizace těla</i>) Popis: Obsahuje pozice jednotlivých částí těla.	Název oddílu: pohyby Popis: Obsahuje pohybové sekvence.
Název: x.txt (<i>oddílový</i>) Popis: Obecný oddílově strukturovaný soubor.	Název oddílu: slovník Popis: Obsahuje skloňování podstatných jmen
	Název oddílu: struktury Popis: Obsahuje strukturované znalosti, které mohou mít nějakou posloupnost (měsíce, dny v týdnu, pojmenování nálady,...), nebo synonymní význam (Honza, Jan, Honzík,...).
	Název oddílu: znalosti Popis: Obsahuje ostatní znalostní a vědomostní soubory (vtipy, jazykolamy, aktuální den, měsíc, rok, křestní jména, svátky, asociační znalosti (pes = štěká,...))
	Název oddílu: zvuky Popis: Obsahuje zvukové soubory, které využívá pro zvukovou produkci reakce.

Primární

Jsou základní datové soubory, které program primárně prochází při zadání podnětu a které většinou odkazují pomocí proměnných na soubory sekundární.

Sekundární

Jsou doplňkové datové soubory, které jsou rozděleny ve složkách (oddílech) a na které odkazují primární datové soubory.

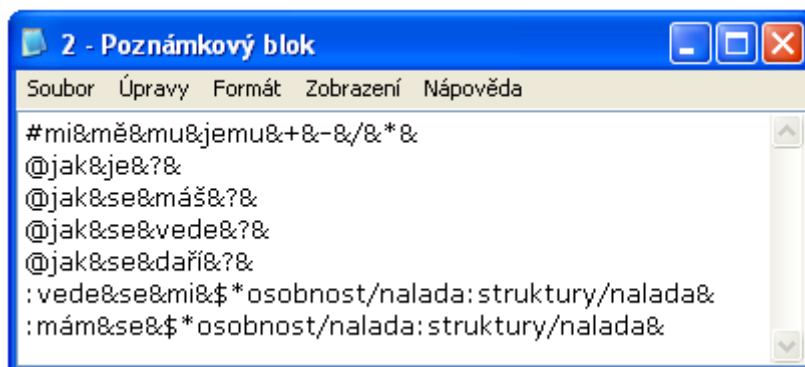
2.2 Primární datové soubory

Základní znalostní soubory jsou textové soubory s příponou **.txt**, uložené ve složce „**data**“ a očíslovány celým číslem.

Zde je několik typů souborů, které jsou popisovány v následujících kapitolách:

- a) **Obecný základní** (0,1,2,3,4.txt,...)
- b) **Slovosledový** (s.txt)
- c) **Obecný oddílový** (0,1,2,3,4.txt,...)
- d) **Pohybový** (telo.ini)

2.2.1 Obecný základní



Příklad základního znalostního souboru

2.2.1.1 Význam proměnných

= slova, která podnět nesmí obsahovat

@ = slova, která podnět musí obsahovat

: = možné odpovědi

& = spojení jednotlivých prvků do řetězce

\$ = variabilní hodnota

2.2.1.2 Proměnná „#“ (křížek)

Slova, která podnět nesmí obsahovat. Jsou uložena vždy na prvním řádku, který začíná znakem „#“. Pokud podnět bude obsahovat slova, která se nachází v této části, dále neprohledává zbytek souboru a pokračuje k dalšímu. V případě funkce \$CHtext použije text z historie.

Hodnoty v této proměnné jsou neměnné.

2.2.1.3 Proměnná „@“ (zavináč)

Slova, která podnět musí obsahovat. Čím více se slova v podnětu shodují s těmito slovy v jednotlivých řádcích, tím vyšší je pravděpodobnosti odpovědi.

Program vyhodnocuje pravděpodobnost, tedy % porovnává řetězec podnětu a řetězce v tomto souboru. Pokud % hodnota přesáhne nastavené hranice, je povolena příslušná reakce.

Hodnoty v této proměnné mohou být variabilní.

Variabilní hodnoty

- „CH“ znamená Čtení z Historie. Souvisí s funkcí „\$ZH“ u proměnné dvojtečka. Program přečte potřebné slovo z historie. Stačí, aby bylo zadáno u jednoho z jakýchkoliv slov.

Nesmí být samostatně! - (slovo1+\$CH+slovo2+).

Příklad použití: \$CHnahoru (čte poslední slovo z historie) a slovo „nahoru“ nechá ve slovech.

2.2.1.4 Proměnná „:“ (dvojtečka)

Pokud existuje více možností – tedy je více řádků, které začínají dvojtečkou, vybírá náhodně jednu z nich.

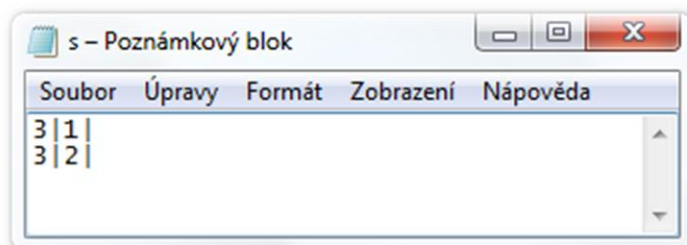
Konstantní hodnoty jsou pouze slova, která se vypíší jako text odpovědi.

Hodnoty v této proměnné mohou být variabilní.

Variabilní hodnoty začínají znakem “\$”

2.2.2 Slovosledný

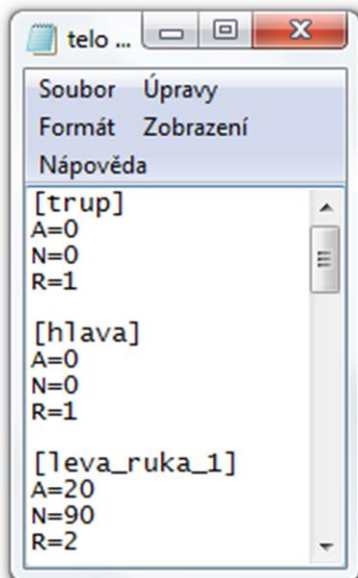
Soubor označen jako „s.txt“. Obsahuje slovosled, který je potřeba dodržet.



Pokud k odpovědi využije soubor 3.txt a 1.txt, vypíše vždy jako první soubor 3.txt a až poté soubor 1.txt.

2.2.3 Pohybový

Označen jako „**telo.ini**“. Obsahuje aktuální a nové pozice jednotlivých částí těla, společně s rychlostí, se kterou se uveče na případnou pozici novou.



Jednotlivě odděleny jsou následující prvky:

Trup, hlava, Levá ruka 1 a 2, Pravá ruka 1 a 2, Levá noha 1 a 2, Pravá noha 1 a 2, Levé a pravé chodidlo.

A = Aktuální hodnota, kterou čte vizualizační program a podle ní zobrazuje stav.

N = Nová hodnota, na kterou se musí dostat hodnota **A**.

R = Rychlost, se kterou se musí **A** dostat na hodnotu **N**. Nebo-li po kolika se bude hodnota **A** přičítat/odečítat.

2.3 Sekundární datové soubory

Tyto datové soubory rozšiřují variabilitu programu a umožňují širší škálu vědomostí pro zlepšení, a nové prvky. Jsou rozděleny do jednotlivých složek – oddílů.

Veškerá slova by měla být v základním tvaru. Jejich skloňování a gramatická úprava probíhá přesfunkce v primárních datových souborech.

Seznam sekundárních oddílů:

- | | |
|-------------|--------------|
| a) Lidé | e) Slovník |
| b) Obrázky | f) Struktury |
| c) Osobnost | g) Znalosti |
| d) Pohyby | h) Zvuky |

2.3.1 Lidé

Soubory obsahující informace o lidech, se kterými již program komunikoval. Informace o aktuálním uživateli jsou uloženy v souboru **x.txt**. Tedy pokud se uživatel zeptá na své jméno, odpoví mu vypsaním prvních dvou řádků.

2.3.2 Obrázky

Oddíl, který je používán pro vizualizační modul pomocí obrázků.

Tento modul se zapíná pomocí programu „obrazkove_zobrazeni.exe“.

V nastavení programu je možnost deaktivovat tento modul.

Všechny obrázky musí mít pouze následující formáty: jpg, png, bmp, gif (i animovaný).

Obrázky používané offline jsou uloženy ve složce „offline“.

Aktuální zobrazovaný obrázek má název **00.přípona**.

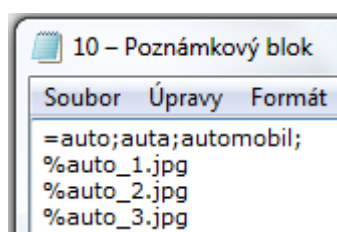
Soubor **01.txt** je pomocný soubor pro rozeznání správné přípony obrázku.

Soubory s odkazy na obrázky jsou číslovány od 10 výš.

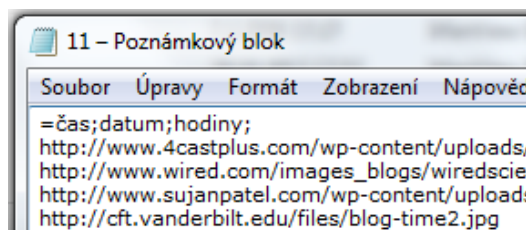
Pro offline obrázky musí být počáteční znak „%“.

Program automaticky obrázky vyhledává na základě slov v konverzaci.

V případě, že obrázek nebude stažen, vypíše do odpovědi text s hláškou.



Zobrazení offline obrázku



Zobrazení online obrázku

2.3.3 Osobnost

Uloženy hodnoty o aktuálním stavu vlastní osobnosti, jako například aktuální nálada (tu využívá například variabilní odkaz pro výběr slova pro odpověď).

Oddílový soubor x.txt obsahuje všechny ostatní znalosti o vlastní osobě.

Hodnota	Rozmezí	Popis
nálada	1 - 5	1 = nejlepší 5 = špatná
x.txt	---	Oddílový soubor

2.3.4 Pohyby

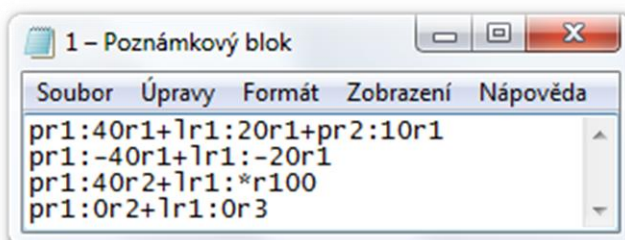
Umístění pohybových souborů: data/pohyby/

Typy souborů: .txt

Rychlost pohybů je od 1 do x, kde jednotkou je jeden cyklus časovače.

Příklad:

Fáze	S čím pohybovat	Na hodnotu	Rychlost
1	Pravá ruka 1	40	1
	Levá ruka 1	20	1
	Pravá ruka 2	10	1
2	Pravá ruka 1	-40	1
	Levá ruka 1	-20	1
3	Pravá ruka 1	40	2
	Levá ruka 1	---	100
4	Pravá ruka 1	0	2
	Levá ruka 1	0	3



Hodnota ***r100** znamená zůstat na stejné pozici po dobu 100 cyklů.

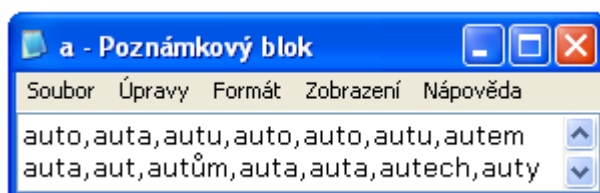
Časová osa:

Část těla	Časové jednotky
Pravá ruka 1	
Levá ruka 1	
Pravá ruka 2	

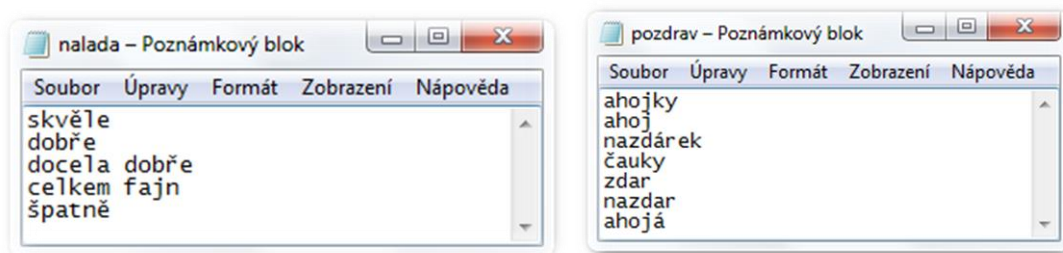
2.3.5 Slovník

Oddíl obsahuje seznam všech českých slov a jejich slovních tvarů podle následující struktury. Pro urychlení vyhledávání jsou slova rozdělena do jednotlivých souborů abecedně, podle počátečního písmene.

a) **Podstatná jména** – pádové skloňování – 1. až 7. pád.



2.3.6 Struktury



Soubory ve složce „struktury“ představují soubory s určitými řetězci, které jsou vybírány podle předem definovaného klíče (variabilní hodnoty).

Například řetězec ze souboru „nalada“ je vybírán podle variabilního klíče (\$*), získaného ze souboru, kde je uložena hodnota aktuální nálady. Hodnota nálady = 1, potom „skvěle“. Hodnota nálady = 5, potom „špatně“. Co řádek, to hodnota.

Více informací v kapitole „**Chyba! Nenalezen zdroj odkazů.**“

2.3.7 Znalosti

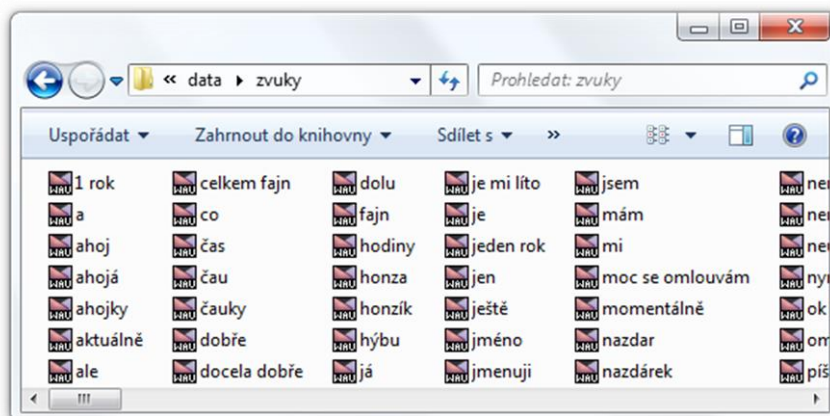
Složka obsahuje znalostní soubory různě strukturovaných souborů. Rozděleny jsou na obecné a konkrétní.

- a) Obecné (určeny číslem)
- b) Konkrétní (určeny názvem)

2.3.8 Zvuky

Složka obsahuje zvukové nahrávky, ve formátu **.wav**, jednotlivých slov a slovních spojení, které mohou být obsaženy v textu odpovědi.

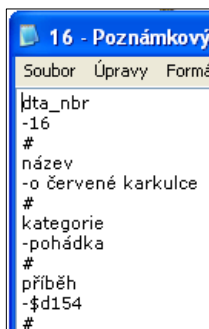
Pokud v odpovědi bude text, který neobsahuje zvukový soubor, zvuk nepřehraje.



2.4 Oddílové soubory

Jedná se o speciální typ souborů, které mohou být uloženy v různých složkách. Používají se většinou jako znalostní soubory, ve kterých jsou data uspořádána v jednotlivých oddílech.

Tyto oddílové soubory rovněž umožňují používání programových funkcí (příkazů). Podmínkou je, že v každém pod-oddílovém řádku může být pouze jedna funkce a jen samostatně, bez jakéhokoliv dalšího textu.



Například v tomto oddílovém souboru číslo 16 je v oddílu „příběh“ použita funkce \$d odkazující na primární datový soubor s číslem 154. V tomto primárním souboru je uložen daný příběh.

	<p>Každý oddíl je ukončen znakem #.</p> <p>A = systémový oddíl – musí zůstat stejný pro všechny soubory. Obsahuje jedinou hodnotu, která je rovná názvu souboru, ve kterém je uložen.</p> <p>B a C jsou další pod-oddíly.</p> <p>B udává název oddílu.</p> <p>C udává jednotlivé hodnoty daného oddílu.</p> <p>Jednotlivé hodnoty (řetězce) začínají vždy jedním specifickým znakem - parametrem. Ten může být jakýkoliv (číslo, znak, písmeno). Parametry mohou být v jednom oddíle různé.</p> <p>Pozor! S číselnými parametry pracují některé specifické funkce.</p>
--	---

3 Programovací funkce

3.1 Stručný seznam dostupných funkcí

Práce s pamětí

(:) **\$*zdroj:cíl**

Vypíše text z cíle z čísla řádku, který je ve zdroji.

(:) **\$x:soubor**

Vypíše náhodný řádek ze souboru.

(:) **\$vznak:zdroj:cíl#d?**

Vypíše **všechny** řádky v cíli začínající znakem, které se nacházejí pod řetězcem, který je uložen ve zdroji.

(:) **\$vznak!zdroj:cíl**

Vypíše **náhodný** řádek v cíli začínající znakem, který se nachází pod řetězcem, který je uložen ve zdroji.

(:) **\$vznak;parametr:zdroj**

(:) **\$vznak*parametr:zdroj**

(:) **\$vcislo;parametr:zdroj**

(:) **\$cčíslo:zdroj**

(:) **\$tčíslo:zdroj**

(:) **\$u:cíl**

Práce s pohyby

(:) **\$P?**

(:) **\$Px**

Práce s historií

(:) **\$ZHslovo**

(@ #) **\$CHslovo**

(:) **\$SH**

Speciální funkce

(:) **\$d?**

(:) **\$p**

(@) **\$p**

(@) **\$p!**

(:) **\$0**

(:) **\$W:soubor**

(:) **\$C:složka:oddíl1+oddíl2=d?:d?**

Vyhledávací funkce

(:) **\$q:soubor1:soubor2-?+d?:d?**

(:) **\$q!zdroj:řetězec:znak**

(:) **\$k:zdroj:znak:řetězec+d?:d?**

(:) **\$s:soubor:řádek:pád**

(:) \$s!slovo:pád
(:) \$s;pád
(:) \$V:soubor+d?:d?
(:) \$S:soubor

Práce s obrázky:

(:) \$od:zdroj:cislo1:cislo2
(:) \$od!zdroj:cislo1:cislo2

3.2 Práce s pamětí

(:) \$***zdroj:cíl**

Funkce

- V souboru „**zdroj**“ přečte číslo na prvním řádku.
- V souboru „**cíl**“ vybere text na příslušném čísle řádku.
- Vybraný text vypíše.

Příklad použití: \$*osobnost/nalada:strukтуры/nalada

- V souboru „osobnost/nalada.txt“ je uložené číslo 3.
- V souboru „strukтуры/nalada.txt“ vybere 3. řádek a text vypíše.

(:) \$**x:soubor**

Funkce

- Ze souboru vybere náhodný řádek a ten vypíše.

Příklad použití: \$x:strukтуры/поздрав

(:) \$vznak:zdroj:cíl#d? (v 3.2)

Funkce

- Více v kapitole 5.3 Funkce vyhledávání.
- V souboru „cíl“ vyhledá řádek, na kterém se nachází řetězec ze **zdroje**.
- Pod tímto řádkem hledá řádky (dokud nenarazí na znak „#“), které začínají zadaným **znakem**.
- Tyto řádky **všechny** vypíše do odpovědi.
- Znak může být jakýkoliv. Číslo, písmeno, speciální znak,...
- V případě nenalezení daného znaku použije zadaný datový soubor

Příklad použití: \$v-:znalosti/aktualni_datum_bez_roku:znalosti/svatky#d130

- Podle data vypíše, kdo má svátek. Řetězec se jménem musí začínat znakem „-“.
- Pokud řádek se znakem „-“ nebude existovat, vypíše datový soubor s číslem 130.

(:) \$vznak!zdroj:cíl

Funkce

- Viz předchozí funkce s obměnou náhodného výběru.
- V souboru „cíl“ vyhledá náhodné číslo, které zjistil ze **zdroje**.
- Pod tímto řádkem hledá řádky (dokud nenarazí na znak „#“), které začínají zadaným **znakem**.
- Tyto řádky vypíše do odpovědi.
- Ve zdroji musí být vždy celé číslo. Například pro číslo 5 vybere náhodné od 0 do 5.

(:) \$vznak;oddíl:zdroj

Funkce

- V souboru „zdroj“ vyhledá **oddíl**. (znak, řetězec, číslo)
- Pod tímto řádkem hledá řádky (dokud nenarazí na znak „#“), které začínají zadaným **znakem**.
- Tyto řádky **všechny** vypíše do odpovědi.
- Znak může být jakýkoliv, ale **ne číslo** - písmeno, speciální znak,...

Příklad použití: \$v-;názor:znalosti/filmy/x

- V souboru znalosti/filmy/x.txt vyhledá řetězec „názor“ a vypíše následující řádky začínající znakem „-“. (59)

(:) \$vznak*oddíl:zdroj#d?

Funkce

- V souboru „zdroj“ vyhledá **oddíl**. (znak, řetězec, číslo)
- V tomto oddílu hledá řádky (dokud nenarazí na znak „#“), které začínají zadaným **znakem**.
- Do odpovědi vypíše **náhodný řádek**.
- V případě nenalezení reaguje daným datovým souborem. (nemusí být definován)
- Znak může být jakýkoliv, ale **ne číslo** - písmeno, speciální znak,...

Příklad použití: \$v-*názor:znalosti/filmy/x#d12

- V souboru znalosti/filmy/x.txt vyhledá řetězec „názor“ a vypíše následující náhodný řádek začínající znakem „-“. Pokud nenalezne řádek, reaguje datovým souborem číslo 12.

(:) \$vcislo;oddíl:zdroj

Funkce

- V souboru „**zdroj**“ vyhledá **oddíl**. (znak, řetězec, číslo)
- Do odpovědi vypíše **řádek na pozici „cislo“**. Číslo musí být od 1 do 9.

Příklad použití: \$v1*názor:znalosti/filmy/x

- V souboru znalosti/filmy/x.txt vyhledá oddíl „názor“ a vypíše první řádek (bez počátečního znaku).

(:) \$cčíslo:zdroj

Funkce

- odkaz na datový soubor, ze kterého se vypíše číslo, které je na příslušném řádku

Příklad použití: \$c2:znalosti/vtipy-data (v odpovědi použije číslo z 2. řádku ze souboru)

Číslo za znakem „c“ nesmí být záporné, desetinné a nula a nesmí přesahovat počet řádků ve vyhledávaném souboru.

(:) \$tčíslo:zdroj

Funkce

- odkaz na datový soubor, ze kterého se vypíše text, který je na příslušném řádku

Příklad použití: \$t2:znalosti/vtipy-data (v odpovědi použije text z 2. řádku ze souboru)

Číslo za znakem „t“ nesmí být záporné, desetinné a nula a nesmí přesahovat počet řádků ve vyhledávaném souboru.

(:) \$u:cíl

Funkce

- uložení řetězce z podnětu do cílového souboru na konec.

Příklad použití: \$u:znalosti/prijmeni

- Do souboru „znalosti/prijmeni“ uloží řetězec podnětu.

3.3 Pohyby

(:) \$P?

Funkce

- Aktivuje pohybový cyklus, který je uložen pod číslem „?“.

Příklad použití: \$P3

- V pohybovém souboru č. 3 je uloženo třeba zamávání. Tímto se mávání aktivuje.

(:) \$Px

Funkce

- Reset všech pohybových cyklů. Všechny pohyby (otočení) nastaví na „0“ rychlostí 5.

Příklad použití: \$Px

- Uvedení do základní polohy – reset.

3.4 Práce s historií

(:) \$ZHslovo

Funkce

- Zapiše do historie zadané „slovo“. Může tím být i řetězec. „slovo“ se v odpovědi nezobrazí.

Příklad použití: \$ZHruce-nahoře (při pokynu „zvedni ruce“)

- Nyní ví, že má ruce nahoře - že je zvedl.

(@ #) \$CHslovo

Funkce

- Čte z historie „slovo“. Může tím být i řetězec. „slovo“ se v odpovědi nezobrazí.

Příklad použití: \$CHruce-nahoře (společně s textem „děkuji“)

- Může na podnět reagovat „Rád jsem zvedl ruce“

(:) \$SH

Funkce

- Smaže aktuální historii.

Příklad použití: netřeba...

3.5 Speciální

(:) \$d?

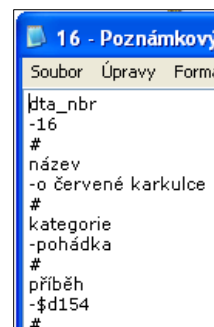
Funkce

- Odkaz na jiný datový soubor, který bude použit v odpovědi.

Příklad použití: \$d12&\$d13 (pro odpověď použije datový soubor číslo 12 a 13)

Rozsah použití:

- Základní datové soubory
- Znalostní soubory (pouze jednoduchý jednořádkový příkaz)



(:) \$p

Funkce

- Vypíše řetězec, který byl naposledy vyhledáván.

Příklad použití: \$p

(@) \$p

Funkce

- Detekuje řetězec, který vyhledá na základě toho, že slova, která jsou zadána, že podnět musí obsahovat, vynechá, a zbytek podnětu je ten vyhledávaný řetězec.

Příklad použití: \$p

Datový soubor	Podnět	Výstup
@jak&dělá&\$p&?&	jak dělá pes?	pes
	ahoj, jak dělá pes?	ahoj, pes
	jak dělá kočkovitá šelma?	kočkovitá šelma
	ahoj, jak dělá kočkovitá šelma?	ahoj, kočkovitá šelma

(@) \$p!

Funkce

- Detekuje řetězec, který vyhledá na základě toho, že z podnětu odstraní řetězce, které nemají být podle zadání a detekovaná slova jsou na přesně daném místě.

Příklad použití: \$p!

Datový soubor	Podnět	Výstup
@jak&dělá&\$p!&?&	jak dělá pes?	pes
	ahoj, jak dělá pes?	pes
	jak dělá kočkovitá šelma?	kočkovitá šelma
	Ahoj, jak dělá kočkovitá šelma?	kočkovitá šelma

(:) \$0

Funkce

- Vypočítá vložený příklad v podnětu.
- @ musí obsahovat \$p – zjištění předmětu.

Příklad použití: \$0

(:) \$W:soubor (v 3.2)

Funkce

- Ve složce na adrese souboru vytvoří nový znalostní soubor s novým číslem tak, že do tohoto souboru zkopíruje data ze souboru x.txt.
- *využití modulu hlavního správce souborů*

Příklad použití: \$W:znalosti/lide

- Ve složce *znalosti/lide* vytvoří soubor s novým číslem, do kterého zkopíruje soubor x.txt

3.6 Vyhledávací

(:) \$q:soubor1:soubor2-?+d?:d?

Funkce

- V souboru 1 vyhledá řetězec, který se shoduje s řetězcem v podnětu a podle nalezení/nenalezení reaguje a zapíše.

Příklad použití: \$q:znalosti/krestni_jmena:lide/x-1 +d53:d0

- V souboru „znalosti/krestni_jmena“ prohledá jednotlivé řádky a porovná je se slovy v podnětu.
- V případě, že **najde shodu**:
 - zapíše daný řetězec do souboru „lide/x“ na 1. řádek
 - přidá odpověď datového souboru „d53“.
- V případě nenalezení shody odpoví datovým souborem „d0“.

(:) \$q!zdroj:oddíl:znak (v 3.1)

Funkce

- Ve zdroji vyhledá oddíl a do něj zapíše předmět, který začíná zadaným znakem.

Příklad použití: \$q!lide/x:jmeno:- (\$p = honza)

- V souboru „lide/x“ vyhledá oddíl „jmeno“.
- Přidá řádek s obsahem „-honza“.

(:) \$k:zdroj:znak:oddíl+d?:d?

Funkce

- Ve zdroji vyhledá oddíl, který se shoduje s řetězcem v podnětu a podle nalezení/nenalezení následujícího slova s parametrem reaguje.

Příklad použití: \$k:lide/x:-jmeno+d53:d0

- V souboru „lide/x“ vyhledá oddíl „jmeno“
- Pokud pod tímto řetězcem existuje řetězec s parametrem „-“, reaguje „d53“
- Pokud pod tímto řetězcem neexistuje řetězec s parametrem „-“, reaguje „d0“

(:) \$s:soubor:řádek:pád

Funkce

- Ze souboru si zjistí slovo, které je na daném řádku. Poté dané slovo vyhledá ve slovníku a podle posledního čísla zjistí pád daného slova a to slovo v příslušném pádě vypíše.

Příklad použití: \$s:lide/x:1:5

- V souboru „lide/x.txt“ nalezne řetězec 1. řádku (což je křestní jméno).
- Poté otevře soubor ve slovníku, podle prvního písmena (aby prohledávání bylo rychlejší).
- V souboru ze slovníku vyhledá dané slovo a zjistí, na kterém řádku se nachází.
- Podle vyhledaného řádku zjistí slovo, které je v daném pádě daného slova a to slovo vypíše.

(:) \$s!slovo:pád

Funkce

- Zadané slovo vyhledá ve slovníku a podle posledního čísla zjistí pád daného slova a to slovo v příslušném pádě vypíše.

Příklad použití: \$s!honza:5

- Otevře soubor ve slovníku, podle prvního písmena (aby prohledávání bylo rychlejší).
- V souboru ze slovníku vyhledá dané slovo a zjistí, na kterém řádku se nachází.
- Podle vyhledaného řádku zjistí slovo, které je v daném pádě slova „honza“ a to slovo vypíše.

(:) \$s;pád (v 3.1)

Funkce

- Slova z předmětu vyhledá ve slovníku a vypíše v zadaném pádě.
- V případě nenalezení vypíše jen \$p.
- Pro vyhledávání správného řetězce je potřeba umístit do řádku @ funkci „\$p“

Příklad použití: \$s;5 (\$p = honza)

- Vypíše 5. Pád slova honza: „honzo“.

(@) \$S:soubor

Funkce

- Za tuto funkci dosadí postupně všechna slova ze souboru.
- V případě více takových funkcí v jednom řetězci @ je všechny zkombinuje.
- V případě, že v řádku je více slov, tak je podle mezer rozdělí (aby mohl porovnávat s podnětem)

Příklad použití: \$S:strukтуры/pozdrav

- Přidá všechna slova ze souboru pozdrav.

(:) \$V!znaksoubor:oddíl+d?:d?

Funkce

- V zadaném **souboru** vyhledá vyhledávaný oddíl.
- Pod tímto oddílem vyhledá řádky začínající daným znakem.
- Porovná řádky s předmětem vyhledávání
- V případě, že **znak=?**, nebude na znaku záležet
- Pro vyhledávání správného řetězce je potřeba umístit do řádku @ funkci „\$p“

Příklad použití: \$V!-osobnost/x:mám_rád+d133:d134 (předmět je „rohlík“)

- Otevře soubor „osobnost/x.txt“.
- Vyhledá oddíl „mám_rád“ a zjistí všechny řádky, které začínají znakem „-“.
- Zjištěné řádky porovná s předmětem vyhledávání, se slovem „rohlík“
- Pokud v podnětu nalezne shodný řetězec, vypíše reakci d133. Pokud ne, vypíše reakci d134.

(:) \$V:složka:oddíl+d?:d? (v 3.4)

Funkce

- V zadané **složce** prohledá číselné soubory a vyhledá oddíl a v něm vyhledávaný řetězec.
- Pro vyhledávání správného řetězce je potřeba umístit do řádku @ funkci „\$p“

Příklad použití: \$V:znalosti:název+d54:d55

- Prohledá číselné (oddílové) soubory a názvech oddílů hledá „název“
- Pokud oddíl nalezne, hledá, jestli se v něm nachází řetězec s předmětu.
- Pokud nalezne shodný řetězec, vypíše reakci d54 a soubor zkopíruje jako **x.txt** (právě používaný)
- Pokud nenalezne, vypíše reakci d55.

(:) \$C:složka:oddíl1+oddíl2=d?:d? (v 3.5)

Funkce [Compare - porovnat]

- V zadané **složce** prohledá číselné soubory a vyhledá oba oddíly. Tyto oddíly poté porovná se slovy v předmětu vyhledávání, jestli se v obou vyskytuje alespoň jeden řetězec.
- Pro vyhledávání správného řetězce je potřeba umístit do řádku @ funkci „\$p“

Příklad použití: \$C:znalosti:název+vlastnost=d159:d160 [štěká pes?] > předmět je „štěká pes“

- Prohledá číselné (oddílové) soubory a názvech oddílů hledá „název“ a „vlastnost“.
- Vytvoří kombinační řetězce: štěká; pes; štěká pes
- Vyhledá soubory, které obsahují dané dva oddíly.
- Poté v nalezených souborech prohledá tyto dva oddíly a porovná slova ze seznamem kombinací.

- Pokud se i tam vyskytuje jeden případ, použije datový řetězec d159 (kladná reakce). Pokud ne, použije d160 (záporná reakce).

3.7 Práce s obrázky

\$od:zdroj:cislo1:cislo2

Funkce

- vybere z databáze soubor, ze kterého vybere odkazy na **náhodné obrázky**, které stáhne z internetu a zobrazí je.

Příklad použití: \$od:obrazky/auta:10:3

V souboru „obrazky/auta.txt“ vybere 10 náhodných odkazů a postupně každý z nich zobrazí na 3 sekundy.

\$od!zdroj:cislo1:cislo2

Funkce

- Ze zdroje vybere odkaz na řádku číslo „cislo1“ a obrázek zobrazí po dobu „cislo2“ sekund.

Příklad použití: \$od!obrazky/auta:10:3

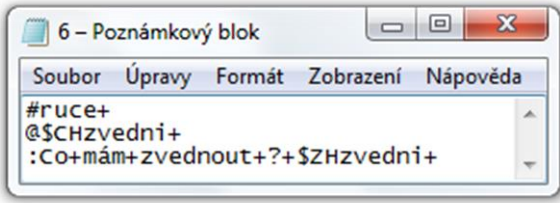
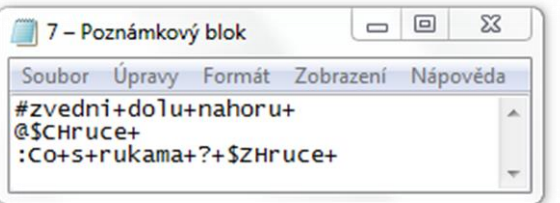
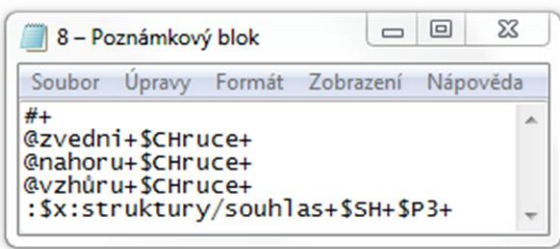
V souboru „obrazky/auta.txt“ vybere odkaz na 10. Řádku a ten stáhne a zobrazí

5 Podrobný popis jednotlivých funkcí

5.1 Práce s historií a její využívání

Pro práci a využití historie slouží tři základní funkční prvky, které jsou uloženy v Obecných datových souborech. Těmi prvky jsou: \$CH (čtení historie), \$ZH (zápis historie), \$SH (smazání historie).

Příklad použití pro příkaz: „zvedni ruce“:

 <pre>#ruce+ @\$CHzvedni+ :Co+mám+zvednout+?+\$ZHzvedni+</pre>	<p>Podnět nesmí obsahovat slova: ruce</p> <p>Podnět musí obsahovat slova: zvedni, slovo z historie</p> <p>Reakce je následující: Co mám zvednout? + zápis do historie slovo „zvedni“.</p>
 <pre>#zvedni+dolu+nahoru+ @\$CHruce+ :Co+s+rukama+?+\$ZHRuce+</pre>	<p>Podnět nesmí obsahovat slova: zvedni, dolu, nahoru</p> <p>Podnět musí obsahovat slova: Ruce, slovo z historie</p> <p>Reakce je následující: Co s rukama? + zápis do historie slovo „ruce“.</p>
 <pre>#+ @zvedni+\$CHruce+ @nahoru+\$CHruce+ @vzhůru+\$CHruce+ :\$X:struktury/souhlas+\$SH+\$P3+</pre>	<p>Podnět nesmí obsahovat slova: ---</p> <p>Podnět musí obsahovat slova: Zvedni, ruce, slovo z historie Nebo další řádky...</p> <p>Reakce je následující: Dobře + smazání historie + pohyb zvednutí rukou nahoru</p>

Varianta A

Podnět je „zvedni“ => Soubor 6 je 100%, soubor 8 je 50% - vybere reakci souboru 6.txt

Podnět je „ruce“ => Soubor 7 je 100%, soubor 8 je 50% - vybere reakci souboru 7.txt

Varianta B

Podnět je „zvedni“ a poté je podnět „ruce“

=> Soubor 6 je 100%, soubor 8 je 50% - vybere reakci souboru 6.txt

=> Soubor 8 je 100% - vybere reakci souboru 8.txt

Varianta C

Podnět je „ruce“ a poté je podnět „zvedni“

=> Soubor 7 je 100%, soubor 8 je 50% - vybere reakci souboru 7.txt

=> Soubor 8 je 100% - vybere reakci souboru 8.txt

Varianta D

Podnět je „zvedni ruce“ => Soubor 8 je 100% - vybere reakci souboru 8.txt

Může vyvstat otázka: „Proč je v souboru 6 a 7 použito \$CH? Je to z toho důvodu, že pokud se v historii bude nacházet slovo, které podnět nesmí obsahovat, aby danou reakci nepoužil.“

5.2 Práce s datem a časem + jeho využívání

Aktuální datum a čas jsou automaticky pravidelně ukládány do znalostních souborů uložených v adresáři „**data/znalosti/**“.

Primárně využívané soubory, kde jsou uloženy číselné hodnoty:

- data/znalosti/aktualne_hodiny (0-24)
- data/znalosti/aktualne_minuty (0-59)
- data/znalosti/aktualni_den_v_mesici (1-31)
- data/znalosti/aktualni_den_v_tydnu (1-7)
- data/znalosti/aktualni_mesic_v_roce (1-12)
- data/znalosti/aktualni_rok

Sekundárně využívané soubory, kde jsou uloženy doplňující informace:

- data/struktury/dny_v_tydnu (pondělí, úterý, ...)
- data/znalosti/aktualne_mesice (leden, únor,...)

Příklad řetězce odpovědi na otázku „co je dnes za den?“:

dnes+je+\$*znalosti/aktualni_den_v_tydnu:struktury/dny_v_tydnu+,+\$x:znalosti/aktualni_den_v_mesici+.+\$*znalosti/aktualni_mesic_v_roce:struktury/mesice+\$x:znalosti/aktualni_rok+

=> dnes je pondělí, 19. červenec 2013

5.3 Funkce vyhledávání

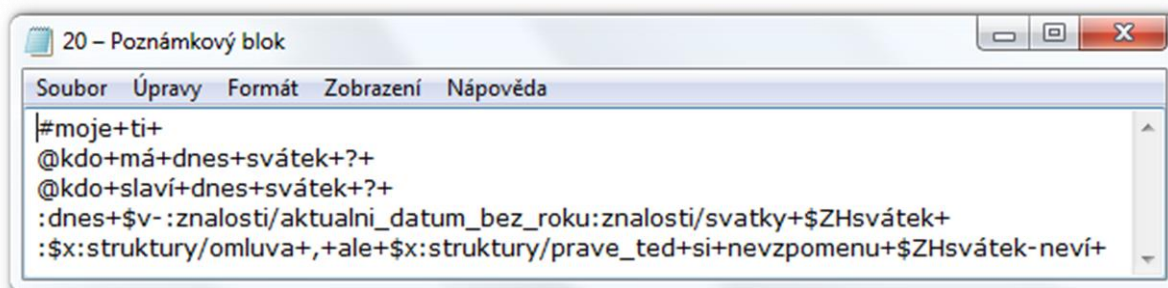
Parametr funkce: \$v?:zdroj:cíl_vyhledávání

- Za „?“ se dosazuje znak, podle kterého program vyhledá správný řetězec.

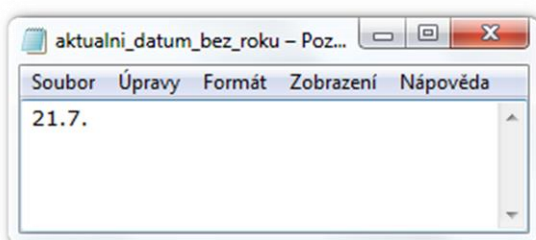
Parametr funkce: \$v?!zdroj:cíl_vyhledávání

- Nahrazením znaku „!“ za znak „!“ se vygeneruje náhodné číslo z daného čísla ze zdroje. Včetně nuly.

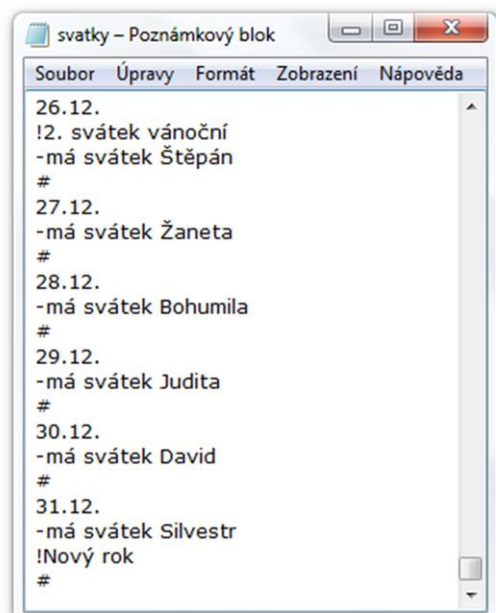
Příklad použití pro podnět: „Kdo má dneska svátek?“



(Vyhledávání podle pomlčky)




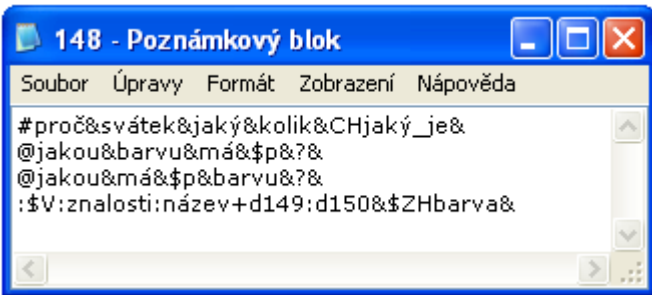
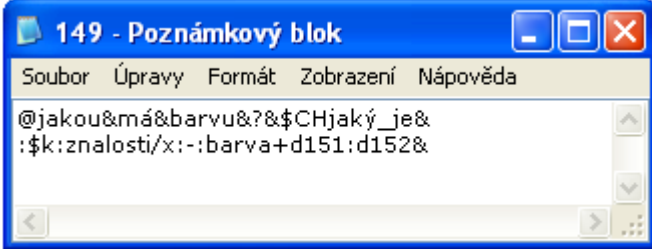
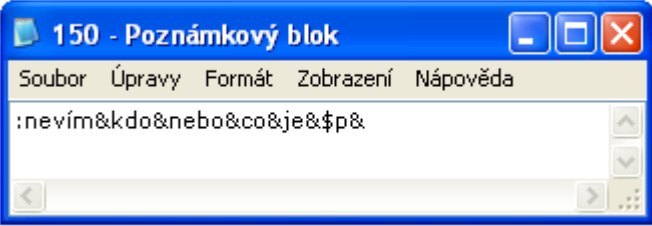
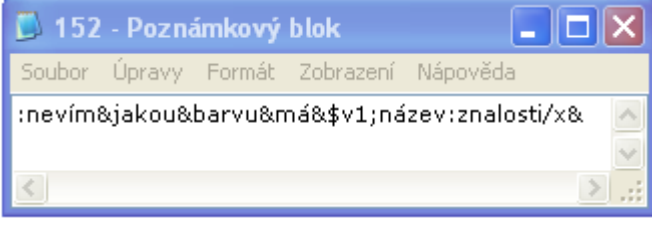
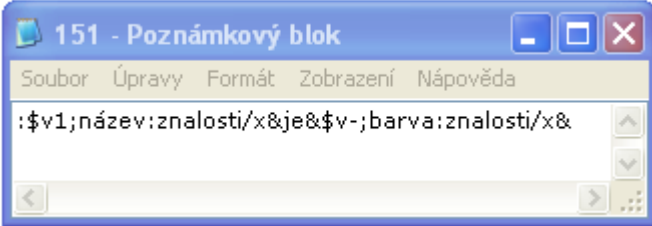
Je vytvářen automaticky a automaticky pravidelně aktualizován.



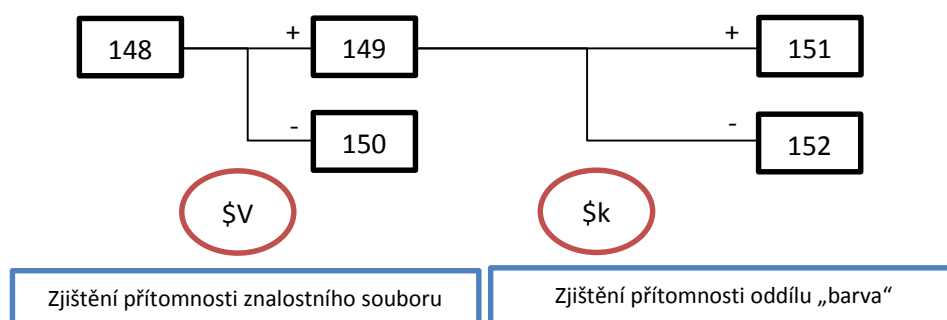
- Seznam musí být ukončen znakem „#“
- Podle data vyhledá datum
- Podle vyhledávacího znaku (pomlčky) vyhledá odpověď.
- V případě více odpovědí je seřadí za sebou a proloží spojkou „a“ (dnes mají svátek Petr a Pavel)
- V případě nenalezení odpovědi zůstane řetězec prázdný.

5.3.1 Vyhledávání znalostí s kombinací funkcí \$V a \$k

Používané soubory

Vyhledávaný soubor	Datové soubory
	 
 	

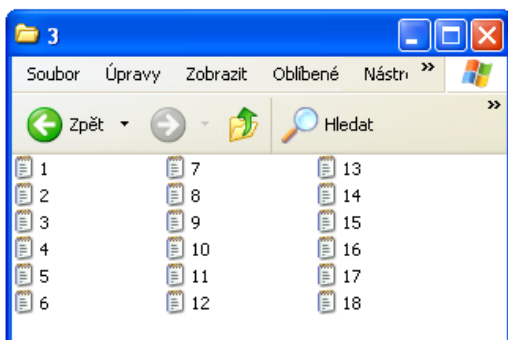
Logické propojení datových souborů



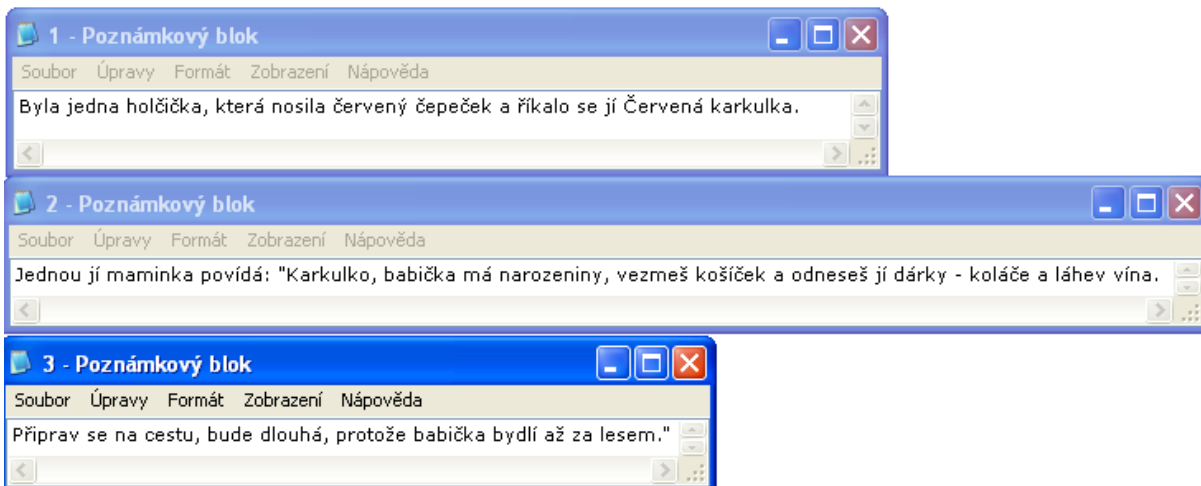
5.4 Sestavení příběhu

5.4.1 Potřebné soubory:

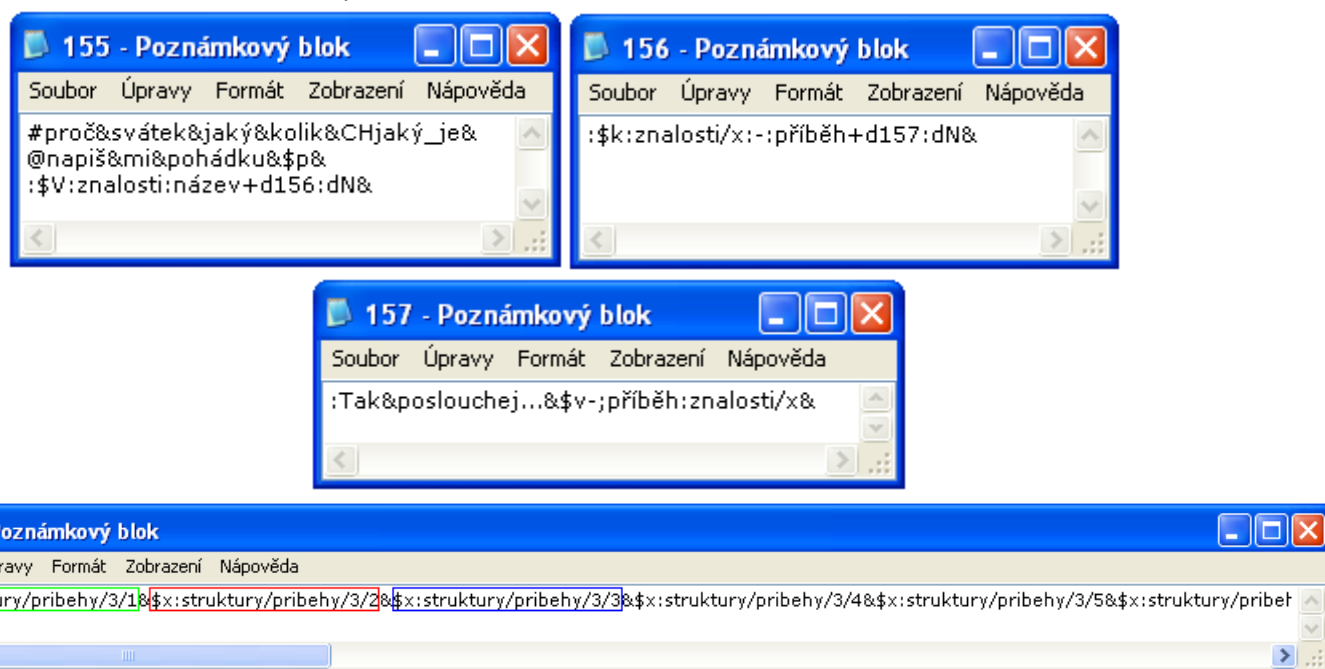
- a) Data s příběhem – uloženy např. ve složce **data/struktury/pribeh**



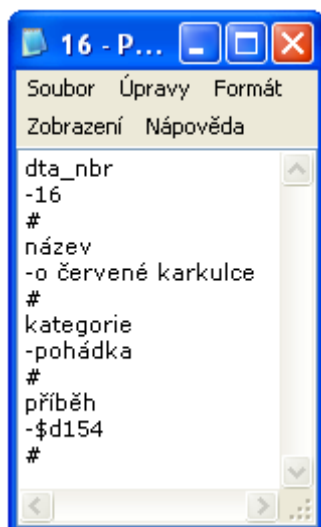
V každém souboru jsou uloženy jednotlivé části příběhu. V každém může být i více řádků, ze kterých program podle zadané funkce vybere buď konkrétní řádek, nebo náhodný.



- b) Základní datové soubory



c) Znalostní soubor



5.4.2 Princip fungování

Na základě základních datových souborů 155, 156 a 157 provede analýzu vyhledání znalostí podle kapitoly „Vyhledávání znalostí s kombinací funkcí \$V a \$k (kapitola 5.3.1)“.

V případě nalezení dané znalosti (název) a parametru (pomlčka) v příslušném oddílu (příběh) znalostního souboru (16.txt) vypíše příslušný text – v tomto případě použije k vypsání datový soubor s číslem 154.

V tomto soubor jsou použity v sérii funkce \$x:, které postupně vyhledávají v daných strukturovaných souborech náhodné řádky a ty vypisují. Jednotlivé soubory na sebe musí slovosledně navazovat (aby byl příběh srozumitelný).

Výsledek:

