



Návrh softvérovej vrstvy pre používateľské rozhranie  
podporujúce nevidiacich ľudí pri práci s ambientným systémom

# RUDO

Milan Hudec a Ján Karabáš





Fakulta prírodných vied Univerzity Mateja Bela  
v Banskej Bystrici

Milan Hudec a Ján Karabáš

Návrh softvérovej vrstvy pre používateľské  
rozhranie podporujúce nevidiacich ľudí pri práci  
s ambientným systémom

**RUDO**



Banská Bystrica  
2018

Autori: RNDr. Milan Hudec, PhD.  
doc. Mgr. Ján Karabáš, PhD.

Recenzenti: doc. RNDr. Gabriela Andrejková, CSc.  
Mgr. Ing. Zdeněk Smutný, PhD.  
prof. Ing. Dušan Šimšík, PhD.

Vydavateľ: Belianum. Vydavateľstvo Univerzity Mateja Bela  
v Banskej Bystrici

Edícia: Fakulta prírodných vied

*Elektronická edícia, verzia 1.1 (5. februára 2018)*

ISBN: 978-80-557-1392-2

## Obsah

<b>1</b>	<b>Ambientné systémy v inteligentných budovách .</b>	<b>7</b>
1.1	Dôvody a ciele publikácie	10
1.2	História vývoja Aml systému RUDO	12
1.3	Aml systém RUDO	14
1.4	Architektúra systému RUDO	14
1.5	Používateľské rozhranie pre nevidiacich	20
<b>2</b>	<b>Inštalácia OS Linux .....</b>	<b>27</b>
2.1	Inštalácia operačného systému...	28
2.2	Inštalácia a konfigurácia balíčkov	32
2.3	Inštalácia tlačiarne a skenera	34
<b>3</b>	<b>Inštalácia Aml systému... ..</b>	<b>37</b>
3.1	Automatická inštalácia	38
3.2	Konfigurácia	39

3.3	Postup inštalácie Aml systému RUDO na server	41
<b>4</b>	<b>Adresáre, skripty, príkazy...</b>	<b>43</b>
4.1	Adresáre a skripty	44
4.2	Príkazy	46
<b>5</b>	<b>Skripty modulu ROWS</b>	<b>55</b>
<b>6</b>	<b>Programy modulu ROWS a Aml RUDO</b>	<b>67</b>
6.1	Bežné programy	68
6.2	Vývojové nástroje	72
6.3	Programy k publikácii...	74
6.4	Funkcie asistenčného softvéru GOBTLK	75
<b>7</b>	<b>Spracovanie textu</b>	<b>79</b>
7.1	Práca so špeciálnym editorom	80
7.2	Práca s hypertextom a tlač dokumentov	81
<b>8</b>	<b>Dáta, čísla a zvuk</b>	<b>93</b>
8.1	Kalendár a zápisník	93
8.2	Programovateľný kalkulátor	95
8.3	Práca s nahrávkami	107
<b>9</b>	<b>Záver</b>	<b>111</b>
9.1	Pokrytie práce na príkazovom riadku	112
9.2	Pokrytie služieb pre aplikácie	112
9.3	Pokrytie služieb pre komandér	113
9.4	Pokrytie služieb pre Aml systém RUDO	114
9.5	Záver	114
<b>10</b>	<b>Skratky a pojmy</b>	<b>115</b>
	<b>Zoznam bibliografických odkazov</b>	<b>117</b>

## 1. Ambientné systémy v inteligentných budovách

Klasické ponímanie komfortu pri bývaní bolo v minulosti zo stavebného hľadiska obsiahnuté v optimálnom navrhnutí architektúry, polohy budovy, vykurovacieho systému, rozvodu vody, elektriny a telefónnych liniek.

Pri ďalšom zvyšovaní komfortu sa prihliadalo na kvalitu vnútorného zariadenia a elektrických spotrebičov.

Súčasná doba prináša s rozvojom informatiky nové ponímanie komfortu pri bývaní.

Stavebnou súčasťou budov sa stali dátové rozvody počítačových sietí. Sieť býva dostupná aj pomocou bezdrôtovej technológie WiFi. Budovy začínajú byť vybavované centrálnym počítačom pripojeným na lokálnu počítačovú sieť. Domový server môže takto vyhodnocovať údaje zo senzorov nainštalovaných v budove a automaticky ovládať najrozličnejšie sofistikované zariadenia.

V domových serveroch moderných budov sa inštalujú informačné systémy, ktoré nazývame ambientné systémy, sú zamerané:

- na vzdialenú správu zariadení,
- na odovzdávanie informácií o stave priestorov,
- na zabezpečenie,

- na ovládanie zariadení a úsporu energií.

Budovy vybavené ambientnými systémami, senzormi a zariadeniami sa nazývajú inteligentné budovy.

Informatizácia budov však neponúka len zvyšovanie komfortu v bežnom ponímaní. Súčasťou ambientných systémov v inteligentných budovách môžu byť aj asistenčné technológie, ktoré pomáhajú pri bývaní starým alebo zdravotne hendikepovaným ľuďom [1, 4, 9, 36].

Súčasný bývanie v asistovanom prostredí (*Ambient Assisted Living*, ďalej AAL) je prioritne zamerané na asistenčné služby pre starších ľudí [1, 2, 4, 26, 35, 36]. AAL systémy sú preto zamerané na pomoc:

- pri onemocnení pohybovej sústavy [5, 25, 28],
- pri slabozrakosti [35],
- pri ľahšej forme stareckej demencie [5],
- pri údržbe hygienických potrieb a zvýšení bezpečnosti [3, 19, 22].

AAL systémy taxonomicky klasifikujú osoby, ktoré zo zdravotného hľadiska vyžadujú asistenčnú podporu [2, 4, 25]. Pri tejto činnosti vyhodnocujú dáta z pohybových senzorov alebo kamier [20, 22].

Pri krízovej klasifikácii systém vyšle podnet na privolanie opatrovateľskej pomoci alebo zdravotníckeho zásahu.

Súčasťou AAL systémov je prispôsobené alebo adaptabilné používateľské rozhranie [8, 35], ktoré môže vhodne kompenzovať zdravotný hendikep.

Môžu byť využité aj doplňujúce služby kontrolujúce hygienické podmienky, ekoparametre – teplota, intenzita svetla a podobne [5].

Poslednou zmienenou vlastnosťou AAL systémov je aj ich ekonomické hodnotenie - náklady na zriadenie, licenčné obmedzenia a náklady na údržbu [5].

V tejto publikácii je predstavená softvérová vrstva modulu ROWS, ktorá bola vyvinutá so zámerom podpory nevidiacich ľudí [6, 23] pri práci s ambientným systémom RUDO (ďalej AmI – **AM**bie*nt Intelligence*).

Modul ROWS je súčasťou tohoto AmI systému, ktorý je na Fakulte prírodných vied UMB v Banskej Bystrici vyvíjaný od roku 1998.



Vytvorená softvérová vrstva modulu ROWS ponúka širokú zostavu programov a skriptov, ktoré sa môžu použiť dvomi spôsobmi:

1. ako jednoduché príkazy používané na príkazovom riadku operačného systému Linux Debian,
2. ako softvérová vrstva používaná semigrafickým používateľským rozhraním pre nevidiacich.

Nevidiacemu používateľovi sa takto môže ponúkať práca s počítačom na administrátorskej úrovni na príkazovom riadku, pričom môžu byť používané príkazy sprostredkované aj iným používateľským rozhraním. Návrh uvádzanej softvérovej vrstvy je zameraný:

1. na bežnú prácu s počítačom,
2. na administrátorskú prácu s počítačom,
3. na odbornú prácu v oblasti informatiky,
4. na bežnú obsluhu zariadení ambientného systému RUDO,
5. na administrátorskú obsluhu ambientného systému RUDO.

Vytvorená softvérová vrstva modulu ROWS bola implementovaná do AmI systému v roku 2010 a testovaná päť rokov v rodinnom dome, kde je AmI systém RUDO prevádzkovaný.

Ide o rodinný dom s dvomi bytmi, pričom v jednom z nich býva rodina – manželia s dcérou, kde muž je nevidiaci.

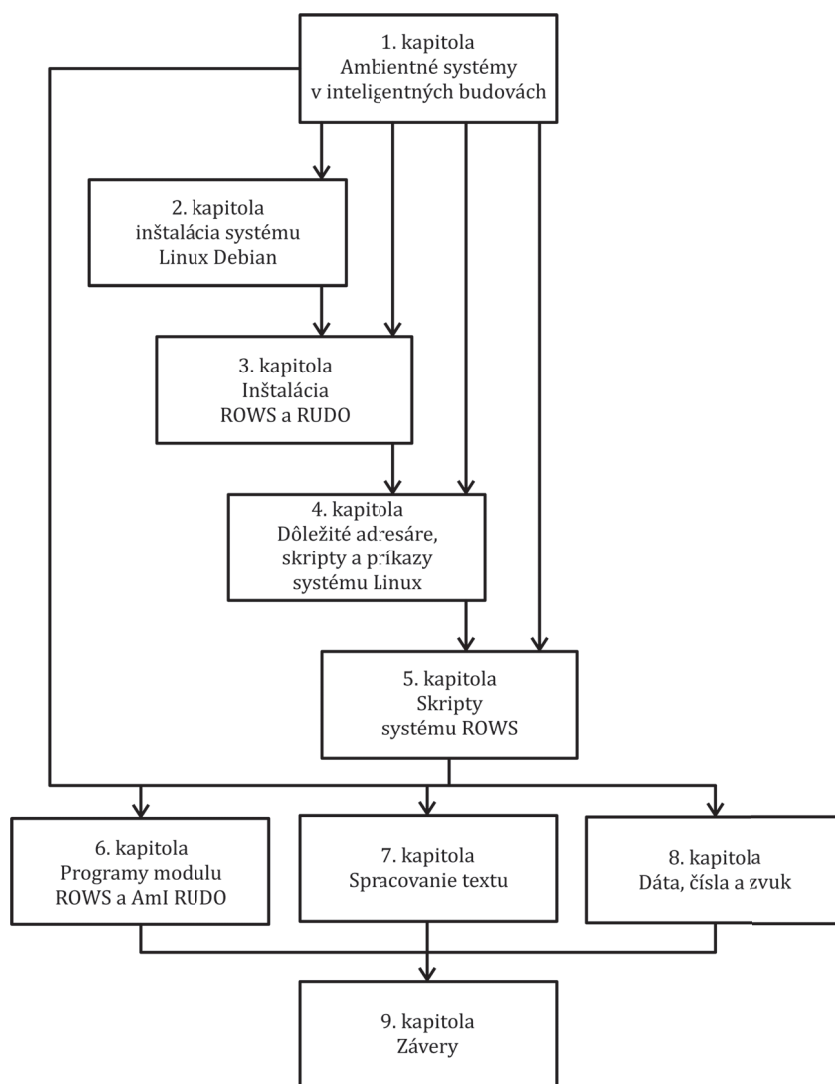
Počas piatich rokov testovania sa od nevidiaceho muža vyžadovala samostatnosť pri obsluhu ambientného systému na bežnej a administrátorskej úrovni.

Dôvody a ciele tohoto výskumu sú uvedené v ďalších podkapitolách. Zároveň sú v nich rámcovo zhrnuté funkcie AmI systému RUDO s dôrazom na špeciálne používateľské rozhranie pre nevidiacich.

Kapitoly 2 a 3 opisujú inštaláciu operačného systému Linux Debian zameranú na prevádzkovanie AmI systému RUDO a automatickú inštaláciu tohoto AmI systému. Kapitoly 4–6 opisujú vyvinutú a testovanú softvérovú vrstvu programov a skriptov modulu ROWS. Kapitoly 7 a 8 opisujú softvérové aplikácie používané pri práci s textom, číslami a zvukom. Závery tohoto výskumu sú zhrnuté v kapitole 9, ktorá zároveň ponúka možnosti ďalších vylepšení.

Nižšie je uvedený tematický rozvojový diagram kapitol tejto publikácie. Prototyp AmI systému RUDO, ktorý bol vyvinutý na akademickej pôde Univerzity Mateja Bela v Banskej Bystrici, je špecifický svojim zameraním na asistenciu pri bývaní pre nevidiacich

ľudí. V nasledujúcej podkapitole je zdôvodnená opodstatnenosť vývoja AmI systémov zameraná na podporu nevidiacich bez vekového obmedzenia.



### 1.1 Dôvody a ciele publikácie

Podľa Svetovej zdravotníckej organizácie má približne 1 až 1.5% populácie vážne problémy so zrakom. Únia nevidiacich a slabozra-

kých na Slovensku (ďalej ÚNSS) má cca 4000 členov so zrakovým postihnutím. Z toho je približne 60% žien a 40% mužov.

Na základe poskytnutých informácií klientov v rámci služieb ÚNSS sa odhaduje približne 16% s neúplným alebo základným vzdelaním, 60% so stredným odborným alebo vyšším pomaturitným vzdelaním a 24% s vysokoškolským vzdelaním.

Orientačne boli oslovení piati nevidiaci respondenti, ktorí v dotazníku vyjadrili svoje potreby vyplývajúce z ich zrakového hendikepu.

Z vyššie uvedených štatistických údajov je vidieť, že AmI systémy sú využiteľné aj pre potreby nevidiacich ľudí. Súčasné prototypy AAL systémov sú však zamerané na asistenčné služby pre starých ľudí alebo pre ľudí s iným zdravotným hendikepom.

Nedostatok v oblasti vývoja AmI systémov pre nevidiacich bol dôvodom na zostrojenie prvého prototypu AmI systému RUDO v roku 2002 [11]. V súčasnosti je systém RUDO testovaný už vo svojej štvrtej verzii v obytných priestoroch dvojpodlažného rodinného domu, v ktorom býva nevidiaci človek so svojou rodinou.

Dominantným cieľom výskumu, ktorý je uvádzaný v tejto publikácii, je návrh, vytvorenie a otestovanie vrstvy asistenčného softvéru, ktorý bude využiteľný a plne postačujúci pri obsluhu počítača a obsluhu AmI systému RUDO nevidiacimi používateľmi. Vyvinuté asistenčné technológie musia postačovať:

1. pri bežnej práci s počítačom,
2. pri administrátorskej práci s počítačom,
3. pri odbornej práci v oblasti informatiky,
4. pri bežnej obsluhu zariadení AmI systému RUDO,
5. pri administrátorskej obsluhu AmI systému RUDO.

Druhým cieľom tejto publikácie je podrobný opis postupu špeciálnej inštalácie domového servra a klientskej stanice tak, aby na týchto počítačoch mohol byť nainštalovaný AmI systém RUDO. Inštalácia a práca na počítači je zameraná na textovú konzolu a príkazový riadok.

Tretím cieľom je opis inštalácie AmI systému RUDO, ktorého súčasťou je modul ROWS používaný nevidiacim človekom pri bežnej obsluhu PC.

Štvrtým cieľom je opis funkcie pomocných skriptov operačného systému Linux, pomocných skriptov a programov patriacich do

modulu ROWS, pomocou ktorých je realizovaná softvérová vrstva špeciálneho používateľského rozhrania pre nevidiacich.

Účelovo nainštalované klientske stanice a domový server budú kvalitným východiskom pre prácu a využívanie asistenčných služieb AmI systému RUDO.

V súčasnosti bola dokončená štvrtá verzia prototypu AmI systému RUDO, ktorá pracuje pod operačnou platformou Linux Debian.

V ďalších podkapitolách bude čitateľ stručne oboznámený s históriou vývoja a funkciami tohoto systému. Do pozornosti je obzvlášť dané používateľské rozhranie pre nevidiacich, ktoré bolo vyvinuté špeciálne pre AmI systém RUDO.

### 1.2 História vývoja AmI systému RUDO

V tejto podkapitole stručne predstavíme históriu vývoja systému RUDO, ktorého začiatok sa datuje do roku 2000.

Ešte pred začiatkom vývoja, ktorý bol smerovaný k ambientným systémom vzniklo v roku 1997 na Fakulte prírodných vied Univerzity Mateja Bela pracovisko zamerané na vývoj softvérových kompenzačných pomôcok pre nevidiacich. Produkty z tohoto obdobia boli neskôr použité ako komponenty pri vývoji AmI systému RUDO.

Prvým podnetom pre vývoj AmI systému s asistenciou pri bývaní bola potreba nevidiaceho človeka v zmysle bezpečnosti rozlíšiť známe prichádzajúce osoby od neznámych.

Prvá verzia preto obsahovala taxonomický systém so senzormi pohybu, ktoré pracovali na princípe hodnotenia pomocou neurónovej siete s adaptačným mechanizmom Back Propagation. O tomto riešení bol publikovaný článok [11].

Ukázalo sa, že takýto domáci asistent je veľmi zaujímavý a pre nevidiaceho človeka využiteľný aj v iných oblastiach. V druhej verzii bol k systému RUDO pripojený digitálny multimeter METEX cez sériové rozhranie RS 232, pre prácu v oblasti odbornej informatiky a elektroniky. V bytovom reproduktore RUDO čítal pomocou vyhľadávacieho systému nastavené elektrotechnické veličiny, rozsahy a merané hodnoty.

V tejto verzii bol zároveň implementovaný audiosystém, ktorý nevidiacemu pomáhal pri dohľade nad malými deťmi v exteriéri.

Prvé dve verzie pracovali na operačnej platforme 32-bitového operačného systému FreeDOS.

V roku 2005 bol začatý vývoj tretej verzie s vlastným syntetizérom „text to speech“ pre operačnú platformu Linux Debian. Tretia verzia bola dokončená v roku 2010, pracovala pod OS Linux, mala navyše syntetizér GOBLIN verzia 1.0, softvérové prostriedky na spracovanie textov pre nevidiacich a ovládač pre klávesnicu, ktorý umožňoval písanie v Braillovom písme pre nevidiacich.

Možnosť písania v Braillovom písme je veľmi dôležitou súčasťou AmI systému RUDO, lebo nevidiacich podporuje v oblasti gramotnosti v bodovom, slepeckom písme.

Nevidiaci človek, ktorý používa vytvorený prototyp ambientného systému trpí navyše čiastočnou parézou pravej ruky. Tento ďalší hendikep mu neumožňuje používanie desaťprstového písania na bežnej klávesnici.

Softvérové predefinovanie funkcie bežnej klávesnice na Braillov mód sa tak stalo pre tohoto človeka jediným možným prostriedkom písania na bežnej počítačovej klávesnici.

Prvé tri verzie systému RUDO neboli ako celok publikované. Publikované boli v slovenskom jazyku iba vybrané časti – viď napr. [13, 14, 15, 16].

RUDO sa ukázal ako veľký pomocník a v súvislosti s rozvíjajúcimi sa systémami domácej automatizácie sa otvorili ďalšie možnosti.

V roku 2016 bol ukončený vývoj štvrtej verzie systému RUDO, ktorá obsahuje navyše automatizáciu vykurovania, zónovú reguláciu a rozvinutejšie softvérové podporné prostriedky pre nevidiacich. Stručné predstavenie tejto novej verzie bolo prezentované v českej vedeckej tlači [17].

Vzhľadom na skutočnosť, že nový hardvér počítača použitý v prototypu v súčasnosti neobsahuje sériové rozhranie RS 232, pripojenie multimetra bolo preprogramované na komunikáciu cez sériové rozhranie USB.

V podkapitolách 1.3 a 1.4 sú predstavené jednotlivé hlavné hardvérové a softvérové komponenty navrhnutého AmI systému RUDO s asistenciou pri bývaní pre nevidiacich vo verzii 4.0 ako celok.

Opis používateľského rozhrania systému RUDO je uvedený v podkapitole 1.5.

Oproti predchádzajúcim verziám systému RUDO patrí medzi hlavné vylepšenia:

- dvojstupňová regulácia vykurovania s rozdelením interiéru na vykurovacie zóny,
- rozvinutejšie používateľské rozhranie,
- rozvinutejšie softvérové a hardvérové prostriedky zamerané na podporu práce nevidiaceho človeka v oblasti odbornej informatiky a elektroniky.

### 1.3 AmI systém RUDO

Pretože je prototyp AmI systému RUDO špecifický svojim určením pre cieľovú skupinu nevidiacich ľudí, svojou činnosťou sa odlišuje od súčasných AAL. Využíva vo väčšej miere klasické ponímanie systémov domácej automatizácie, ktoré sprístupňuje nevidiacim pomocou špeciálneho používateľského rozhrania. Taxonómiu nevyužíva na klasifikáciu nevidiacich, ale naopak na klasifikáciu zvyšných členov domácnosti a hostí [11]. Na základe vyhodnocovania a hlásení pomocou umelo produkovanej reči sa nevidiacim ľuďom čiastočne nahrádza zrak.

Pri vytváraní prototypu AmI systému RUDO bol vývoj asistenčných technológií zameraný na pomoc nevidiacim ľuďom bez vekového obmedzenia. Jedná sa hlavne o asistenciu:

1. pri obsluhu vykurovacieho systému,
2. pri úspore energií,
3. pri rodičovskom dohľade na deti,
4. pri učení sa do školy a vyhotovovaní dokumentov,
5. pri rozpoznávaní prichádzajúcich osôb,
6. pri odbornej práci v oblasti informatiky a elektroniky.

### 1.4 Architektúra systému RUDO

Prototyp ambientného systému - RUDO verzia 4.0, 2016 - je implementovaný na operačnej platforme Linux Debian. RUDO je sieťovo orientovaný produkt pracujúci v rámci lokálnej počítačovej siete. Obsahuje jedenásť nižšie uvedených základných skupín hardvérových a softvérových komponentov.

### 1.4.1 Domový miniserver a router

Na počítači „HP Pro-Liant microserver G7“ je nainštalovaný operačný systém Linux Debian.

Výkon - AMD TURION II neo N54 dual core 2.2 GHz.

Výbava - RAM 2 GB, HDD 250, 6 x USB, 1 x LAN, graphics adapter.

Počítač je dovybavený zvukovým zariadením, ktoré je kompatibilné s ovládačom ALSA pre Linux Debian.

Miniserver je obsluhovateľný štandardným spôsobom pomocou klávesnice a displeja. Pri realizácii prototypu AmI systému RUDO bol však umiestnený mimo obytných priestorov. Preto nebol vybavený klávesnicou a displejom. Je obsluhovateľný len cez vzdialenú správu z klientských počítačových staníc.

Domový miniserver realizuje všetky výpočty, ktoré softvérové servre systému RUDO ponúkajú ako svoje služby.

Súčasťou služieb je aj minimalizačný proces genetického algoritmu, pomocou ktorého sa adaptuje neurónová sieť taxonomického rozpoznávania. Z tohoto dôvodu nie je odporúčaný nižší výkon tohoto počítača.

Pripojenie domového miniservra k routru a ďalším zariadeniam pomocou LAN, USB a audio výstupov je ukázané na blokovej schéme 1.1.

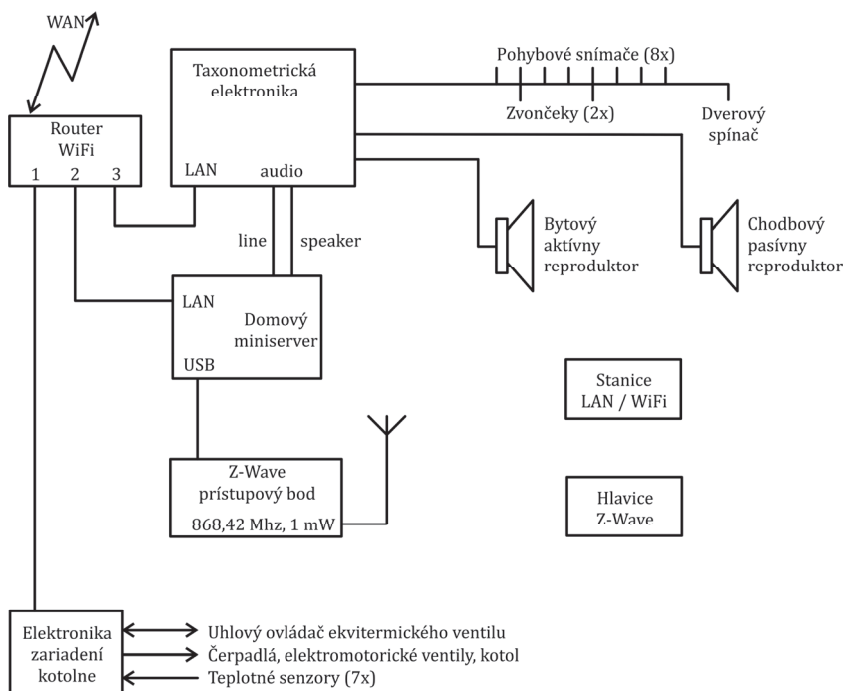
### 1.4.2 Hardvérové V/V zariadenie pripájajúce na lokálnu počítačovú sieť pohybové senzory a domové zvončeky

Elektronika tohoto zariadenia bola navrhnutá špeciálne pre potreby systému RUDO. Zariadenie má svoju IP adresu a je pomocou LAN pripojené k routru (viď 1.1).

Softvérový server komunikuje s týmto zariadením cez sieťové programové rozhranie MODBUS.

Vstupom servra je zber údajov z pohybových senzorov a tlačítok domových zvončekov.

Výstupom servra je podnet na prepínanie domových reproduktorov, pomocou ktorých sa realizujú hlásenia (viď 1.4). Systém RUDO si takto volí lokalitu pre dané hlásenie.



1.1 Bloková schéma AmI systému RUDO

### 1.4.3 Hardvérové V/V zariadenie pripájajúce na lokálnu počítačovú sieť teplotné senzory, elektromotorické ventily, pákový ovládač ekvitermického ventilu a ovládače čerpadiel v kotolni

Elektronika tohoto zariadenia bola navrhnutá špeciálne pre potreby systému RUDO. Zariadenie má svoju IP adresu a je pomocou LAN pripojené k routeru (viď 1.1).

Softvérový server komunikuje s týmto zariadením cez sieťové programové rozhranie MODBUS.

Vstupmi servra sú:

- zber dát z teplotných senzorov,
- zber dát stavu ventilov.

Výstupmi servra sú:

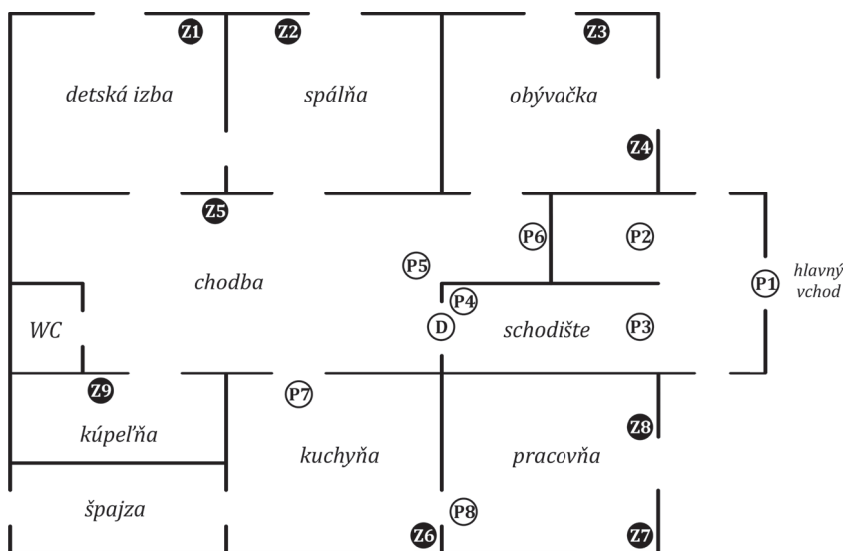
- ovládanie dvoch elektromotorických ventilov,
- ovládanie čerpadiel,
- ovládanie ekvitermického ventilu.



#### 1.4.4 Z-Wave kontrolér

Súčasťou systému RUDO je zónová teplotná regulácia. Teplotná regulácia v jednotlivých zónach sa realizuje pomocou termostatických Z-Wave radiátorových hlavíc. Radiátorové hlavice komunikujú bezdrôtovo so Z-Wave kontrolérom [24, 31], ktorý je cez USB pripojený k domovému miniservru (viď obrázky 1.1 a 1.2).

Softvérový server zónovej teplotnej regulácie, ktorý je nainštalovaný na domovom miniservri, komunikuje s radiátorovými hlavicami cez Z-Wave kontrolér.



1.2 Pôdorys rozmiestnenia senzorov a Z-Wave radiátorových hlavíc; **Zk** - Z-Wave hlavica, **Pm** - pohybový senzor, **D** - dverový spínač

#### 1.4.5 Koncové zariadenia systému RUDO

Na hardvérové V/V zariadenia a Z-Wave kontrolér (viď II., III. a IV.) sú pripojené koncové zariadenia systému RUDO (viď obrázky 1.1 a 1.2), ide o:

- Osem pohybových senzorov,
- dverový spínač,
- 2 x relé na domové zvončeky,
- pákový servoregulátor ekvitermického ventilu,
- dva elektromotorické ventily,
- dve čerpadlá,

- plynový kotol,
- sedem teplotných senzorov,
- 17 termostatických Z-Wave radiátorových hlavíc.

#### **1.4.6 Zariadenia používané na kontrolu detí v exteriéri**

Súčasťou systému RUDO je audioteknika prispôbená na kontrolu malých detí v exteriéri pomocou sluchu. Audioteknika sa zároveň používa na reprodukciu počítačového zvukového výstupu. Táto audio-zostava obsahuje:

- päť kondenzátorových mikrofónov,
- štyri parametrické predzosilovače (spektrálna filtrácia),
- mini mixážne zariadenie,
- zosilňovač,
- reprodukčná sústava.

#### **1.4.7 Softvérový server syntézy hlasu a orientačných zvukov**

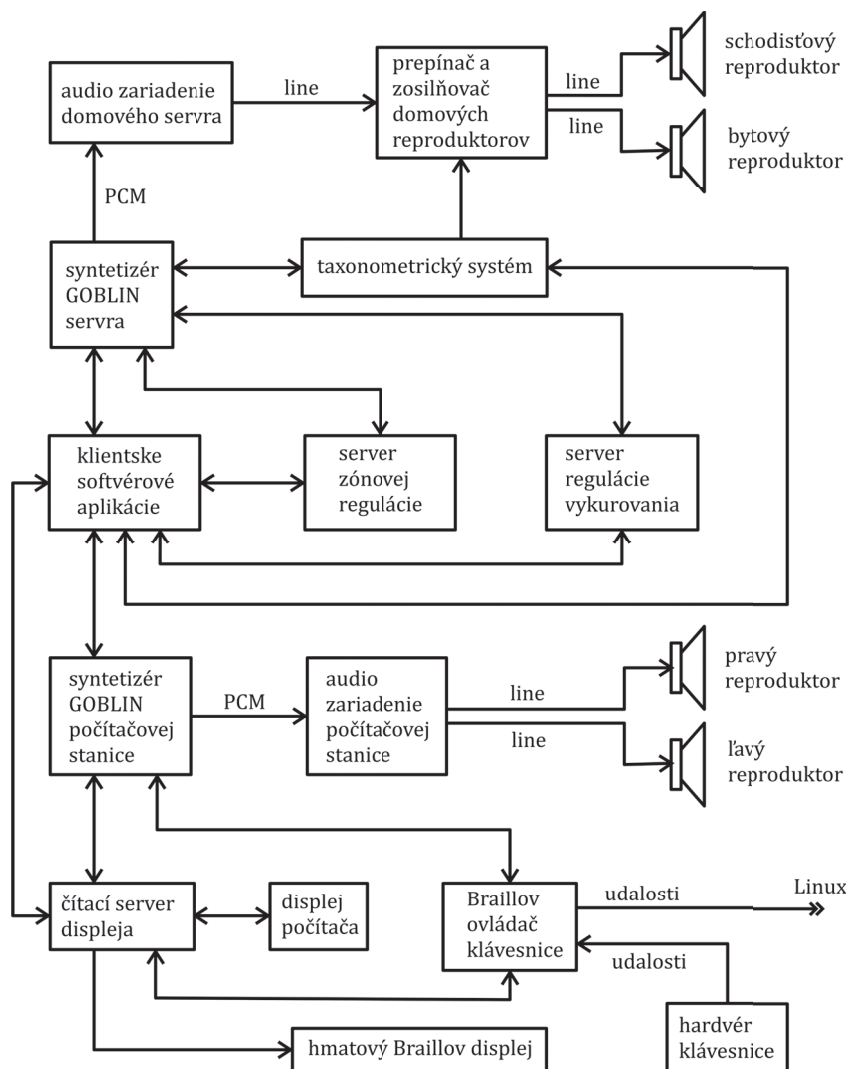
Dôležitou súčasťou používateľského rozhrania pre nevidiacich je umelá produkcia reči [16, 30, 34]. AmI systém RUDO používa svoj vlastný syntetizér – GOBLIN, ktorý je vyvíjaný špeciálne pre potreby nevidiacich ľudí (viď **1.3**). Obsahuje šesť dôležitých súčastí:

- syntéza „text to speech“,
- možnosť viacerých druhov hláskovania,
- označovanie veľkých písmen,
- voľba rýchlosti rozprávania,
- možnosť generovania orientačných a emočných zvukov,
- kompatibilita so softvérovým asistentom pri čítaní.

Syntetizér GOBLIN je nainštalovaný na domovom miniservri a tiež na každom klientskom počítači. Prepojenie softvérových serverov a poskytovanie ich služieb klientským aplikáciám je zobrazené na blokovej schéme **1.3**.

#### **1.4.8 Softvérový server asistencie pri čítaní**

Súčasťou používateľského rozhrania pre nevidiacich je softvérový asistent pri čítaní. Automaticky vyberá z plochy displeja informácie, ktoré majú byť čítané prioritne a ktoré čítané byť nemajú [16], viď **1.3**.



**1.3** Bloková schéma softvérových serverov a poskytovania ich služieb klientským aplikáciám. Neoznačené komunikačné cesty reprezentujú obojsmernú sieťovú komunikáciu.

Zároveň ponúka nástroje na imperatívny výber textu z plochy displeja na čítanie pomocou syntetického hlasu. Imperatívny výber sa vykonáva pomocou klávesnice alebo na základe požiadavky klientskej aplikácie (viď **1.3**).

### 1.4.9 Softvérový server zabezpečenia a taxonometrie

Táto služba informuje nevidiaceho o pohybe pomocou orientačných zvukov. Upozorňuje na únik tepla, ak neboli zatvorené dvere smerom do exteriéru.

Dôležitou súčasťou je rozpoznávanie známych prichádzajúcich ľudí a identifikácia neznámych pomocou pohybových senzorov. Taxonometriu a rozpoznávanie vykonáva neurónová sieť [7, 11, 16, 21, 27, 29, 32], viď **1.3**.

### 1.4.10 Softvérový server obsluhy zariadení kotolne

Systém RUDO obsahuje softvérový automat, ktorý realizuje vykurovací proces a ohrev vody v bojleri (viď **1.3**). Dôraz pri návrhu automatizácie vykurovania bol daný na dve dôležité potreby nevidiaceho človeka:

- úspora energií,
- plná obsluhovateľnosť vykurovania pomocou špeciálneho používateľského rozhrania pre nevidiacich.

V súčasnosti sa totiž nevyrábajú vykurovacie systémy, ktoré môžu byť obsluhované bez potreby zraku.

### 1.4.11 Softvérový server teplotnej zónovej regulácie

Systém zónovej regulácie umožňuje nevidiacemu zorientovať sa v teplotách jednotlivých miestností – zón. Ponúka prostriedky na nastavenie požadovaných teplôt (viď **1.3**).

Dôležitou súčasťou tejto služby sú časové harmonogramy, pomocou ktorých je možné nastaviť vykurovanie v zónach a časoch tak, aby sa kúrilo len v práve používaných miestnostiach. Vytvára sa tak ilúzia celoplošného vykurovania, ktorá bez straty komfortu výrazne šetrí energie. V nasledujúcej podkapitole sú podrobnejšie opísané technológie špeciálneho používateľského rozhrania pre nevidiacich. Systém RUDO využíva toto rozhranie dvomi spôsobmi:

- na osobnú komunikáciu s používateľom pri práci na klientskej počítačovej stanici,
- na verejnú komunikáciu pomocou domových reproduktorov.

## 1.5 Používateľské rozhranie pre nevidiacich

Prvotný rozvoj informatiky priniesol nové možnosti v oblasti kompenzácie zrakového hendikepu. Najnovší vývoj prináša výrobky s

dotykovými displejmi, ktoré zväčša neumožňujú inštaláciu asistenčných technológií.

S takýmito bariérami sa používateľ môže stretnúť pri úplne bežných výrobkoch ako tlačiarne, rádiá, DVD prehrávače, televízory, plynové a elektrické vykurovacie kotle, práčky a mnoho iných výrobkov. Nevidiaci opäť zostáva so svojim hendikepom bezradný a často neschopný integrácie do takého života, aký prináša súčasnosť.

RUDO komunikuje v rámci lokálnej počítačovej siete a Wi-Fi na princípe server/klient. Preto je obsluhovateľný pomocou klientských aplikácií nainštalovaných na počítačoch pripojených meta-licky alebo cez Wi-Fi.

Obsluha je tiež možná pomocou vzdialenej správy cez internet.

Klientské aplikácie sú vybavené asistenčnými technológiami pre nevidiacich, ktoré sprostredkujú informácie pomocou syntetického hlasu a hmatového displeja.

Používateľské rozhranie je prispôbené tak, aby nevidiacim obyvateľom nahrádzalo pohľad na displej. Hlasový výstup sa preto nerealizuje len cez reproduktory klientskej počítačovej stanice. V byte sú nainštalované navyše dva reproduktory, cez ktoré server ohlasuje dôležité udalosti.

Napríklad ak sa vybíjajú batérie v niektorej zo Z-Wave hlavíc, systém RUDO zahlási v bytovom reproduktore stav batérie a umiestnenie radiátora v budove. Hlásenie sa realizuje asi týždeň pred úplným vybitím a opakuje sa viackrát denne. Po jednoduchej výmene batérií sa hlásenie automaticky zruší.

Nevidiaci je takto včas informovaný s presnou špecifikáciou umiestnenia danej hlavice, čím nedôjde k výpadku vykurovania na danom radiátore.

Energia batérií pokryje čas od polovice až po celú vykurovaciu sezónu. Ich spotreba je daná početnosťou automatických teplotných korekcií, teda závisí od polohy daného radiátora v budove.

Vysielanie Z-Wave signálu nie je energeticky zaťažujúce, lebo hlavice pracujú v spiacom režime. Čas pravidelného prebúdzania je daný tzv. wakeup intervalom, ktorý je nastavený na päť minút. Systém RUDO umožňuje tento interval prestaviť až do 30 minút, čím sa spotreba batérií ešte zníži. Teplotné korekcie sa totiž vykonávajú vždy pri prebudení hlavice.

V testovacom prototype bol zvolený najkratší interval prebúdzania z dôvodov empirického otestovania najväčšej spotreby.

Kratší interval prebúdzania zvyšuje rýchlosť reakcie hlavice na podnet používateľa z klientskej aplikácie alebo na automatický podnet zo servra zónovej regulácie.

### 1.5.1 Špeciálny vstup pre nevidiacich

Aby mohol nevidiaci používateľ ovládať AmI systém, je potrebné prispôbiť vstup z bežnej klávesnice na vstup v Braillovom – bodovom písme.

Softvér, ktorý túto funkciu zabezpečuje, číta udalosti bežnej klávesnice zo zariadenia `/dev/input/eventX`, [16, 18, 33] pričom toto zariadenie zakáže pre iné procesy. Zároveň otvorí zariadenie emulácie klávesnice `/dev/uinput` [16, 18, 33] na zápis.

Takouto filtráciou udalostí bežnej klávesnice [16] rozšíri jej funkciu aj na zápis v Braillovom písme pre nevidiacich (viď 1.4–1.7). Rozšírená funkcia klávesnice sa ľahko používateľsky zapína alebo vypína, čo spôsobuje, že táto úprava neobmedzuje vidiacich používateľov.

1 4	f j	f j	1 4
2 5	d k	d k	2 5
3 6	s l	s l	3 6
(i)	(ii)	m a	7 8
		(iii)	(iv)

1.4 V znaku (i) sú k bodom Braillovho písma označené príslušné klávesy na klávesnici. V znaku (ii) sú body Braillovho písma označené príslušným indexom podľa zaužívanej konvencie; znaky (i) a (ii) vyjadrujú 6-bodový Braillov znak. V znaku (iii) sú k bodom Braillovho písma označené príslušné klávesy na klávesnici, písmenom ‚m‘ je označený medzerník. V znaku (iv) sú body Braillovho písma označené príslušným indexom podľa zaužívanej konvencie; znaky (iii) a (iv) vyjadrujú 8-bodový Braillov znak.

Pri rozšírenej funkcii klávesnice sú znaky `asdfjkl`; vnímané ako body Braillovho písma. Preto sa musia tieto klávesy stláčať naraz v príslušných kombináciách (viď obrázky 1.5–1.7).

Takýto filtračný softvér sa inštaluje na klientské počítače, ale aj na domový miniserver. Nevidiaci tak môže vykonávať bežnú obsluhu, ale aj celkovú údržbu systému domácej automatizácie.

Pri programovaní filtra pre nevidiacich pre bežné klávesnice bola zistená ich hardvérová nespôsobilosť pri väčšine výrobcov a typov klávesníc.

Pri klávesniciach je uvádzaný parameter KRO-n (Key RollOver), ktorý určuje, koľko klávesov môže byť naraz stlačených. Pre počítačové Braillovo písmo sa vyžaduje KRO-8, avšak väčšina klávesníc sa pohybuje od KRO-3 až po KRO-6.

V zmysle humanizácie informatiky sa v tejto veci žiada zaviesť nový štandard pre klávesnice. Môžu mať zabudovaný hardvérový prepínač alebo možnosť výberu v BIOSe, ktorý umožní KRO-8, na prácu v Braillovom písme pre nevidiacich. Takýmto systematickým krokom sa môže podporiť gramotnosť nevidiacich ľudí v bodovom - Braillovom písme, ktorá je pre nich veľmi dôležitá.

Test klávesnice na parameter KRO-n je veľmi jednoduchý. V akomkoľvek editore sa naraz stlačia klávesy asdfjkl; . V editore sa zobrazia buď všetky špecifikované znaky, alebo len ich obmedzený počet.

a	b	c	d	e	f	g	h	i	j
⠁	⠃	⠉	⠙	⠑	⠋	⠗	⠈	⠊	⠚
k	l	m	n	o	p	q	r	s	t
⠅	⠇	⠓	⠝	⠕	⠏	⠖	⠞	⠎	⠞
u	v	x	y	z	w				
⠥	⠧	⠭	⠽	⠵	⠺				

### 1.5 Braillovo písmo pre nevidiacich

Prvých desať znakov neobsahuje body s indexami 3 a 6. Druhá desiatka je podobná prvej, ale obsahuje navyše bod s indexom 3. Tretia desiatka je neúplná a podobná druhej, obsahuje navyše bod s indexom 6. Výnimkou je znak ‚w‘.

Čísla sú veľmi podobné prvej desiatke znakov, lebo sú v nich body znížené o jeden bodový riadok nadol, ako je vidieť v **1.6**. Základné Braillovo písmo sa preto veľmi ľahko zapamätá. Je potrebné pamätať si len prvých desať písmen abecedy podľa **1.5**. Braillovo písmo v **1.6** sú kódované podľa štandardu amerického počítačového bodového písma. Kombinácia slovenského Braillovho písma s americkým počítačovým bodovým písmom umožňuje pomocou jedného stlačenia klávesov zápis všetkých znakov – bežné, znaky s diakritikou, číselné znaky, počítačové programátorské znaky.

1	2	3	4	5	6	7	8	9	0
⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠
.	,	:	;	?	!	-	"	+	/
⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠
(	)	{	}	*	\$	'	`	~	
⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠
\	^	[	]	_	@	=	<	>	
⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠	

1.6 Číselné a nealfanumerické Braillove znaky

Navyše umožňuje pri čítaní na hmatovom displeji použitie 6-bodového módu, čím sa enormne urýchli čítanie pomocou hmatu. V 1.7 sú ukázané Braillove znaky s diakritikou, ktoré sú používané v slovenskom a českom jazyku. V znakoch s diakritikou je použitý navyše bod s indexom 8.

Ak chce nevidiaci používateľ zapísať veľké písmeno, pridá ku kombinácii bodov navyše aj kláves medzerník. Znaky s diakritikou budú mať potom navyše body s indexom 7 a 8.

á	ä	č	ď	ě	é	í	l'	ĺ	ň
⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠	⠠
ó	ô	ů	ú	ý	ž				
⠠	⠠	⠠	⠠	⠠	⠠				

1.7 Braillove znaky s diakritikou

### 1.5.2 Špeciálny výstup pre nevidiacich

AmI systém RUDO má implementovaný syntetizér [16, 30, 34] GOBLIN, ktorý môže spolupracovať s filtrom klávesnice pre nevidiacich, s hmatovým – bodovým displejom alebo sa môže použiť pri bežnom sprostredkovaní informácií.



Pri sprostredkovaní informácií z bežného displeja GOBLIN spolupracuje s navigačným - asistenčným softvérovým servrom, ktorý pomáha nevidiacemu orientovať sa na obrazovke a čítať tie informácie, ktoré sú pre neho v danej chvíli dôležité.

GOBLIN je nainštalovaný na klientských počítačoch, kde využíva zvukové zariadenie príslušných počítačov.

Zároveň je tiež nainštalovaný na domovom miniservri. Táto implementácia realizuje hlásenia v bytových reproduktoroch. Je ale možné použiť ju na bežné sprostredkovanie informácií v prípade, že klientský počítač nemá k dispozícii zvukové zariadenie alebo sa toto zariadenie používa na iné účely.

GOBLIN ponúka niekoľko režimov a rýchlostí rozprávania, medzi ktorými je aj niekoľko typov hláskovaní, ktoré sa používajú na spätnú opravu napísaných textov.

V inteligentných budovách môžu bývať napríklad aj rodičia s nevidiacim dieťaťom, ktoré potrebuje asistenciu pri výučbe do školy, pri písaní domácich úloh a podobne.

### 1.5.3 Syntetizér GOBLIN

Pri vývoji technológie syntézy pre AmI systém RUDO boli stanovené dva ciele, ktoré ovplyvnili výber metodiky a priorít.

1. Syntetizér musí byť navrhnutý výslovne pre potreby nevidiacich používateľov počítačov. Znamená to naplnenie šiestich nižšie uvedených bodov:

- zrozumiteľná syntéza „text to speech“,
- možnosť viacerých druhov hláskovania,
- označovanie veľkých písmen,
- voľba rýchlosti rozprávania,
- možnosť generovania orientačných a emočných zvukov,
- kompatibilita so softvérovým asistentom pri čítaní.

Dôraz na hláskovanie a označovanie veľkých písmen je pre nevidiaceho veľmi dôležitý pri spracovaní textov a ich spätných opravách.

Nastavenie rýchlosti rozprávania a zručnosť pri využívaní softvérového asistenta pri čítaní dominantne ovplyvňuje konečnú produktivitu nevidiaceho používateľa.

Orientačné a emočné zvuky používateľa informujú o udalostiach, ktoré sú v danej chvíli pre neho dôležité. Špecifické zvuky sú pritom priradené konkrétnym udalostiam.

2. Druhým cieľom bolo, aby si mohol nevidiaci sám laicky a pomerne rýchlo vytvoriť hlasový korpus [16]. Prakticky to znamená, naučiť počítač rozprávať hlasom napr. svojej manželky alebo dcéry. Zachovanie hlasu blízkeho človeka je zaujímavou náhradou fotografie, ktorú nevidiaci vidieť nemôže.

Pretože vývoj syntézy bol pre systém RUDO dominantný, rozhodujúca bola aj rýchlosť prípravy prvej verzie syntetizéra GOBLIN.

Hlasový korpus bol preto minimalizovaný, pre slovenský jazyk obsahuje 102 častíc hlasu a približne 100 navigačných zvukov.

Takáto syntéza nie je veľmi kvalitná, jadro syntetizéra musí obsahovať generátor koartikulačných prechodov [16, 30, 34], ktoré sa v hlasovom korpuse priamo nenachádzajú.

V druhej verzii, ktorá je v súčasnosti vyvíjaná, sa vychádza z technológie „text to speech“ – tzv. „unit selection“. Keďže ku každej hlasovej vzorke bol priradený jej popisný vektor tak, aby mohla byť dodatočne modifikovaná, rozšírenie bolo nazvané „object selection“ – vzorka a jej vektor vytvárajú objekt.

Aby mohol byť hlasový korpus minimalizovaný, pri syntéze prebieha rozdeľovanie textu na tzv. slabú, strednú a silnú syntézu.

V texte so slabou syntézou sú potlačené koartikulácie.

V texte so strednou syntézou sú dominantné koartikulácie.

V texte so silnou syntézou sú dominantné mikrokoartikulačné javy [16].

Podľa druhu syntézy sa volia na kompozíciu buď fonémy a koartikulačné prechody sa umelo generujú alebo sa volia difóny alebo pri silnej syntéze trifóny.

Pre slovenský jazyk je navrhnutý hlasový korpus, ktorý má niečo vyše 500 hlasových objektov.

Automatické vytváranie hlasového korpusu prechádza jednotlivými vzorkami a nastavuje vhodne ich popisné vektory. Využíva sa pritom spektrálna a keprálna analýza vzoriek.

Pretože do vytvárania hlasového korpusu vstupuje automat, čas prípravy sa skrúti na niekoľko minút. Používateľ – laik – nahrá do mikrofónu asi tri strany A4 špeciálne pripraveného textu, čo trvá približne 1,5 hodiny.

Druhá verzia syntetizéra GOBLIN je vo vývoji. empirické testovanie na čiastkovej verzii hlasového korpusu je však veľmi uspokojivé. Kvalitou sa dostáva niekde medzi kvalitou syntézy difónov a trifónov.

## 2. Inštalácia OS Linux

Prototyp AmI systému RUDO verzia 4.0 2016 bol nainštalovaný na domovom miniservri *HP Pro-Liant microserver G7* procesorom AMD TURION II neo N54 dual core s kmitočtom 2.2 GHz. Použité pamäťové kapacity boli RAM 2 GB a HDD 250 GB.

Domový miniserver realizuje všetky výpočty, ktoré softvérové servre systému RUDO ponúkajú ako svoje služby. Súčasťou služieb je aj minimalizačný proces genetického algoritmu, pomocou ktorého sa adaptuje neurónová sieť taxonomického rozpoznávania. Z tohoto dôvodu nie je odporúčaný nižší výkon tohoto počítača.

Domový server a klientské stanice musia byť vybavené zvukovým zariadením, ktoré je kompatibilné s ovládačom ALSA.

AmI systém RUDO nevyžaduje nainštalované grafické používateľské rozhranie. Nižšie uvádzaná inštalácia je preto riešená bez využitia tohoto grafického rozhrania. Pri inštalácii servra a klientských staníc s AmI systémom RUDO je dôležité nepodceniť dve nižšie uvedené poznámky:

1. Ak bol k danému počítaču dodaný operačný systém Windows, pri inštalácii Linux by sa nemala zničiť záložná partícia s operačného systému Windows a partícia so zavádzačom UEFI.

Inštalácia s duálnym zavádzaním dvoch operačných systémov vyžaduje prísne dodržiavanie podmienok zavádzača UEFI a špecifik dvoch ponúkaných operačných systémov.

2. Inštalovaný server a klientské stanice sú zamerané na používanie ambientného systému RUDO a modulu ROWS. Napriek tomu bola pri návrhu tejto inštalácie vyvinutá snaha o jej všeobecný charakter.

Pri inštalácii Linuxu bez grafického prostredia sa môže stať, že nainštalovaný operačný systém nebude obsahovať niektoré komponenty potrebné pre jeho ďalšie využitie. V takomto prípade sa vyžaduje, aby boli chýbajúce zložky doinštalované neskôr. Súčasná verzia AmI RUDO je nainštalovaná na počítačoch s operačným systémom Linux distribúcie Debian a Ubuntu. Bola testovaná aj s distribúciou OpenSUSE.

## 2.1 Inštalácia operačného systému bez grafického prostredia

AmI systém RUDO a modul ROWS sú vyvíjané so zámerom, aby mohli byť na počítači používané viacerými spôsobmi. Súčasná verzia systému RUDO už obsahuje aj beta verziu semigrafického používateľského rozhrania HANIBAL, ktoré umožňuje nevidiacemu veľmi rýchlu prácu s počítačom a automatický systém výučby obsluhy operačného systému zameraný špeciálne pre nevidiacich. Rôzne spôsoby používania tohoto AmI systému vyžadujú rôzny prístup už pri inštalácii operačného systému:

1. práca na príkazovom riadku bez grafického prostredia,
2. práca v semigrafickom používateľskom rozhraní HANIBAL bez grafického prostredia,
3. práca na príkazovom riadku v kombinácii s prácou v semigrafickom používateľskom rozhraní HANIBAL bez grafického prostredia,
4. Práca na príkazovom riadku s grafickým prostredím,
5. práca v semigrafickom používateľskom rozhraní HANIBAL s grafickým prostredím,
6. práca na príkazovom riadku v kombinácii s prácou v semigrafickom používateľskom rozhraní HANIBAL s grafickým prostredím.

Cieľom tohoto návrhu je, aby mohol nevidiaci používateľ kombinovať veľmi rýchlu prácu vo svojom špecifickom prostredí ako napr.

príkazový riadok alebo semigrafické používateľské rozhranie HANIBAL s prácou pri bežnej počítačovej výbave grafického prostredia operačného systému.

V tejto podkapitole je uvedený inštalačný postup distribúcie Debian alebo Ubuntu, ktorý je špecifický tým, že nebude nainštalované grafické používateľské prostredie. Počítač bude takto určený na obsluhu nevidiacim človekom pomocou používateľského rozhrania HANIBAL alebo na obsluhu pomocou príkazového riadku. Špeciálna inštalácia OS je opísaná v bodoch, ktorých poradie je potrebné dodržať. Špeciálny znak „~“ určuje v operačnom systéme Linux domovský adresár.

### Rozloženie partícií na disku

```
sda1 - /boot  
sda5 - /  
sda6 - /home
```

### Neinštalovať grafické rozhranie!

#### Úprava správy balíčkov

```
# nano /etc/apt/sources.list
```

Pri tomto kroku musia byť už odstránené inštalačné zdroje – cdrom!

1. Zapoznámkovať všetky riadky začínajúce deb cdrom: ...
2. Pridať túto linku:

```
deb http://ftp.debian.org/debian jessie-backports  
main contrib non-free
```

### Aktualizácia balíčkov a reinštalácia nových verzií

```
# apt-get update  
# apt-get upgrade
```

### Inštalácia základných vývojových nástrojov

```
# apt-get install build-essential
```

### Odstrániť poštový server, aby sa nespúšťal pri štarte

```
# update-rc.d -f exim4 remove
```

alebo skôr (pre Jessie)

```
# systemctl disable exim4  
# systemctl mask exim4
```

Prípadne kompletná deinštalácia `exim`

```
# apt-get purge --auto-remove exim4
```

### Nastaviť sudo konfiguráciu

```
# apt-get install sudo
# visudo
```

Na posledný riadok zapísať:

```
<username> ALL=(ALL) NOPASSWD:ALL
```

Namiesto `<username>` zapísať korektné meno používateľa! Na riadok: `Defaults secure_path=...` dopísať dve cesty na začiatok,

```
<installprefix>/bin/rows/sh
```

```
<installprefix>/bin/rows/bn
```

kde `<installprefix>` je `/usr` alebo `/usr/local` podľa toho, aký inštalčný prefix sa použije pri inštalácii ambientného systému RUDO.

### Nastavenie mena počítača

Editovať súbor `/etc/hostname`, (ak neexistuje tak `/etc/HOSTNAME`); do tohoto textového súboru zapísať nové meno a rebootovať.

### Zmena IP adresy

Editovať súbor `/etc/network/interfaces` a po zmene ešte súbory `/etc/hosts` a `/etc/networks` tak aby zodpovedala aktuálna IP a nové `hostname`

### Zrušiť hlášky jadra

```
# nano /etc/sysctl.conf
```

Riadok

```
kernel.printk = 3 4 1 3
```

nemá byť zapoznámkový.

### Konzola a fonty

```
# nano /etc/default/console-setup
```

```
codeset="Lat2"
```

```
FONTFACE="TerminusBold"
```

```
FONTSIZE="24x12"
```

```
SCREEN_WIDTH=80
SCREEN_HEIGHT=28
#VIDEOMODE=
```

**Pozor!** Ak sa nastaví väčší font, ako sa vmestí na displej, jadro zmení počet riadkov a stĺpcov tak, aby sa daný font zmestil! Fonty sa nachádzajú: `ls /usr/share/consolefonts/Lat2* | less`

### **Inštalácia brltty, Braillovho riadku**

(len ak je Braillov displej pripojený)

Odstránenie existujúcej inštalácie

```
# apt-get remove brltty
```

Skopírovať balík `~/rows/targz/brltty2010.tar.gz` do `/opt` Vykonať:

```
# cd /opt
# tar xvfz brltty2010.tar.gz
# cd brltty2010/brltty
# ./configure --prefix=/usr/local --sysconfdir=/etc
# make install
```

Ďalej vykonať:

```
# cp /opt/brltty2010/etc/brltty/* /etc/brltty
# cp /opt/brltty2010/etc/brltty.conf /etc
```

Editovať `/etc/default/brltty` a zapísať

```
arguments="-q -khudec"
start_in_initramfs=true
```

Editovať `/etc/init.d/brltty`

```
daemon="/usr/local/bin/brltty"
```

Ak sa jedná o novší Debian so systémom `systemd`, vykonať:

```
# systemctl daemon-reload
# systemctl enable brltty
# systemctl start brltty
```

## 2.2 Inštalácia a konfigurácia balíčkov

Operačný systém môže ponúkať vysoký komfort pri práci na príkazovom riadku. Široká ponuka používateľských služieb je definovaná postupnosťou inštalácií jednotlivých programových balíčkov, ktoré sú súčasťou OS. Všetky tieto služby sú vyžadované aj semigrafickým používateľským rozhraním HANIBAL.

AmI systém RUDO a modul ROWS obsahujú adresár `~/rows/sh` v ktorom sú uložené všetky skripty tohoto systému. Skript *instpackages* vykoná všetky nižšie uvedené inštalačné príkazy automaticky. Preto je možné najskôr skopírovať adresár "rows" z inštalačného DVD do domovského adresára. Príkaz

```
$ ~/rows/sh/instpackages
```

potom vykoná všetky nižšie uvedené inštalačné body, ktoré sú uvedené v tejto podkapitole.

Napriek tomu môže byť veľmi zaujímavé prejsť si jednotlivé inštalačné kroky manuálne na príkazovom riadku a sledovať rozsah a korektnosť inštalačného postupu. Inštalačné kroky sú opísané v bodoch, ktoré je potrebné vykonať v nižšie uvedenom poradí.

- a) Inštalácia *ALSA* (Advanced Linux Sound Architecture) a *OSS* (Open Sound System), zvukové programové rozhranie:

```
# apt-get install libasound2-dev alsa-base
# apt-get install alsa-oss alsa-utils oss-compat
```

- b) Inštalácia *alpine*:

```
# apt-get install alpine
```

V adresári `/home/<username>` vykonať: (Namiesto `username` je potrebné zapísať korektné meno používateľa!)

```
# touch .pine-passfile
# rm -r mail
# ln -sf ~/rows/mail mail
```

V *Alpine* nastaviť `s-setup`, `c-configuration`:

1. x - Show cursor
2. x - Force arrow cursor
3. x - Compose cancel confirm uses yes
4. x - Send without confirming
5. x - Expunge without confirming
6. x - Expunge without confirming everywhere



- 7. x - Quit without confirming
- 8. Customized headers: From:Meno Priezvisko <meno@xx.yy>

**c) Inštalácia kompilátora fpc:**

```
# apt-get install fpc
```

alebo skopírovať inštalačný balík do /opt, tam potom:

```
# tar xvfz fpc.tar.gz  
# ./install.sh
```

inštalovať do: /usr/local

**d) Inštalácia dosfs, NTFS + páliče:**

```
# apt-get install dosfstools ntfs-config burn  
# apt-get install dvd+rw-tools cdrskin icedax
```

**e) Inštalácia PERL pre skript na obsluhu internetového slovníka:**

```
# apt-get install perl libwww-perl  
# apt-get install liblwp-online-perl  
# apt-get install libio-socket-ssl-perl
```

**f) Inštalácia kodekov MP3:**

```
# apt-get install lame lame-doc
```

**g) Inštalácia internetových prehliadačov:**

```
# apt-get install lynx links
```

**h) Inštalácia kompresorov:**

```
# apt-get install zip unzip mpack
```

**i) Inštalácia nastavovania času z internetu:**

```
# apt-get install ntpdate ntp
```

**j) Inštalácia vývojového balíčka programovania USB**

```
# apt-get install libusb-dev
```

**k) Inštalácia závislosti pre MODBUS**

```
# apt-get install libpcrc3-dev
```

**l) Inštalácia konvertorov pdf,doc,odt na text**

```
# apt-get install libreoffice  
# apt-get install translate-toolkit  
# apt-get install unoconv poppler-utils  
# apt-get install catdoc antiword
```

- m) Inštalácia kontroly pravopisu *aspell* a vývojového balíčka na programovanie

```
# apt-get install aspell libaspell-dev aspell-en  
aspell-cs aspell-sk
```

- n) Inštalácia OCR systému

```
# apt-get install tesseract-ocr tesseract-ocr-sl  
# apt-get install tesseract-ocr-ces
```

- o) Inštalácia obsluhy digitálneho multimetra

```
# apt-get install sigrok sigrok-cli
```

## 2.3 Inštalácia tlačiarne a skenera

V prototype AmI systému RUDO sú preferované tlačiarne a skenery od výrobcu HP. Výrobca týchto zariadení poskytuje plný komfort pri ich inštalácii a odstraňovaní na príkazovom riadku pomocou príkazov ako `hp-*` napr. `hp-setup`. Pri inštalácii a odstraňovaní tlačiarne od iných výrobcov môžu nastať používateľsky komplikovanejšie situácie. Špeciálne tlačové moduly AmI systému RUDO, ktoré asistujú nevidiacemu používateľovi pri vyhotovovaní dokumentov v čiernotlači, majú k dispozícii fonty pre tlačiarne typu:

- PCL3-PCL6 kompatibilné (napr. tlačiarne HP),
- ihličkové tlačiarne EPSON,
- bodové tlačiarne Index Everest,
- font bez ESC-sekvencií pre akékoľvek tlačiarne (pri tomto fonte však nebudú použité rôzne typy a veľkosti písma).

Nižšie sú uvedené kroky v jednotlivých bodoch, ktoré je potrebné vykonať na príkazovom riadku pri inštalácii alebo odstraňovaní tlačiarne alebo skeneru zo systému Linux Debian alebo Ubuntu.

- a) Inštalácia cups + hplip:

```
# apt-get install cups cups-pdf hplip
```

- b) Pri konfigurácii tlačiarne použiť `hp-setup` a editovať súbor `/etc/cups/printers.conf`, prepísať v ňom na začiatku a konci definície danej tlačiarne `Printer` na `DefaultPrinter` a naopak, ak je iná tlačiareň nastavená ako `DefaultPrinter`, je potrebné zmeniť ju na `Printer`. Predvolená tlačiareň môže byť len jedna! Pri konfigurácii tlačiarne pomocou `hp-setup` sa vykonané zmeny prejavia v súbore `/etc/cups/printers.conf` až po prvej tlači! V nasledujúcom príkaze je `w.x.y.z` IP tlačiarne.

```
# hp-setup -i w.x.y.z
```

Pomocou nasledujúceho príkazu je možné rušiť tlačiarne.

```
# hp-setup -i -r
```

- c) Alternatívnu konfiguráciu tlačiarne je možné vykonať pomocou prehliadača



### 3. Inštalácia AmI systému RUDO a modulu ROWS

Modul ROWS je súčasťou AmI systému RUDO. Ak sa využíva mimo prostredia ambientného systému, spadá do kategórie asistenčného softvéru pre nevidiacich. Takýto softvér vytvorí z bežného počítača kompenzačnú pomôcku. Nevidiaci môže potom na počítači písať, čítať, editovať a tlačiť texty, pracovať so zvukom a využívať ďalšie funkcie systému ROWS.

Podobné využitie má ROWS aj v rámci AmI systému RUDO. Služby ambientného systému poskytuje domový server. Na klient-ských staniciach je nainštalovaný ROWS s prepojením na služby ambientného systému.

V tejto kapitole je ukázaný postup, ako na počítač nainštalovať asistenčný softvér ROWS tak, aby sa na tomto počítači dali neskôr spustiť aj služby ambientného systému RUDO.

Na počítači musí byť nainštalovaný Linux Debian alebo Ubuntu podľa kapitoly 2. Na tomto DVD sa nachádza adresár ROWS, ktorý je potrebné skopírovať do domovského adresára a postupovať ďalej podľa nižšie uvedeného postupu.

### 3.1 Automatická inštalácia

AmI systém RUDO s modulom ROWS obsahuje automatický inštalčný resp. deinštalčný skript `instrudo` resp. `uninstrudo`, ktorý sa nachádza v adresároch:

```
~/rows  
~/rows/sh
```

Špeciálny znak `~` určuje v operačnom systéme Linux domovský adresár, do ktorého je potrebné z DVD skopírovať adresár `rows`. Pred zadáním inštalčného príkazu je nutné zvoliť si ho ako aktuálny adresár:

```
$ cd ~/rows
```

Po zvolení uvedeného aktuálneho adresára je používateľ pripravený zadať na príkazovom riadku inštalčný príkaz:

```
$ sudo ./instrudo
```

Inštalčný skript používateľovi zobrazí na začiatku a konci inštalčného procesu niekoľko otázok, na ktoré je potrebné interaktívne reagovať.

Počas inštalčného procesu sa kopíruje väčšie množstvo dát a dochádza ku kompilácii celého AmI systému RUDO. Inštalácia môže preto trvať aj niekoľko minút v závislosti od rýchlosti počítača. Po ukončení inštalácie bude počítač automaticky reštartovaný.

V ďalšom texte tejto podkapitoly je uvedený inštalčný postup v štyroch bodoch, ktoré v krátkosti vyjadrujú korektný používateľský prístup.

- a) Do domovského adresára je potrebné skopírovať podstrom `rows` z DVD.

- b) Ako aktuálny adresár je potrebné zvoliť `~/rows` a v ňom zapísať príkaz:

```
$ sudo ./instrudo
```

- c) Ambientný systém sa automaticky nainštaluje. Deinštaláciu vykoná príkaz zadaný v adresári `~/rows`:

```
$ sudo ./uninstrudo
```

- d) Syntetizér GOBLIN si automaticky načíta sadu znakov displeja s diakritikou. Ak pri automatickom načítaní sady znakov displeja dôjde k nekompatibilite a pri spätnom čítaní sa nesprávne vyslovujú znaky s diakritikou, sadu znakov je možné zvoliť poloaautomaticky podľa nižšie uvedených krokov:

1. Editovať `/etc/rows/rudo.conf` a zmeniť riadok

`actualfont=on`

na

`actualfont=off`

2. V prázdnom adresári, napr. `~/rows/wrk` vykonať príkaz `ft`, ktorý vygeneruje súbor `actualfont.pp`. Tento súbor je potrebné prekopírovať do dvoch adresárov:

`~/rows/out`

`~/rows/p64`

3. Je potrebné nanovo skompilovať celý systém RUDO spustením príkazu `prjg`.
4. Po reštarte by sa mala čítať diakritika správne.

## 3.2 Konfigurácia

Konfigurácia ambientného systému RUDO sa vykonáva editovaním konfiguračného súboru `/etc/rows/rudo.conf`. Nižšie sú uvedené všetky konfiguračné parametre aj s ich krátkym popisom. *V konfiguračnom súbore `rudo.conf` sa nesmie zamieňať poradie parametrov, nesmú sa vynechávať riadky a nesmie sa ani jeden parameter vymazať!*

`djhp.hp`

Font pre tlačiareň

`macros.hpd`

Súbor s makrami pre tlačiareň

`/usr`

Inštalčný prefix ambientného systému RUDO

`/root`

Adresár s dokumentami

`SOUND=ON`

Frekvenčné zvukové signály on/off

39

Číslo naposledy tlačeného dokumentu v `~/rows/tmp`

`lpr64t`

Meno naposledy tlačeného dokumentu v `~/rows/tmp`

/usr/bin/ppcx64

Cesta ku kompilátoru FPC

snd,dan,jan

Vzorky hlasu a navigačných zvukov

goblin=on

Syntetizér Goblin on/off

gobsvr=installed

Syntetizér Goblin installed/uninstalled

gobsvrport=50500

Port pre syntetizér Goblin

oneframe=on

Používané semigrafické rámiky jednočiarové on/off

goblin,zaloha,vyvoj

Gramatické definície

boilerthermostat=60

Teplota nastavená na plynovom kotli

europerm3ofgas=0.455

EUR za  $m^3$  plynu

linfnheating=3.75

Lineárna zložka únikovej funkcie, priama úmera

powfnheating=3.43

Mocninová zložka únikovej funkcie, nepriama úmera

wakeup=180

WakeUp Interval v sekundách (násobky 60)

repagain=10

Počet prístupov na hlavicu pred spasívnením

wkuptmwait=15000

Časový interval očakávania prebudenia hlavice v milisekundách

vendorid=>;6;;jn

Dynamický identifikačný údaj výrobcu

login=4,login:

Identifikácia pri automatickom login

password=5,Password:

Identifikácia password pri automatickom login

prompt=1,~/

Identifikácia prompt pri automatickom vytváraní konzol



blinduser=milan  
Meno nevidiaceho používateľa  
master=100%  
Hlavná hlasitosť  
pcm=50%  
Hlasitosť pcm  
keyboard=1  
Zariadenie klávesnice, s Braillovým módom  
actualfont=on  
Automatické načítanie aktuálneho fontu displeja  
talking=on  
Rozprávanie po štarte počítača on/off  
braille=on  
Braillov mód po štarte počítača on/off

### 3.3 Postup inštalácie Aml systému RUDO na server

Domový server, na ktorom sú nainštalované služby AmI systému RUDO, je pripojený na lokálnu počítačovú sieť, nemá vlastnú klávesnicu a nemá pripojený ani displej. Pre zjednodušenie inštalačného procesu bola zvolená istá uniformita pri inštalácii servra a bežnej počítačovej stanice.

Pri inštalácii servra sa preto na začiatku vyžaduje, aby bola k nemu pripojená klávesnica a displej. AmI systém RUDO sa inštaluje bežným spôsobom podľa podkapitol 3.1 a 3.2.

Zmena bežnej počítačovej stanice na server, sa vykoná až neskôr, editáciou súboru `/etc/rc.local`.

Tento používateľský zákrok vyžaduje administrátorské práva. V súbore `/etc/rc.local` sa nahradí obsah riadku

```
goblinstart autologin
```

novým obsahom

```
rudostart
```

Po uložení nového obsahu súboru `/etc/rc.local` na disk sa bude počítač po reštarte správať úplne inak. Neotvorí sa automaticky päť pracovných textových konzol, aktivujú sa len služby AmI systému RUDO a na displeji sa zobrazí „login“ operačného systému“. Po uložení zmien na disk je preto najlepšie počítač korektne vypnúť a

umiestniť ho fyzicky na miesto domového servra. Server je potrebné pripojiť na napájaciu a dátovú sieť a všetky zariadenia AmI systému. Domový server je týmto postupom pripravený na prevádzku.

Editáciu súboru `/etc/rc.local` je možné vykonať príkazom operačného systému Linux resp. pomocou skriptu modulu ROWS:

```
$ sudo nano /etc/local
```

respektíve

```
$ auto
```

## 4. Dôležité adresáre, skripty a príkazy systému LINUX

ROWS a RUDO sú zamerané prioritne na prácu v textovom režime na príkazovom riadku operačného systému Linux Debian resp. Ubuntu alebo na prácu v semigrafickom používateľskom rozhraní HANIBAL. Aby mohla byť takáto práca plnohodnotná, je potrebné poznať niekoľko dôležitých adresárov, príkazov a skriptov operačného systému.

V rokoch 2010-2016 bol vykonaný prieskum, na základe ktorého sa empiricky vyšpecifikovali dôležité adresáre, skripty a príkazy OS Linux, ktoré sú pre používateľa AmI systému RUDO dôležité nielen na bežnú obsluhu ale aj pri realizácii vývoja tohoto systému. Výsledky prieskumu sú uvedené v tejto kapitole. Používateľ sa môže prioritne zamerať na osvojenie si vyšpecifikovaných príkazov, čo mu uľahčí zoznamovanie sa s OS Linux.

Druhou možnosťou je využitie semigrafického používateľského rozhrania HANIBAL, ktoré obsahuje pri každej voľbe aj stručný popis jej funkcie. Automatické popisy funkcií sa nevidiacemu používateľovi odovzdávajú pomocou syntetickej reči alebo pomocou hmatového displeja. Prostredie HANIBAL nevidiaceho používateľa vyučuje, ako obsluhovať OS Linux zároveň popri jeho používaní.

Semigrafické používateľské rozhranie je navrhnuté tak, aby si mohol sám používateľ voľiť štruktúru pracovných okien a tiež aj ich obsah a funkcie. Nevidiaci si takto môže vytvárať svoje špecifické pracovné okná, v rámci ktorých sa potom nemusí obávať, že napr. rýchlou prácou poškodí nastavenie systému alebo svoju vlastnú prácu v inej oblasti.

Na definíciu štruktúry okien, ich obsahu a funkcií nevidiaci používa rovnomenný jazyk HANIBAL, v ktorom pre každú položku v pracovnom okne musí zadať postupnosť príkazov OS Linux alebo príkazov, ktoré ponúka AmI systém RUDO.

Opäť sa dostávame k opodstatnenosti prieskumu z rokov 2010-2016. Semigrafické používateľské rozhranie HANIBAL má preddefinovanú štruktúru pracovných okien a funkcií, ktoré využívajú len príkazy OS Linux získané na základe uvedeného prieskumu.

Ak chce nevidiaci používateľ zmysluplne meniť funkcie položiek v pracovných oknách, musí si čo najlepšie osvojiť adresáre, skripty a príkazy tejto kapitoly.

K jednotlivým príkazom, skriptom a adresárom je k dispozícii popis v súbore `~/rows/txt/pomoc.txt`, ktorý si môže používateľ prehliadať pomocou skriptu `pomoc` alebo priamo v používateľskom prostredí HANIBAL v domovskom okne zvolením položky *Pomoc*.

## 4.1 Adresáre a skripty

V tejto podkapitole je vytvorený prehľad adresárov a skriptov zostavený na základe prieskumu z rokov 2010-2016. Bežný používateľ nemusí rozumieť, na čo sa väčšina z nižšie uvedených adresárov a skriptov používa. Dôležité sú v prvom rade pre vývojárov AmI systému RUDO. Pritom predpokladáme, že vývoj tohoto systému môžu vykonávať aj nevidiaci vývojári, ktorí používajú špeciálne vývojové nástroje patriace do tohoto AmI systému. Ďalej je uvedený zoznam vyšpecifikovaných adresárov a skriptov.

`/media`

Adresár so zariadeniami `usb`, `sd`, `dvd` a pod.

`/mnt`

Adresár so zariadeniami

`/sbin`, `/usr/sbin`

Adresáre so systémovými príkazmi (staticky linkované spustiteľné súbory)

/bin, /usr/bin

Adresáre so systémovými príkazmi

/lib, /lib32, /lib64

Knižnice, ovládače a skryté programy. Ovládače sú súbory s rozšírením .ko

/usr/share/consolefonts

Adresár s fontami, Lat2\* fonty s lat2 tabuľkou

/lost+found

Informácia o stratených a nájdených sektoroch

/opt

Optional, býva prázdny

/boot

Štartovacie jadro Linuxu

/dev

Abstrakcie zariadení

/proc

Komunikačné rozhranie jadra systému

/sys

Spojený /dev a /proc, novšia verzia

/root

Domovský adresár používateľa „root“

/selinux, /srv

Ochrany

/etc

Konfiguračné súbory Linuxu

/home

Domovský adresár s používateľmi

/tmp

Adresár dočasných súborov

/usr

Všetky nainštalované knižnice, programy a pod.

/usr/local/bin

Používateľské programy

/var

Sieťové pripojenia, logy a pod.

/var/spool/cups  
Tlačené dokumenty  
/etc/printcap  
Tlačové fronty  
/etc/cups  
Konfigurácia tlačiarne, súbor printers.conf  
/etc/rc.local  
Vykoná sa pred spustením LOGIN  
/home/blinduser/.bashrc  
Vykoná sa po prihlásení používateľa <blinduser>

## 4.2 Príkazy

Zoznam príkazov OS Linux, ktorý je špecifikovaný v tejto podkapitole, bol získaný na základe empirického testovania v rámci prieskumu z rokov 2010-2016.

Táto špecifická zostava príkazov OS Linux definuje prvú časť programového rozhrania, pomocou ktorého komunikuje semigrafické používateľské prostredie HANIBAL s operačným systémom. Funkcie položiek v oknách prostredia HANIBAL sa algoritmicky skladajú z postupnosti štyroch typov príkazov, ktoré sú definované v rovnomennom jazyku HANIBAL:

1. zostava príkazov OS Linux z tejto podkapitoly,
2. zostava skriptov modulu ROWS,
3. zostava príkazov modulu ROWS a AmI systému RUDO,
4. zostava interných príkazov prostredia HANIBAL.

Body 1-3 spolu definujú celé programové rozhranie - softvérovú vrstvu, ktorá leží medzi používateľským prostredím HANIBAL a operačným systémom Linux.

Veľmi dôležitou vlastnosťou tohoto programového rozhrania je jeho uniformita pri používaní na príkazovom riadku alebo pri definícii funkcií položiek v oknách prostredia HANIBAL.

Ak si nevidiaci používateľ zvykne na opakované používanie postupnosti príkazov na príkazovom riadku, môže túto postupnosť zapísať do definície v jazyku HANIBAL. V prostredí takto vznikne nová požadovaná funkcia, ktorá sa vykoná jednoduchou voľbou danej položky v okne.

Pri zapnutí počítača s AmI systémom RUDO, ktorý je používaný ako pracovná stanica, sa automaticky otvorí päť pracovných konzol:

1. konzola s prostredím HANIBAL 1,
2. konzola s prostredím HANIBAL 2,
3. konzola s prostredím HANIBAL 3,
4. konzola s poštovým klientom ALPINE,
5. konzola s hodinami, kalendárom a diárom modulu ROWS.

Konzoly 1-3 ponúkajú zároveň prácu aj na príkazovom riadku voľbou príslušnej funkcie v okne prostredia HANIBAL. V tomto prostredí sa tiež ponúkajú služby prehliadania internetu v textovom režime pomocou prehliadačov LINKS alebo W3M. Pretože pracovná stanica vybavená AmI systémom RUDO pracuje na viacerých textových konzolách, uvedieme najskôr dôležitú kombináciu klávesov, ktorá sa v operačnom systéme Linux používa na prepínanie konzol: **Alt** + **F1**, **Alt** + **F2**, ..., **Alt** + **F6**, prípadne **Ctrl** + **Alt** + **F1**, **Ctrl** + **Alt** + **F2**, ..., **Ctrl** + **Alt** + **F6**. Klávesa **F1** znamená prvú konzolu v poradí, **F2** znamená druhú konzolu v poradí atď. V inštalácii distribúcie Debian sa automaticky pri štarte inicializuje šesť konzol.

Ďalej je uvedený zoznam príkazov OS Linux, ktorý bol získaný na základe empirického testovania v rámci prieskumu z rokov 2010-2016. Tento zoznam tvorí prvú časť programového rozhrania medzi OS Linux a používateľským prostredím HANIBAL.

shutdown -h 0

Vypne počítač

reboot, shutdown -r 0

Reštart

nano

Textový editor

sudo <cmd>

Vykoná sa príkaz <cmd> s právami superužívateľa

date, time

Dátum, čas

ls

Výpis (aktuálneho) adresára

ls -l

Výpis adresára s veľkosťami súborov a právami

ls -a

Výpis adresára aj so špeciálnymi súbormi s bodkou na začiatku

ls -d

Nebudú sa vypisovať obsahy podadresárov

```
ls -d */
```

Vypíšu sa len názvy podadresárov

```
sudo lsmod
```

Vypíšu sa zariadenia systému

```
lsusb
```

Vypíšu sa USB zariadenia

```
dmesg
```

Upresnenie stavu USB zariadení

```
uname -r
```

Verzia jadra systému

```
cat
```

Výpis súboru na obrazovku (ako TYPE v DOS)

```
cat /dev/null > súbor
```

Vynuluje obsah súboru bez zmeny jeho atribútov

```
more
```

Výpis súboru na obrazovku po častiach veľkosti obrazovky Napr. ls

| more - listovanie po stránkach

```
less
```

Funguje ako more, ale má aj vyhľadávanie. Pri spustenom less príkaz / a reťazec znakov, vyhľadá daný reťazec, na ďalší výskyt stlačiť n. Ukončuje sa príkazom q.

```
tail
```

Vypíše posledných 10 riadkov súboru

```
tail -n <číslo> <meno>
```

Vypíše posledných <číslo> riadkov súboru <meno>

```
tail -f <meno>
```

Vypisuje pribúdajúce riadky na konci súboru, CTRL-C vypisovanie ukončí

```
uptime
```

Vypíše informácie „system load“, dôležitou informáciou je záťaž procesora „load average: x.xx, y.yy, z.zz“, kde „x“ je aktuálna záťaž, „y“ je priemerná za posledných 5 minút a „z“ je priemerná za posledných 15 minút. Hodnota 1.00 znamená, že procesor spracúva všetky procesy a pritom nespí. Hodnota menšia ako 1,0 znamená, že procesor aj spí, hodnota väčšia ako 1,00 znamená, že procesy musia čakať. Maximálna možná záťaž je 99.



top

Výpis aktuálnych procesov, q ukončenie

vmstat

Výpis stavu pamäte, q ukončenie

rm

Vymazávanie súborov

rm -r <cesta>

Rekurzívne vymazávanie celého podstromu adresárov

cp <meno1> <meno2>

Kopírovanie súborov a adresárov

mv <meno1> <meno2>

Presun súborov a adresárov

cd <cesta>

Zmena aktuálneho adresára na <cesta>

man <nazov>

Manuál k príkazu <nazov>

who

Vypíše prihlásených používateľov

file <súbor>

Určí typ zadaného súboru

ln -s <súbor> <link>

Vytvorí symbolickú linku link na súbor súbor

ln -sf <adresár> <link>

Bezpodmienečne vytvorí symbolickú linku link na adresáradresár

mkdir, rmdir

Vytvorí, zruší adresár

find <cesta> -name <meno>

Najde súbor meno v podstrome <cesta>

which <aplikácia>

Ukáže cestu k danej aplikácii (k binárke programu)

rename 's/\.pp\$//' \*.pp

Odstráni zo všetkých súborov.pp extenziu \. - znak bodka, . - ľubovoľný jeden znak

rename 's/ab/cd/' \*

Reťazec „ab“ sa nahradí „cd“

```
rename 'y/A-Z/a-z/' *
```

Prevedie mená súborov s veľkými písmenami na malé

```
passwd
```

Zmena hesla aktuálneho užívateľa

```
sudo passwd -d <user>
```

Pre používateľa <user> zruší heslo

```
ps -u <meno>
```

Vypíše všetky procesy používateľa <meno>

```
kill -9 <pid>
```

Zruší proces <pid>, ktorý sa dá zistiť pomocou príkazu ps

```
pidof <názovprocesu>
```

Vypíše číslo procesu

```
killall -9 <názovprocesu>
```

Ukončí daný proces

```
env
```

Vypíše premenné prostredia

```
rm /.ssh -r
```

Nový autentifikačný kľúč pre ssh - vymazať adresár /.ssh, alebo v ňom súbor known\_hosts. Ak sa zmenil len jeden server, vymazať príslušný riadok v known\_hosts.

```
/etc/init.d/služba <akcia>
```

<akcia> = Restart, reset, start, stop služby

```
gzip, gunzip súbor.gz
```

Kompresia, dekompresia

```
zip, unzip
```

kompresia, dekompresia

```
tar cvf súbor.tar *v
```

Pakovanie

```
tar xvf súbor.tar
```

Rozpakovanie

```
tar cvfz súbor.tar.gz *
```

Pakovanie s kompresiou

```
tar xvfz súbor.tar.gz
```

Rozpakovanie s dekompresiou

```
dpkg -x balik.deb <cesta>
```

Rozbalí balik.deb do adresára <cesta>, rozbaľovanie debianovských balíkov

```
sudo apt-get update
```

Nahrá nové verzie balíčkov z internetu

```
sudo apt-get upgrade
```

Nainštaluje nové verzie balíčkov do systému

```
sudo apt-get install <menobalicka>
```

Nainštaluje do systému balíček <menobalicka>

```
sudo apt-get remove <menobalicka>
```

Odištaluje zo systému balíček <menobalicka>, ak sa použije voľba  
--purge, vyčistí aj konfigúráky

```
sudo apt-get autoremove
```

Odstráni nepotrebné záznamy a kolízie

```
apt-cache search <retazec>
```

Vyhľadáva balíčky s reťazcom <retazec>

```
apt-cache depends <menobalicka>
```

Závislosti balíčka <menobalicka>

```
sudo modprobe <nazovovladaca>
```

Uloží modul ovládača do RAM a aktivuje ho

```
sudo modprobe -r <nazovovladaca>
```

Odstráni modul ovládača z RAM

```
chmod a+x meno
```

Nastaví atribút spustiteľnosti súboru meno

```
chmod a+w -R <cesta>
```

Nastaví atribút povoleného zápisu pre celý adresár

```
lp <meno>
```

Vytlačí súbor

```
lp <meno> -d PDF
```

Prevedie súbor na súbor.pdf, ktorý sa bude nachádzať v adresári  
~/PDF

```
lp <meno> -d HPLJ
```

Súbor sa vytlačí na tlačiarňu s názvom HPLJ

```
lpstat -p -d
```

Listuje existujúce tlačiarne

```
lpq
```

Zisťuje ID tlačovej úlohy

```
lprm <jobid>
```

Zruší tlačovú úlohu

```
hp-scan -o <meno>.jpg
```

Skenovanie 300 DPI, čiernobielo

```
hp-scan -m color -r 600
```

Skenovanie 600 DPI, farebne -o súbor.png - do súbor.png

```
tesseract <meno1>.png <meno2>
```

OCR systém, prefotený text meno1.png prevedie do textového súboru meno2.txt. S voľbou -l slk pričom použije rozpoznávanie slovenčiny. Na rozpoznávanie češtiny sa použije parameter ces.

```
hp-setup -i w.x.y.z
```

Instalácia skenera a tlačiarne, w.x.y.z je IP adresa zariadenia. Konfiguračné zmeny vykonané pomocou hp-setup sa môžu prejaviť v súbore /etc/cups/printers.conf až po prvej tlači! Pri konfigurácii tlačiarne použiť hp-setup a editovať súbor /etc/cups/printers.conf, prepísať v ňom na začiatku a konci definície danej tlačiarne „Printer“ na „DefaultPrinter“ a naopak, ak je iná tlačiareň nastavená ako „DefaultPrinter“, je potrebné zmeniť ju na „Printer“. Predvolená tlačiareň môže byť len jedna!

```
hp-setup -i -r
```

Odstraňovanie nepotrebných tlačiarň. Odstránenie tlačiarne sa prejaví v súbore /etc/cups/printers.conf až po prvej tlači! Pri konfigurácii tlačiarne použiť hp-setup a editovať súbor /etc/cups/printers.conf, prepísať v ňom na začiatku a konci definície danej tlačiarne „Printer“ na „DefaultPrinter“ a naopak, ak je iná tlačiareň nastavená ako „DefaultPrinter“, je potrebné zmeniť ju na „Printer“. Predvolená tlačiareň môže byť len jedna!

```
scanimage -L
```

Výpis existujúcich skenerov

```
mkfs -t ext3 /dev/sdb1
```

Formátovanie ext3 zariadenia sdb1, len ako superpoužívateľ. Potom v koreni zariadenia zadať príkaz ako superpoužívateľ: chmod 777 . - nastaví práva pre ďalších používateľov

```
mkfs.vfat /dev/sdb1
```

Formátovanie fat32 zariadenia sdb1, len ako superpoužívateľ Pozor! sda je systémový disk! Zariadenie nesmie byť pripojené Parameter -c kontroluje sektory, -n labelname - nastaví menovku disku man mkdosfs - opis všetkých možností

`mkfs.msdos /dev/sdb1`

Formátovanie fat16 zariadenia sdb1, len ako superpoužívateľ. Pozor! sda je systémový disk! Zariadenie nesmie byť pripojené. Parameter `-c` kontroluje sektory, `-n labelname` - nastaví menovku disku. man `mkdosfs` - opis všetkých možností

`fdisk /dev/sdb`

pozor! nie sdb1, lebo chceme prístup k tabuľke partícií. Pozor! sda je systémový disk! Zariadenie nesmie byť pripojené. Primárne partície 1-4, extended od 5 vyššie

`du <cesta>`

Veľkosť podstromu

`df <disk>`

Miesto na disku

`fsck.ext3 -aC /dev/sdb1`

Skontroluje filesystém disku /dev/sdb1 a opraví chyby, disk nesmie byť mountnutý. Ďalšie možnosti: `fsck.vfat`, `fsck.ext4`

`sudo ifup/ifdown eth0`

Zapne/vypne metalickú sieť

`sudo ifup/ifdown wlan0`

Zapne/vypne Wi-Fi

`/sbin/ifconfig eth0`

Vypíše aktuálnu IP adresu

`chown <user>:<group> <súbor>`

Zmena vlastníka súboru na vlastníka `<user>:<group>`

`chown dan:dan adr -R`

Zmena vlastníka adresára na vlastníka dan

`netstat -ap | grep <číslo>`

Vypíše PID a meno procesu, ktorý používa port <číslo>, použiť s právami superpoužívateľa

`unoconv x y súbor.z`

Skonvertuje súbor s formátom z (napr. docx) x - `-d document`, y - `-format=txt` na formát txt alebo iný, parameter `-show` vypíše podporované formáty

`unoconv -l &`

Spustí pred prvým použitím príkazu `unoconv`, tento listener sa volá v skripte `unoconvlistener`

`antiword m.doc > m.txt`

Súbor `m.doc` skonvertuje na `m.txt`

`catdoc m.doc > m.txt`

Súbor `m.doc` skonvertuje na `m.txt`

`catdoc m.xls > m.txt`

Súbor `m.xls` skonvertuje na `m.txt`

`pdftotext m.pdf`

Súbor `m.pdf` skonvertuje na `m.txt`

`sudo ntpdate -u <url>`

Synchronizácia, nastavenie času, `<url>` môže byť napríklad: `ntp.ubuntu.com`

`wget -r -l2 <url>`

Skopíruje s daného miesta na internete obsah, prečíta odkazy v súboroch a skopíruje ich až po druhé vnorenie

## 5. Skripty modulu ROWS

Modul ROWS obsahuje veľa skriptov, ktoré boli vytvorené s cieľom uľahčenia a urýchlenia práce na príkazovom riadku a práce pri definícii funkcií používateľského prostredia HANIBAL. Skripty modulu ROWS je možné rozdeliť do dvoch skupín:

1. skripty vykonávajúce algoritmus príkazov,
2. skripty vykonávajúce len jeden príkaz.

V druhom bode má vytváranie skriptov zmysel len v prípade, keď má príkaz volaný v skripte viac takých parametrov, ktoré nie je potrebné meniť pri volaní skriptu. Znamená to, že používateľ nemusí opakovane zapisovať dlhý príkazový riadok. Zapíše len krátke volanie skriptu a tie parametre, ktoré je potrebné nastaviť zakaždým s inou hodnotou.

Popis funkcií všetkých skriptov sa nachádza v súbore \$HOME ▶ rows ▶ txt ▶ pomoc.txt, ktorý je možné prehliadať pomocou skriptu pomoc na príkazovom riadku alebo v prostredí HANIBAL voľbou položky Pomoc.

Zostava skriptov modulu ROWS definuje druhú časť programového rozhrania, pomocou ktorého komunikuje semigrafické používateľské prostredie HANIBAL s operačným systémom. Funkcie

položiek v oknách prostredia HANIBAL sa algoritmicky skladajú z postupnosti štyroch typov príkazov, ktoré sú definované v rovnomennom jazyku:

1. zostava príkazov OS Linux (viď 4.2),
2. zostava skriptov modulu ROWS,
3. zostava príkazov modulu ROWS a AmI systému RUDO,
4. zostava interných príkazov prostredia HANIBAL.

Body 1–3 spolu definujú celé programové rozhranie – softvérovú vrstvu, ktorá leží medzi používateľským prostredím HANIBAL a operačným systémom Linux.

Veľmi dôležitou vlastnosťou tohoto programového rozhrania je jeho uniformita pri používaní na príkazovom riadku alebo pri definícii funkcií položiek v oknách prostredia HANIBAL.

Ak si nevidiaci používateľ zvykne na opakované používanie postupnosti skriptov modulu ROWS na príkazovom riadku, môže túto postupnosť zapísať do definície v jazyku HANIBAL. V prostredí takto vznikne nová požadovaná funkcia, ktorá sa vykoná jednoduchou voľbou danej položky v okne.

Ďalej je uvedený rozpis všetkých skriptov modulu ROWS, ktoré definujú druhú časť programového rozhrania medzi prostredím HANIBAL a operačným systémom Linux. instrudo

Inštalátor ambientného systému RUDO, nachádza sa v adresári rows, z ktorého sa aj inštalácia začína.

uninstrudo

Deinštalátor ambientného systému RUDO, nachádza sa v adresári rows, z ktorého sa aj deinštalácia začína.

instpackages

Inštalácia potrebných systémových modulov, použiť len v prípade, že niektoré požadované funkcie nefungujú.

adr

Záznamník adries

tel

Záznamník telefónnych čísel

c2w <prvý-track> <počet-trackov>

Prevod audio CD na súbor WAV

m2w <súbor-bez-rozšírenia>

Prevod MP3 na WAV



w2m <súbor-bez-rozšírenia>

Prevod WAV na MP3

burn -C

Kopírovanie CD (program)

brn <rýchlosť> <súbor-bez-rozšírenia>

Napáli obraz, súbor.iso na DVD

iso <súbor-bez-rozšírenia> <adresár>

Vytvorí obraz ISO pre DVD

cddvdiso <súbor-bez-.iso>

Vytvorí ISO obraz z existujúceho média

ensk <anglický-výraz>

Internetový slovník

sken <slovenský-výraz>

Internetový slovník

desk <nemecký-výraz>

Internetový slovník

skde <slovenský-výraz>

Internetový slovník

epdf2slv <súbor>

Internetový slovník, prevedie anglický súbor.pdf na textový súbor.slv  
a spustí editor na čítanie.

dpdf2slv <súbor>

Internetový slovník, prevedie nemecký súbor.pdf na textový súbor.slv  
a spustí editor na čítanie.

enc <cudzí-výraz>

Internetový encyklopedický slovník

syn <slovo>

Internetový synonymický slovník

lpdf <meno>

Vytlačí meno.pdf s obojstrannou tlačou

a4sg <súbor>

Zoskenuje formát a4 do súbor.pdf 150 dpi

a5sg <súbor>

Zoskenuje formát a5 do súbor.pdf 150 dpi

a4sgc <súbor>

Zoskenuje formát a4 do súbor.pdf farebne 150 dpi

a5sgc <súbor>

Zoskenuje formát a5 do súbor.pdf farebne 150 dpi

a4sgd <súbor>

Zoskenuje formát a4 do súbor.pdf duplexne 150 dpi

a5sgd <súbor>

Zoskenuje formát a5 do súbor.pdf duplexne 150 dpi

a4sgcd <súbor>

Zoskenuje formát a4 do súbor.pdf farebne a duplexne 150 dpi

a5sgcd <súbor>

Zoskenuje formát a5 do súbor.pdf farebne duplexne 150 dpi

scf <súbor> <w>, <x>, <y>, <z>

Zoskenuje výrez fotografie farebne 300 dpi do súbor.pdf, zo zdroja urobí výrez – ľavý horný roh  $w, x$ , pravý dolný roh  $y, z$  v milimetroch, rozmer a4 je  $210 \times 297$  mm

ocr meno

Zoskenuje text a prevedie ho do súboru meno.txt. Používa sa OCR systém *tesseract*.

cn+ slovo

Internetový popis kľúčových slov C++

Translate.pm

Slovníkový modul

preloz.pl

Slovníkový skript v jazyku Perl

netdic

Skript slovníka používaný v programoch

alsa

Informácie o programovom rozhraní ALSA

asp

Informácie o programovom rozhraní *aspell*

auto

Editácia /etc/rc.local, pred prihlásením

brc

Editácia ~/.bashrc, príkazy po prihlásení

fnt

Editácia *console-setup*, nastavenie konzoly

goblinstart

Volá sa v /etc/rc.local

rudostart

Volá sa v /etc/rc.local

goblinstop

Volá sa v /etc/init.d/alsa-utils

pomoc

Čítanie tohoto textu

plan

Poznámky k aktuálnej práci

kody

Kategorizácia publikácií

bashex

Príklad skriptu

ct+

Popis príkazov C++

bev

Evanjelická Biblia

bse

Slovenská ekumenická Biblia

bro

Roháčkova Biblia

cfg

konfigurácia ROWS

cpy

kopírovanie súborov, na konci potlesk

dvd+ dvd-

mount / umount CD a DVD

sd+ sd-

mount / umount SD kariet

fab+ fab- fac+ fac- . . . fag+ fag-

mount / umount USB fat32 zariadení

ntb+ ntb- ntc+ ntc- . . . ntg+ ntg-

mount / umount USB ntfs zariadení

e3b+ e3b- e3c+ e3c- . . . e3g+ e3g-

mount / umount USB linux-ext3 zariadení

e4b+ e4b- e4c+ e4c- . . . e4g+ e4g-

mount / umount USB linux-ext4 zariadení

usb+ usb- usc+ usc- ... usg+ usg-

mount / umount USB fat32 neštandardných zariadení

deb dec ded dee def deg

pozri zariadenia sdb, sdc, sdd

eshv

editácia skriptov

txt súbor\_bez rozšírenia

Prehliadanie súboru.hpr a čítanie pomocou syntetizéra Goblin,  
nečíta príkazy HPR

ft

Informácia o aktívnom fonte displeja, zároveň vygeneruje súbor  
actualfont.pp do adresára ~/rows/out, actualfont.pp je súčasťou  
systému ROWS, implementuje aktuálnu kódovú tabuľku fontu.

gobkil

kill syntetizéra GOBLIN

gobrst

reset syntetizéra GOBLIN + kompilácia

goblin parameter [compile]

Kde parameter je

- start - štart syntetizéra Goblin,
- stop - ukončenie syntetizéra Goblin,
- reset - reset syntetizéra Goblin,
- restart - reset syntetizéra Goblin.

Ak sa pri resete syntetizéra použije druhý parameter compile, syntetizér sa pred resetovaním najskôr skompiluje.

hvp x%

hlasitosť slúchadiel / line

kil

skratka sudo killall -9

lbedat

editácia lbe.dat

lin

poznámky k Linuxu

lsa meno

ls -l | more ale najprv adresáre

lsn meno

ls | more ale najprv adresáre

maxv

master hlasitosť maximum

offb

systemový beep vypnúť

p

. p voľba adresára rows/out

s

. s voľba adresára rows/sh

t

. t voľba adresára rows/doc

m

. m voľba adresára rows/mail

wv

. w voľba adresára rows/wrk

u x

. u x voľba adresára /media/usx

d x

. d x voľba adresára /media/dax

pine

poštár *alpine*

pintb

poštár *alpine* s parametrom hesla

ppf <meno>

hľadanie súborov \*.pp pre Pascal

ppt <meno> <reťazec>

hľadanie súborov \*.pp s daným reťazcom

paf <meno>

hľadanie súborov \*.pas pre Pascal

pat <meno> <reťazec>

hľadanie súborov \*.pas s daným reťazcom

icf <meno>

hľadanie súborov \*.inc pre Pascal

ict <meno> <reťazec>

hľadanie súborov \*.inc s daným reťazcom

puf meno

hľadanie súborov \*.ppu pre Pascal

prg  
kompilácia programov ROWS

prj  
kompilácia programových modulov ROWS

prjg  
kompilácia celého systému ROWS

dir meno  
ls -d bez listovania v podadresároch

fcl  
FCL dokumentácia pre Pascal

fpdoc  
dokumentácia k Pascalu

prog  
dokumentácia k programom Pascalu

ref  
referenčná dokumentácia k Pascalu

rte  
dokumentácia k chybovým hláseniam Pascalu

rtl  
RTL dokumentácia k Pascalu

sdn  
vypnutie počítača

tlkcr1  
prečíta sa riadok s promptom a kurzorom

con  
povie číslo aktuálnej konzoly

pp  
kompilácia Pascal s priradením behových chýb k číslam riadkov zdrojového textu

ib <meno>  
Inštalácia programu meno z adresára ~/rows/bin do systému

syncntp  
Synchronizácia, nastavenie času

hwup  
Domáca WIFI up

hwdn

Domáca WIFI down

wwup

Pracovná WIFI up

wwdn

Pracovná WIFI down

hlup

Domáca sieť up

hldn

Domáca sieť down

wlup

Pracovná sieť up

wldn

Pracovná sieť down

adv

Kompilácia testovacích programov zariadenia Advantech USB 4750

adv1, adv2, adv3, advt

Testy zariadenia Advantech USB 4750

dmm1, dmm2, dmm3, dmm4, dmm5

Vývoj programov na čítanie displeja multimetrov pripojených cez USB

advman

Zobrazí technickú dokumentáciu k zariadeniu Advantech USB 4750

adam

Zobrazí technickú dokumentáciu k zariadeniam ADAM

zwcf

Prehliadanie manuálu Z-Wave funkcií

zwcr

Prehliadanie referenčného manuálu Z-Wave

zwci

Prehliadanie implementačného manuálu Z-Wave

zwck

Prehliadanie kompletného manuálu Z-Wave

zw0nodes

Zo Z-Wave kontrolera načíta existujúce adresy zariadení siete /dev/ttyUSB0

zw1nodes

Zo Z-Wave kontrolera načíta existujúce adresy zariadení siete /dev/ttyUSB1

zw0nodes

Zo Z-Wave kontrolera načíta voľné adresy siete /dev/ttyUSB0

zw1nodes

Zo Z-Wave kontrolera načíta voľné adresy siete /dev/ttyUSB1

cdm

. cdm voľba pracovného adresára pre vývoj Z-Wave technnológie

*Open ZW C++*

cds

. cds voľba pracovného adresára pre vývoj Z-Wave technnológie

*Open ZW C++* zdrojové programy

gzw

Kompilácia Open Z-Wave projektu v C++ a vytvorenie linuxovej knižnice pre Pascal

c+

Editácia programu v C++ a kompilácia

zw

Dokumentácia k *Open ZW*

zwio

Spustenie procesu prijímania dát zo ZW siete, projekt *Open ZW*

zyw, fwl

Dokumentácia k nastaveniu sieťovej ochrany na zariadení ZYWALL

kaz

Zoznam zvukových kníh pre nevidiacich z knižnice v Levoči, kazety

mp3

Zoznam zvukových kníh pre nevidiacich z knižnice v Levoči, CD /

mp3

mun menofoldra

Z mailového foldra vyexportuje do aktuálneho adresára prílohy

poznamky

Poznámky, zápisník

recsynt

Kniha *Naprogramujte si svoj rečový syntetizér s neurónovou sieťou* v textovom súbore



cnv <subor.format1> <format2>

Skonvertuje súbor s formátom 1 na formát 2, napr. súbor.docx na súbor.txt

pdf2txt <súbor>

Konverzia z PDF na text, zalomenie dĺžok riadkov a zobrazenie v editore

doc2txt <súbor>

Konverzia z doc na text, zalomenie dĺžok riadkov a zobrazenie v editore

docx2txt <súbor>

Konverzia z docx na text, zalomenie dĺžok riadkov a zobrazenie v editore

odt2txt <súbor>

Konverzia z odt na text, zalomenie dĺžok riadkov a zobrazenie v editore

unoconvlistener

Skript, ktorý je potrebné spustiť pred prvým použitím príkazu unoconv, cnv a pod.

cpk adresár x Y

Knihy do 100 kapitol na diktafón LS1x z adresára na usx do foldru DSS\_FLDY

cp0 adresár x Y

Knihy do 1000 kapitol na diktafón LS1x z adresára na usx do foldru DSS\_FLDY

wgetaip

Stiahne časopisy AIP

aip

. aip voľba pracovného adresára AIP

findaip reťazec

Hľadá v stiahnutých časopisoch AIP

cpaip meno

Skopíruje číslo AIP meno.txt do aktuálneho adresára



## 6. Programy modulu ROWS a AmI RUDO

Najdôležitejšou súčasťou modulu ROWS a AmI systému RUDO je ich špecifická softvérová výbava. Programy v tomto systéme pracujú výlučne v textovom alebo semigrafickom režime. V prípade semigrafických prostredí ide o používateľské rozhrania, ktoré sú navrhnuté špeciálne na prácu pre nevidiacich ľudí. Softvér modulu ROWS a AmI systému RUDO môžeme rozdeliť do štyroch skupín:

1. bežné programy (viď 6.1),
2. vývojové nástroje (viď 6.2),
3. programy k publikáciám (viď 6.3),
4. obsluha softvérového asistenta pri čítaní (viď ??).

Popis funkcií všetkých programov modulu ROWS a AmI systému RUDO sa nachádza v súbore `~/rows/txt/pomoc.txt`, ktorý je možné prehliadať pomocou skriptu `pomoc` na príkazovom riadku alebo v prostredí HANIBAL voľbou položky Pomoc.

Zostava programov modulu ROWS a AmI systému RUDO definuje tretiu a najdôležitejšiu časť programového rozhrania, pomocou ktorého komunikuje semigrafické používateľské prostredie HANIBAL s operačným systémom. Funkcie položiek v oknách prostredia

HANIBAL sa algoritmicky skladajú z postupnosti štyroch typov príkazov, ktoré sú definované v rovnomennom jazyku:

1. zostava príkazov OS Linux (viď 4.2),
2. zostava skriptov modulu ROWS (viď 5),
3. zostava príkazov modulu ROWS a AmI systému RUDO,
4. zostava interných príkazov prostredia HANIBAL.

Body 1-3 spolu definujú celé programové rozhranie – softvérovú vrstvu, ktorá leží medzi používateľským prostredím HANIBAL a operačným systémom Linux.

Veľmi dôležitou vlastnosťou tohoto programového rozhrania je jeho uniformita pri používaní na príkazovom riadku alebo pri definícii funkcií položiek v oknách prostredia HANIBAL.

Ak si nevidiaci používateľ na príkazovom riadku zvykne na opakované používanie kombinácie postupností z bodov 1-3, môže túto postupnosť algoritmu zapísať do definície v jazyku HANIBAL. V prostredí takto vznikne nová požadovaná funkcia, ktorá sa vykoná jednoduchou voľbou danej položky v okne.

## 6.1 Bežné programy

Nevidiaci používatelia počítačov vybavených AmI systémom RUDO, ktorí sa nezúčastňujú vývoja tohoto AmI systému, sú považovaní za bežných používateľov. Z hľadiska vývoja AmI systému RUDO je bežným používateľom aj nevidiaci administrátor OS Linux.

Softvérová výbava, ktorá je opísaná v tejto podkapitole, je zameraná práve pre takúto skupinu používateľov. Neobsahuje teda vývojové nástroje používané pri výskume a testovaní AmI systému RUDO.

ahojstart

Povie „ahoj“ - použité pri štarte

bib

Zvuková Biblia s konkordanciou

e2s <en-výraz>

Anglicko-slovenský internetový slovník s pamäťou cache, fungujúci aj offline

s2e <sk-výraz>

Slovensko-anglický internetový slovník s pamäťou cache, fungujúci aj offline

etos <súbor1> <súbor2>

Internetový slovník, prekladajúci anglický textový súbor do slovenského textového súboru

stoe <súbor1> <súbor2>

Internetový slovník, prekladajúci slovenský textový súbor do anglického textového súboru

dtos <súbor1> <súbor2>

Internetový slovník, prekladajúci nemecký textový súbor do slovenského textového súboru

stod <súbor1> <súbor2>

Internetový slovník, prekladajúci slovenský textový súbor do nemeckého textového súboru

ed

Textový editor systému ROWS s kontrolou pravopisu, slovníkmi a Bibliou

goboff

Vypnutie syntetizéra GOBLIN

gobon

Zapnutie syntetizéra GOBLIN

gobquit

Odstránenie servera GOBSVR z pamäti

gobsvr

Server syntetizéra GOBLIN

gobtlk

Server na predefinovanie štandardnej klávesnice na klávesnicu pre nevidiacich a server sprostredkovania informácií z displeja

gobtlk -server

Inštalácie pre server bez klávesnice

gobtlk -nobrlv

Braillova klávesnica pri štarte vypnutá

gobtlk -notlk

Syntetizér pri štarte vypnutý

gobtlk -kbdN

Inštalácia na klávesnicu číslo N

grm

Definícia gramatických výnimiek pre syntetizér GOBLIN

lbe

Vykazovanie dochádzky

lcnv

Konverzia textov (sady znakov)

le

Textový editor ako ed, ale bez kontroly pravopisu

lc

Programovateľná kalkulačka

lff

Vyhľadávanie súborov v štruktúre adresárov

lffd

Vymazávanie súborov v štruktúre adresárov

lfft

Vyhľadávanie súborov v štruktúre adresárov so zadáním hľadaného reťazca znakov v danom súbore

sudofont

Informácia o aktívnom fonte displeja, zároveň vygeneruje súbor actualfont.pp, ktorý je súčasťou systému ROWS, implementuje aktuálnu kódovú tabuľku fontu.

lpr

Tlač hypertextových dokumentov napísaných v jazyku pre nevidiacich - WOF, ktorý umožňuje jednoduchú kontrolu grafického formátu

ltd

Čas, dátum a záznamník

pripraveny

Povie pripravený pri zapnutí PC

rows

Kontrola integrity ROWS a generovanie DVD do publikácie

snd

Generovanie orientačných zvukov v skriptoch

spl

Kontrola pravopisu bez editácie

tlkcrline

Povie obsah riadku displeja s kurzorom (pre skripty)

ukonceny

Povie „ukončený“ pri vypnutí PC

volsnd

Nastavovanie hlasitosti, hĺbok a výšok osobitne pre každý orientačný zvuk

wp

Záznamník a prehrávač zvuku s indexáciou zvukových súborov

ziar

Porovnávanie svietivosti rôznych typov žiaroviek

iban

Prevod starého typu účtu na iban, overenie korektnosti iban, výpočet sumarizačného čísla

vn <rok>

V danom roku vypočíta dátum veľkonočnej nedele

taxonometria

Taxonometrický systém a orientačné zvuky

kurenie

Server obsluhujúci kúrenie

unikovakrivka

Technické nastavovanie tvaru únikovej krivky. Výpočet vychádza z parametrov *linfnheating* a *powfnheating* v konfiguračnom súbore *rudo.conf*. Výpočet berie do úvahy externú teplotu  $-10$ , nastavenú internú teplotu  $22$  a únikový koeficient  $1.12$ . Program odovzdá hodnotu únikovej funkcie. Na základe tohoto údaju sa menia parametre *linfnheating* a *powfnheating* v konfiguračnom súbore *rudo.conf*.

zonova0regulacia, zonova1regulacia

Server obsluhujúci zónovú reguláciu prízemí a poschodia

rk

Klient používaný na ovládanie kúrenia

rz0, rz1

Klientské aplikácie používané na ovládanie zónovej regulácie prízemí a poschodia

rudokuric

Klient používaný na ovládanie kúrenia, tento klient je možné inštalovať na bežné PC, používa zvukové zariadenie na vykurovacom servri

rudozoner0, rudozoner1

Klientské aplikácie používané na ovládanie zónovej regulácie prízemí a poschodia, tieto programy je možné inštalovať na bežné PC, používajú zvukové zariadenie na domovom servri

rudoserver

Spustí všetky servre sda Rudo

beep

Systémové pípnutie

sudo console

Povie číslo aktuálnej konzoly

autostart

Volané pri štarte PC, vytvára konzoly

pomoc

Konverzia pomoc.txt na pomoc.hpr

wraptxt <meno.txt>

Zalomí v texte dlhé riadky na dĺžku 64 znakov

wraptxt <meno.txt> <N>

Zalomí v texte dlhé riadky na dĺžku N

setcfg <riad> <reťazec>

V konfiguračnom súbore rudo.conf zmení riadok riad na hodnotu reť

rudocfg <riad>

Vráti hodnotu riadku riad z konfiguračného súboru rudo.conf

rowsdir

Vráti adresár s ROWS v závislosti na používateľovi, používa sa v skriptoch

rowsprefix

Vráti inštalačný prefix - adresár, používa sa v skriptoch

rudopasswd

Uloží zašifrované heslo nevidiaceho používateľa do systému RUDO, ktoré sa potom môže použiť pri automatickom prihlasovaní

## 6.2 Vývojové nástroje

Softvérová výbava uvádzaná v tejto podkapitole je používaná nevidiacim vývojárom AmI systému RUDO. Umožňuje prácu v nasledovných oblastiach:

1. návrh topológií neurónových sietí,
2. adaptáciu sietí pomocou genetického algoritmu,
3. modifikácie neurónových systémov a ich implementáciu do programových modulov,



4. vytváranie hlasových korpusov pre syntetizér,
5. programovanie a ladenie rozsiahlych softvérových balíkov,
6. meranie elektrotechnických veličín,
7. programovanie a práca s bezdrôtovým rozhraním Z-Wave,
8. programovanie a práca s galvanickými oddeľovačmi,
9. ďalšie podporné softvérové aplikácie.

Nižšie sú uvedené konkrétne softvérové aplikácie, ktoré sú využívané nevidiacim vývojárom AmI systému RUDO.

nlga

Vývojové prostredie pre neurónové siete

ga

Program na testovanie genetického algoritmu

ve

Vývojové prostredie syntézy unity selection, syntézy fonémov a vyhlasovacieho systému (súhrnná syntéza GOBLIN)

gobhid

Stav spracovania hlasových databáz

pc

Kompilátor jazyka Pascal s editorom

h2pp

Prevod hlavičkových súborov súborov jazyka C na unity Pascalu

rdk

Odchytávanie kódov klávesnice

brltest

Program používaný pri testovaní hmatového displeja

cpu

Výpis o výkone procesorov PC

myip

Výpis aktuálnej IP adresy a HostName

font

Informácia o aktívnom konzolovom fonte, zároveň vygeneruje súbor actualfont.pp, ktorý je súčasťou systému ROWS.

addtodict

Program používaný pri editácii slovníkovej pamäti cache

dictowords

Program používaný pri editácii slovníkovej pamäti cache

words

Program používaný pri editácii slovníkovej pamäti cache

pp

kompilácia Pascal s priradením behových chýb k číslam riadkov zdrojového textu

testfirmware

Test firmvéru USB

voc

Testovací program používaný pri vývoji analýzy hlasu

pp

Kompilácia s popisom chyby

a4750, advtest

Testy zariadenia Advantech 4750

adv1, adv2, adv3

Testy zariadenia Advantech 4750

zwiodrv zwnet

Testovacie programy siete Z-Wave

wraprtl <oldrtl.txt>

Sformátuje súbor oldrtl.txt a uloží ho do newrtl.txt, formátuje na 80 stĺpcov, ak sa uvedie číselný parameter, bude sa formátovať na uvedený počet stĺpcov.

dmm

Čítanie obsahu displeja na multimetri

### 6.3 Programy k publikácii „Naprogramujte si svoj rečový syntetizér s neurónovou sieťou“

Na Fakulte prírodných vied Univerzity Mateja Bela vzniklo v roku 1997 pracovisko zamerané na vývoj softvérových asistenčných technológií pre nevidiacich používateľov počítačov (viď 1.2).

Od roku 2004 boli v rámci tohoto pracoviska vzdelávaní študenti informatiky. Boli im prezentované informačné technológie, ktoré sa používajú pri návrhu asistenčného softvéru zameraného pre nevidiacich používateľov PC.

Takýmto spôsobom sa rozširovalo poznanie študentov nielen v oblasti všeobecnej informatiky. Zároveň im bol odovzdávaný nový rozmer pohľadu na technológie, ktoré môžu byť využité aj v rámci

pomoci zdravotne hendikepovaným ľuďom. Bolo poukazované nepriamo na sociálnu zložku informatiky, ktorá môže pomáhať nevidiacim ľuďom nielen v segregovaných komunitách, ale aj pri ich integrácii do kolektívov zdravých ľudí.

V rámci tohoto pracoviska boli vydané publikácie [12, 16], ktoré sa používali pri vzdelávaní študentov informatiky vo vyššie spomínanej oblasti.

Prednášajúcim bol nevidiaci vývojár AmI systému RUDO, ktorý do tohoto systému implementoval pre študentov ukážkové aplikácie, pomocou ktorých mohol odprezentovať zvolené témy z informatiky aj napriek svojmu zrakovému hendikepu.

Ďalej uvádzané programy sú zamerané na vzdelávanie študentov informatiky, obsahovo prislúchajú k monografii [16]. V kooperácii s modulom ROWS zároveň ponúkajú asistenčnú službu nevidiacemu prednášajúcemu v procese vyučovania vidiacich študentov.

ahoj

Príklad programu, ktorý povie „ahoj svet“

automat

Príklad konečného softvérového automatu

talkbricks

Príklad automatu akceptujúceho jazyk TALKBRICKS

melodia

Príklad programu, ktorý zahrá melódiu

cli

Príklad softvérového klienta

svr

Príklad softvérového servera

## 6.4 Funkcie asistenčného softvéru GOBTLK

Súčasťou modulu ROWS je syntetizér GOBLIN, ktorý sa nainštaluje pri zapnutí počítača volaním programu *gobsvr*. Tento syntetizér sám nevykonáva žiadnu činnosť, keď sa mu ale cez sieťovú komunikáciu klient/server zašle textový reťazec, GOBLIN tento textový reťazec prečíta syntetickým hlasom cez audiozariadenie počítača.

Aby mohol nevidiaci používateľ čítať obsah displeja, je potrebná ešte jedna služba, ktorá umožňuje výber informácií z plochy displeja a ich následné zaslanie syntetizérovi GOBLIN. V module ROWS túto

službu sprostredkúva čítač obrazovky, ktorý sa nainštaluje volaním programu *gobtlk*.

Program *gobtlk* navyše predefinuje bežnú klávesnicu na Braillov mód. Táto funkcia umožňuje písanie Braillovho – bodového písma. *Gobtlk* ponúka tri dôležité funkcie:

1. automatický alebo interaktívny výber textu z plochy obrazovky,
2. možnosť písania v Braillovom písme,
3. spätná hlasová kontrola stláčaných klávesov.

Pri nainštalovaní služieb *gobtlk* je automaticky zapnutý Braillov mód písania a tiež aj čítanie obsahu obrazovky. Nižšie uvedené parametre môžu tieto nastavenia zrušiť.

Parameter *-nobrl* vypne Braillovu klávesnicu

Parameter *-notlk* vypne syntetizér GOBLIN

Ak je služba čítania obsahu obrazovky zapnutá, *gobtlk* umožňuje interaktívnu obsluhu pomocou viacerých kombinácií horúcich klávesov. Zoznam klávesových kombinácií používaných pri čítaní displeja je uvedený nižšie.

Ctrl + Shift + F12

Ukončí program GOBTLK

Alt + AltR

Prepínanie písania semigrafiky

Ctrl + AltR/CtrlR

Vyp/Zap Braillova klávesnica „ASDF“, „JKL+Space“

Shift + AltR/CtrlR

Vyp/Zap syntetizér GOBLIN pre textový režim obrazovky

Shift + Alt + CtrlR

Vypíše túto pomoc na konzolu, z ktorej bol program spustený

Ctrl + Alt + CtrlR

Prepínač typu hláskovania

Ctrl + Shift + Alt + CtrlR

Prepína syntézu fonémov a nehomogénnu syntézu

Ctrl + Shift + Down

Pozíciu čítaného riadku presunie na riadok s kurzorom prečíta ho alebo stopne čítanie

Ctrl + Shift + Up

Prečíta čítaný riadok alebo stopne čítanie

**Ctrl** + **Up/Down**

Presunie pozíciu vyššie/nížšie a prečíta riadok

**Shift** + **Up/Down**

Presunie vyššie/nížšie a prečíta riadok, prázdne preskakuje

**Ctrl** + **Shift** + **Right**

Pozíciu čítaného slova presunie na slovo s kurzorom prečíta ho alebo stopne čítanie

**Ctrl** + **Shift** + **Left**

Prečíta čítané slovo alebo stopne čítanie

**Ctrl** + **Left/Right**

Presunie pozíciu naspäť/ďalej a prečíta slovo

**Ctrl** + **Shift** + **Alt** + **Right**

Pozíciu čítanej hlásky presunie na kurzor prečíta hlásku alebo stopne čítanie

**Ctrl** + **Shift** + **Alt** + **Left**

Prečíta čítanú hlásku alebo stopne čítanie

**Ctrl** + **Alt** + **Left/Right**

Presunie pozíciu naspäť/ďalej a prečíta hlásku

**Ctrl** + **Alt** + **Up/Down**

HOME/END pre pozíciu čítanej hlásky

**Ctrl** + **Shift** + **Alt** + **Up/Down**

HOME/END pre pozíciu čítaného slova

**Shift** + **Left/Right**

Hlasitosť čítania

**Shift** + **Alt** + **Up/Down**

Rýchlosť čítania

**Shift** + **Alt** + **Left/Right**

Výška hlasu

**Alt** + **CtrlR**

Povie číslo aktívnej konzoly



## 7. Spracovanie textu

Asistenčný softvér pre nevidiacich používateľov počítačov, ktorý je opisovaný v tejto kapitole, používa syntetizér GOBLIN piatimi spôsobmi:

1. čítanie obsahu displeja pomocou aplikácie GOBTLK,
2. nastavovanie parametrov čítania pomocou servera GOBTLK (program GOBTLK sa pri štarte v pamäti RAM rozdelí na aplikáciu čítania a server poskytujúci služby),
3. priame volanie zvukových funkcií servera GOBSVR,
4. priame volanie služby syntézy servera GOBSVR (syntéza s generátorom koartikulačných prechodov),
5. priame volanie služby syntézy servera GOBSVR (syntéza „object selection“ - vývoj novej technológie).

Programové aplikácie ROWS zväčša kombinujú prvé štyri uvedené služby s ďalšou možnosťou použitia hmatového displeja. V nasledujúcej podkapitole je opísaná práca so špeciálnym editorom pre nevidiacich, ktorý sa používa nielen pri písaní bežných textov, ale aj pri programovaní alebo používaní zápisníka. Jednotný prístup pri spracovaní textov výrazne urýchlí prácu nevidiaceho človeka na počítači.

## 7.1 Práca so špeciálnym editorom

Editor s asistenčnou technológiou pre nevidiacich sa používa v ROWS pri spracovaní akýchkoľvek textov:

1. dokumenty s hypertextovými príkazmi jazyka WOFF (viď 7.2),
2. texty bez príkazov jazyka WOFF,
3. zdrojové texty programov a definičných súborov.

Editor obsahuje slovenskú kontrolu pravopisu a interaktívny prekladový slovník anglicko-slovenský a slovensko-anglický. Program *le* je aplikáciou editora bez kontroly gramatiky, program *ed* aplikuje editor so zložkou kontroly gramatiky. Pri spracovaní textových dokumentov v slovenskom jazyku sa veľkosti a typy písma zapisujú pomocou hypertextových príkazov jazyka WOFF, ktoré začínajú znakom „~“ (viď 7.2). Hypertextové príkazy umožňujú také formátovanie zdrojového textu, ktoré je ľahko čitateľné nevidiacim človekom. Avšak po kompilácii a vytlačení na bežnej tlačiarňi bude mať tento text charakter dokumentu programu *Word* alebo *Writer*.

Esc	Ukončenie práce
F10	Ukončenie bez uloženia zmien
F1	Pomocná obrazovka
F4	Dĺžka riadka pri zalamovaní
F2	Uloženie zmien
F3	Uloženie pod iným menom
F9	Zrušenie nastavených blokov
Ctrl + PgUp	Začiatok textu
Ctrl + PgDown	Koniec textu
Ctrl + Enter	Preklad slova s kurzorom angličtina → slovenčina
Shift + Enter	Preklad slova s kurzorom slovenčina → angličtina
Ctrl + Shift + Enter	Vkladanie/mazanie anglického slova s kurzorom do/zo slovníka
Ctrl + K Y	Vymazanie bloku
Ctrl + K B	Označenie začiatku bloku
Ctrl + K K	Označenie konca bloku
Ctrl + K C	Kopírovanie bloku
Ctrl + K V	Presun bloku
Ctrl + K F	Formátovanie bloku
Ctrl + K R	Načítanie bloku zo súboru
Ctrl + K W	Zápis bloku do súboru



Ctrl + Q	F	Nájdienie reťazca dopredu
Ctrl + Q	B	Nájdienie reťazca dozadu
Ctrl + R	F	Prepísanie reťazcov dopredu
Ctrl + R	B	Prepísanie reťazcov dozadu
Ctrl + L		Opakuj reťazcovú operáciu
Ctrl + B		Preveď reťazec na blok

### 7.1 Register editačných príkazov

-w	Parameter vypne zalamovanie
-w64	Zapne zalamovanie na 64 znakov
-v	Parameter zapne prehliadací mód, nedovolí robiť zmeny
-13 -9	nastaví kurzor na súradnice na slúpec 13 a riadok 9

### 7.2 Parametre príkazového riadku programov *le* a *ed*

## 7.2 Práca s hypertextom a tlač dokumentov

Program pre tlač dokumentov sa v systéme ROWS volá *lpr*. Zdrojový dokument s textom musí obsahovať hypertextové príkazy v jazyku WOLF. Tento jazyk umožňuje nevidiacemu používateľovi rýchlu prípravu dokumentu. Kompilátor jazyka WOLF vykonáva kontrolu integrity hypertextových príkazov. Ak príkaz nedáva zmysel alebo by pri jeho vykonaní došlo k formátovacej chybe, text sa nevytlačí. LPR vygeneruje súradnice a opis chyby, čím do určitej miery nevidiacemu človeku kontroluje pripravovaný formát dokumentu.

### 7.2.1 Volanie programu *lpr*

Funkcie *lpr* sú voliteľné pomocou hypertextových príkazov v jazyku WOLF. Niektoré z nich sa však dajú voliť priamo pri volaní programu pomocou jeho parametrov. *lpr* s parametrami *-h* alebo *-?* zobrazí na displeji jednoduchý rozpis všetkých parametrov a hypertextových príkazov WOLF. Rozpis sa prehliada pomocou klávesov **PgDown** a **PgUp**, ukončí sa stlačením **Esc**. Ďalej sú uvedené všetky parametre *lpr*. Prvý parameter je buď *-?*, alebo *-H*, alebo názov súboru na vytlačenie. Druhý parameter je:

- názov súboru s hlavičkou dokumentu,
- pomlčka, ak sa má tlačíť len päta dokumentu bez hlavičky,

- niektorý z príkazov, ktoré začínajú pomlčkou.

Tretí parameter je:

- názov súboru s päťou dokumentu,
- parameter začínajúci pomlčkou.

-S	Súbory sa budú číslovať spolu (vytváranie viazaní)
-n	n je číslo, od ktorého sa budú číslovať strany
-N	Najskôr tlač nepárnych strán, potom párnych
-P	Duplexná tlač s výmenou pozície čísel strán
-T	Neformátovať text
-A	Formátovať program
-X	Syntaktická kontrola príkazov v texte
-Kn	n je počet kópií
-H	Tlačiť hlavičku na každú stranu
-F	Tlačiť poznámku pod textom na každú stranu
-D-  +	Dátum vypnúť   zapnúť tlač na každú stranu
-M-  +	Čas vypnúť   zapnúť tlač na každú stranu
-R-	Číslovanie strán vypnúť
-On	n je ľavý okraj
-Ln	n je dĺžka riadka
-Gn	n je počet riadkov na stranu
-En	n je počet medzier posunutia v odseku
-I	Číslovanie riadkov zapnúť, len pri tlači programu
-Vn	n je počet riadkov hlavičky, štandard je n = 5
-Zn	n je počet riadkov poznámky pod textom, štandard je n = 3
-Bn	n je počet vynechaných riadkov medzi riadkami, maximálne n = 6
-B8	Zapnutie 8-bodového Braila (len pri bodovej tlačiarni)
-[n	Vytlačí text od n-tej strany
-]m	vytlačí text po m-tú stranu
-DE	Odstráni príkazy z textu a výsledok uloží do súboru.dhp
-WO	Vytlačí len obsah dokumentu
-+o	Vytlačí dokument aj s obsahom
--x.y	Výstup uloží do súboru x.y
-RTF	Generovanie súboru vo formáte Rich Text Format.

### 7.2.2 Tlač čistého textu a programov

*lpr* umožňuje tlač čistých textov, ktoré nie sú uložené v grafických formátoch a neobsahujú hypertextové príkazy WOF. Príkaz na tlač čistého textu je `lpr text.txt -t` Parameter `-t` vykonáva

zalamovanie dlhých riadkov textu a rozpoznáva v texte odseky, začínajúce tromi medzerami. Ak sa zadá len meno súboru bez cesty, *lpr* hľadá súbor najskôr v aktuálnom adresári, ak ho nenájde, hľadá ho v adresári súborov pre tlač, ktorý je definovaný v *rudo.conf*. Vyššie uvedený príkaz pridá k textu na začiatok dátum a čas tlače, pričom každú stránku očísľuje. Parametre *-d-* *-m-* *-r-* vypnú pridávanie dátumu, času a číslovania strán. Program *lpr* umožňuje tlač zdrojových textov programov. Ak riadok v programe presiahne dĺžku tlačového riadka, *lpr* ho rozdelí a druhú časť riadka položí na pozíciu programového odseku. Nasledujúci príkaz vytlačí program `lpr program.pp -a` Pri tlači píše program *lpr* na displej aktuálny stav tlačovej úlohy.

### 7.2.3 Tlač dokumentov, syntax jazyka WOFF

Dokumenty s hypertextovými príkazmi WOFF majú príponu *.hpr*. Pri tlači dokumentu sa prípona nežadáva. Nižšie uvedený príkaz vytlačí dokument `subor.hpr` `lpr subor` Ak sa v príkaze uvedie parameter *-x*, príkaz len skontroluje syntax dokumentu. Keď je potrebné preniesť dokument do formy čistého textu, použije sa parameter *-de*, program vytvorí `subor.dhp`. Parameter *-rtf* exportuje dokument do programov Word a Writer. Ďalej je uvedená syntax základných príkazov jazyka WOFF.

~~NR~NONR	Nový riadok
~NO	Nový odsek
~HS	Hlavica späť
~NS	Nová strana
~DA	Napíš dátum
~CA	Napíš čas
~PP+/-	Podčiarkovanie zap-vyp
~PD+/-	Tmavá tlač zap-vyp
~PZ+/-	Výrazná tlač zap-vyp
~PK+/-	Kurzíva zap-vyp
~PS+/-	Široká tlač zap-vyp
~PV+/-	Vysoká tlač zap-vyp
~PU+/-	Úzka tlač zap-vyp
~PM+/-	Mini tlač zap-vyp
~PE+/-	Eko tlač zap-vyp
~HT	Horizontálny tab.

~VT	Vertikálny tab.
~C<	Začiatok centrovania
~C>	Koniec centrovania
~VMn	Vynechaj n medzier
~VRn	Vynechaj n riadkov
~Zan	Napíš n krát znak <i>a</i>
~TD	Draft
~TA	Sans serif
~TC	Courier
~TP	Prestige
~TR	Roman
~TS	Script
~TK	Script C
~TB	OCR B
~TO	Orator
~TT	Orator S
~NHn	Nastav šírku horizontálneho tabulátora na n znakov
~NVn	Nastav výšku vertikálneho tabulátora na n znakov
~SGn	n krát signál
~DT?, ~ST?, ~CT?	Typ písma pre dátum, stránkovanie, čas
~RI n	Riadkovanie po n vynechaných riadkov
~DDL n	n medzier na okraj hlavičiek
~DDR n	n znakov na riadok hlavičiek
~OZ+   -	Zapne   vypne zarovnávanie pravého okraja
~RZ	Zarovnanie riadku pred príkazmi konca riadku
~DL n	n medzier na okraj
~DR n	n znakov na riadok
~DS n	n riadkov na stranu
~DH n	n riadkov na hlavičku
~DY n	n riadkov na päť
~DO n	n medzier v odseku
~RD n	Číslovanie od n-tej strany
~DV-Z	Dátum vypnúť - zapnúť na každú stranu
~SV-Z	Číslovanie strán vypnúť - zapnúť

~CV-Z	Čas vypnúť - zapnúť na každú stranu
~R(n ~R)m	Vytlačia sa strany od n po m
~OB	Vytlačí sa k dokumentu aj obsah
~PT	Písmo s tieňom
~PO	Obrys písmen
~OT	Obrys písma s tieňom
~PN	Písmo normál
~~	Znak ~
~'	Nedeliteľná medzera
~+ ~:	Horný index zapnúť   vypnúť
~- ~:	Dolný index zapnúť   vypnúť
~Rn< ~R>	Kde n=1...6, písanie nadpisu Roman
~An< ~A>	Kde n=1...6, Písanie nadpisu Sans Serif
~PR+ -	Zapne   Vypne proporcionálne písmo
~HPR- +	Vypne   zapne akceptovanie príkazov WORD
~PU+	Na bodových tlačiarňach zapne 8-bodový brail
~DP?+/-, ~SP?+/-, ~CP?+/-	Modifikácia písma pre dátum, stránkovanie a čas; pre ? sa neakceptujú znaky S,E,U,V,M

### 7.2.4 Makrá v dokumentoch

Makro je definované postupnosťou základných príkazov jazyka WORD v súbore macros.hpd. V dokumente sa makro volá identicky ako základný hypertextový príkaz.

Formát dokumentu A4 nastavuje napríklad makro ~a4ini. V nasledujúcom zozname sú uvedené všetky štandardne dodávané makrá.

Definície nadpisov, ktoré obsahujú reťazce |a?| resp. |b?| sú začiatkom resp. koncom nadpisu s indexom ?, ktorý bude zaradený do automatického generovania obsahu.

```
A4INI{DV,CV,SV,TR,STC,DTC,CTC,OZ+}
A5INI{DV,CV,SV,TR,STC,DTC,CTC,DL2,DR42,DS40,OZ+}
BR6INI{Dv,Cv,Sv,DL3,DR36,DS28,NH4,NV2,DDL1,DDR40,DH3,DY2}
BR8INI{Dv,Cv,Sv,PU+,DL3,DR36,DS23,NH4,NV2,DDL1,DDR40,DH3,DY2}
A4IP{DV,CV,SV,TR,STC,DTC,CTC,PR+,OZ+}
A5IP{DV,CV,SV,TR,STC,DTC,CTC,DL2,DR42,DS40,PR+,OZ+}
```

T+{PZ+,PD+}  
 T-{PZ-,PD-}  
 K+{PK+,PZ+,PD+}  
 K-{PD-,PZ-,PK-}  
 RO{NR,NO}  
 TXT{VR2,NO}  
 K1<{|a1|TK,PZ+,PD+,PV+,PT,SG2,C<}  
 K>{|b1|C>,PN,PV-,TR,PK-,PD-,PZ-}  
 O1<{|a0|VR3,TB,PK+,PZ+,PD+,SG1,C<}  
 O>{|b0|C>,TR,PZ-,PD-,PK-,PN,NR,NO}  
 OV>{|b0|C>,TR,PZ-,PD-,PK-,PN,NR,NR}  
 V>{|b2|C>,TR,PZ-,PD-,PK-,PN,NO}  
 K2<{|a1|TA,PZ+,PD+,PV+,OT,SG2,C<}  
 O2<{|a0|VR3,Ta,P0,PZ+,PD+,SG1,C<}  
 K3<{|a1|TS,PZ+,PD+,PV+,P0,PK+,SG2,C<}  
 O3<{|a0|VR3,TS,PT,PK+,PZ+,PD+,SG1,C<}  
 K4<{|a1|TP,PZ+,PD+,PV+,Pt,SG2,C<}  
 O4<{|a0|VR3,TC,PT,PZ+,PD+,SG1,C<}  
 K5<{|a1|TC,PZ+,PD+,PV+,Pt,SG2,C<}  
 O5<{|a0|VR3,TP,PT,PZ+,PD+,SG1,C<}  
 K6<{|a1|T0,PZ+,PD+,PV+,Pt,SG2,C<}  
 O6<{|a0|VR3,T0,PT,PZ+,PD+,SG1,C<}  
 K7<{|a1|T0,PZ+,PD+,PV+,SG2,C<}  
 O7<{|a0|VR3,T0,PZ+,PD+,SG1,C<}  
 K8<{|a1|TS,PZ+,PD+,PV+,SG2,C<}  
 O8<{|a0|VR3,TS,PZ+,PD+,SG1,C<}  
 K9<{|a1|TR,PZ+,PD+,PV+,SG2,C<}  
 O9<{|a0|VR3,TR,PZ+,PD+,SG1,C<}  
 K0<{|a1|TC,PZ+,PD+,PV+,SG2,C<}  
 O0<{|a0|VR3,TC,PZ+,PD+,SG1,C<}  
 V1<{|a2|TB,PK+,PZ+,PD+,SG1,C<}  
 V2<{|a2|Ta,P0,PZ+,PD+,SG1,C<}  
 V3<{|a2|TS,PT,PK+,PZ+,PD+,SG1,C<}  
 V4<{|a2|TC,PT,PZ+,PD+,SG1,C<}  
 V5<{|a2|TP,PT,PZ+,PD+,SG1,C<}  
 V6<{|a2|T0,PT,PZ+,PD+,SG1,C<}  
 V7<{|a2|T0,PZ+,PD+,SG1,C<}  
 V8<{|a2|TS,PZ+,PD+,SG1,C<}  
 V9<{|a2|TR,PZ+,PD+,SG1,C<}

```

V0<{ |a2|TC,PZ+,PD+,SG1,C<}
W>{ |b3|C>,TR,PZ-,PD-,PK-,PN,N0}
W1<{ |a3|VR2,TB,PK+,PZ+,PD+,SG1,C<}
W2<{ |a3|VR2,Ta,P0,PZ+,PD+,SG1,C<}
W3<{ |a3|VR2,TS,PT,PK+,PZ+,PD+,SG1,C<}
W4<{ |a3|VR2,TC,PT,PZ+,PD+,SG1,C<}
W5<{ |a3|VR2,TP,PT,PZ+,PD+,SG1,C<}
W6<{ |a3|VR2,T0,PT,PZ+,PD+,SG1,C<}
W7<{ |a3|VR2,T0,PZ+,PD+,SG1,C<}
W8<{ |a3|VR2,TS,PZ+,PD+,SG1,C<}
W9<{ |a3|VR2,TR,PZ+,PD+,SG1,C<}
W0<{ |a3|VR2,TC,PZ+,PD+,SG1,C<}
RA<{ |a4|PZ+,PD+,OT,R2<,C<}
AA<{ |a5|PZ+,PD+,OT,A2<,C<}
RB<{ |a4|PZ+,PD+,OT,R4<,C<}
AB<{ |a5|PZ+,PD+,OT,A4<,C<}
RC<{ |a4|PZ+,PD+,OT,R6<,C<}
AC<{ |a5|PZ+,PD+,OT,A6<,C<}
RR>{ |b4|PZ-,PD-,PN,C>,R>}
AA>{ |b5|PZ-,PD-,PN,C>,A>}

```

### 7.2.5 Predefinovanie konvencií *lpr*

Všetky parametre, príkazy, orientačné melódie a výpisy sú v programe LPR definované pomocou textových definičných súborov:

1. parametre: `params.hpd`,
2. príkazy: `commands.hpd`,
3. melódie: `sounds.hps`,
4. výpisy: `reports.hpd`.

V prvom a druhom prípade stačí nahradiť reťazec na príslušnom riadku iným reťazcom a program *lpr* začne akceptovať hypertextový príkaz pod iným označením.

Melódie sú definované v jazyku FAUN. Zmenou definície sa zmení orientačná melódia stavu tlače.

Znenie informačných výpisov je uložené v súbore `reports.hpd`. Každý riadok súboru je jedným z výpisov. Zmenou textu na danom riadku sa dosiahne zmena príslušného výpisu.

### 7.2.6 Vytváranie fontov pre tlačiareň

*lpr* ovláda tlačiarne pomocou riadiacich sekvencií. Tieto sekvencie sú definované v súboroch.hpf, ktoré budeme nazývať fonty. V konfiguračnom súbore je názov fontu definovaný hneď na prvom mieste.

Ako príklad je ukázaný font *djhp.hpf*, ktorý definuje riadiace sekvencie pre skupinu tlačiarň, ktoré akceptujú príkazy jazyka PCL. Na začiatku každej riadiacej sekvencie je číslo určujúce jej dĺžku. Za ním nasledujú znaky alebo čísla, ktoré sú kódom riadiacej sekvencie.

```

45 27 'E' 27 '&' '1' '2' '6' 'A' 27 '(' '1' '7' 'U' 27 '(' 's'
    '0' 'P' 27 '(' 's' '1' '0' 'H' 27 '(' 's' '1' '2' 'V' 27 '('
    's' '0' 'S' 27 '(' 's' '0' 'B' 27 '&' '1' '0' 'S'
    {štartovací inicializačný reťazec}
0 {ukončovací inicializačný reťazec}
5 27 '&' 'd' '3' 'D' {podčiarkovanie zapnúť}
4 27 '&' 'd' '@' {podčiarkovanie vypnúť}
5 27 '(' 's' '7' 'B' {dvojnásobne pretlačené zapnúť}
5 27 '(' 's' '0' 'B' {dvojnásobne pretlačené vypnúť}
5 27 '(' 's' '3' 'B' {zvýraznené zapnúť}
5 27 '(' 's' '0' 'B' {zvýraznené vypnúť}
5 27 '(' 's' '1' 'S' {kurzíva zapnúť}
5 27 '(' 's' '0' 'S' {kurzíva vypnúť}
0 {široké zapnúť}
0 {široké vypnúť}
12 27 '(' 's' '1' '8' 'H' 27 '(' 's' '8' '5' 'T'
    {kondenzované zapnúť}
11 27 '(' 's' '1' '0' 'H' 27 '(' 's' '3' 'T' {kondenzované vyp}
11 27 '(' 's' '1' '5' 'H' 27 '(' 's' '2' 'T'
    {ekonomické zapnúť}
11 27 '(' 's' '1' '0' 'H' 27 '(' 's' '3' 'T'
    {ekonomické vypnúť}
10 27 '(' 's' '5' 'H' 27 '&' '1' '3' 'D' {vysoké zapnúť}
11 27 '(' 's' '1' '0' 'H' 27 '&' '1' '6' 'D' {vysoké vypnúť}
12 27 '(' 's' '1' '8' 'H' 27 '&' '1' '1' '3' 'D'
    {miniaturne zapnúť}
11 27 '(' 's' '1' '0' 'H' 27 '&' '1' '6' 'D'
    {miniaturne vypnúť}
10 27 '(' 's' '1' 'Q' 27 '(' 's' '3' 'T' {draft}

```



```

10 27 '(' 's' '2' 'Q' 27 '(' 's' '3' 'T' {sans sherif}
10 27 '(' 's' '2' 'Q' 27 '(' 's' '3' 'T' {roman}
10 27 '(' 's' '2' 'Q' 27 '(' 's' '3' 'T' {courier}
10 27 '(' 's' '2' 'Q' 27 '(' 's' '3' 'T' {prestige}
10 27 '(' 's' '2' 'Q' 27 '(' 's' '3' 'T' {skript}
0 {tieň}
0 {obrys}
0 {obrystieň}
0 {normal}
0 {horný index}
0 {dolný index}
0 {koniec indexov}
11 27 '(' 's' '1' '0' 'H' 27 '&' '1' '6' 'D'
    {ukonči proporcionálny nadpis roman}
11 27 '(' 's' '1' '0' 'H' 27 '&' '1' '6' 'D'
    {ukonči proporcionálny nadpis sans sherif}
0 {pípnutie}
1 13 {hlava späť}
2 13 10 {nový riadok}
2 12 13 {nová strana}
0 {koniec súboru}
20 27 '(' 's' '2' 'Q' 27 '(' 's' '8' 'H' 27 '(' 's' '3' 'T' 27
    '&' '1' '5' 'D'
    {roman 1}
20 27 '(' 's' '2' 'Q' 27 '(' 's' '8' 'H' 27 '(' 's' '3' 'T' 27
    '&' '1' '5' 'D'
    {sans sherif 1}
20 27 '(' 's' '2' 'Q' 27 '(' 's' '7' 'H' 27 '(' 's' '3' 'T' 27
    '&' '1' '5' 'D'
    {roman 2}
20 27 '(' 's' '2' 'Q' 27 '(' 's' '7' 'H' 27 '(' 's' '3' 'T' 27
    '&' '1' '5' 'D'
    {sans sherif 2}
20 27 '(' 's' '2' 'Q' 27 '(' 's' '6' 'H' 27 '(' 's' '3' 'T' 27
    '&' '1' '4' 'D'
    {roman 3}
20 27 '(' 's' '2' 'Q' 27 '(' 's' '6' 'H' 27 '(' 's' '3' 'T' 27
    '&' '1' '4' 'D'
    {sans sherif 3}

```

```

20 27 '(' 's' '2' 'Q' 27 '(' 's' '5' 'H' 27 '(' 's' '3' 'T' 27
    '&' '1' '3' 'D'
    {roman 4}
20 27 '(' 's' '2' 'Q' 27 '(' 's' '5' 'H' 27 '(' 's' '3' 'T' 27
    '&' '1' '3' 'D'
    {sans sherif 4}
20 27 '(' 's' '2' 'Q' 27 '(' 's' '4' 'H' 27 '(' 's' '3' 'T' 27
    '&' '1' '2' 'D'
    {roman 5}
20 27 '(' 's' '2' 'Q' 27 '(' 's' '4' 'H' 27 '(' 's' '3' 'T' 27
    '&' '1' '2' 'D'
    {sans sherif 5}
20 27 '(' 's' '2' 'Q' 27 '(' 's' '3' 'H' 27 '(' 's' '3' 'T' 27
    '&' '1' '2' 'D'
    {roman 6}
20 27 '(' 's' '2' 'Q' 27 '(' 's' '3' 'H' 27 '(' 's' '3' 'T' 27
    '&' '1' '2' 'D'
    {sans sherif 6}
10 27 '(' 's' '2' 'Q' 27 '(' 's' '3' 'T' {script C}
10 27 '(' 's' '2' 'Q' 27 '(' 's' '3' 'T' {orator}
10 27 '(' 's' '2' 'Q' 27 '(' 's' '3' 'T' {orator S}
10 27 '(' 's' '2' 'Q' 27 '(' 's' '3' 'T' {OCR B}
0 {proporcionálne}
0 {proporcionálne vypnúť}
0 {znak 0}
0 {znak 255}
35 2 1 {široké 2/1 normálu}
    5 4 {šírka 1 5/4 normálu}
    10 7 {šírka 2 10/7 normálu}
    5 3 {šírka 3 5/3 normálu}
    2 1 {šírka 4 2/1 normálu}
    5 2 {šírka 5 5/2 normálu}
    10 3 {šírka 6 10/3 normálu}
    5 9 {kondenzované 5/9 normálu}
    2 3 {ekonomické 1/1 normálu}
    3 7 {vysoké 3/7 riadkov normálu}
    5 7 {výška 1 5/7 riadkov normálu}
    5 7 {výška 2 5/7 riadkov normálu}
    4 7 {výška 3 4/7 riadkov normálu}

```

---

```
3 7 {výška 4 3/7 riadkov normálu}
2 7 {výška 5 3/7 riadkov normálu}
2 7 {výška 6 2/7 riadkov normálu}
13 7 {miniatúrne 13/7 riadkov normálu}
    58 {maximálny počet riadkov tlačiarne}
0 {transparent mod pre Braillove tlačiarne}
0 {definícia Braillových znakov 0-127 byte na body}
0 {definícia Braillových znakov 128-255 byte na body}
0 {definícia Braillových prefixov a Sl. interpunkcie}
5 27 '&' 'l' '1' 'S' {Duplexná tlač}
```



## 8. Dáta, čísla a zvuk

V tejto kapitole sú opísané ešte tri aplikácie, ktoré spadajú medzi bežné používateľské nástroje. Ide o diár s kalendárom a hodinami, o programovateľný kalkulátor a o prostredie na prácu so zvukom.

Programovateľný kalkulátor je navrhnutý tak, aby svojimi funkciami pokrýval potreby štúdia na základných a stredných školách.

Prostredie na prácu so zvukom sa používa na bežné prehrávanie a nahrávanie zvukových záznamov. Bolo však použité aj pri vývoji hlasového korpusu pre syntetizér GOBLIN.

### 8.1 Kalendár a zápisník

ROWS obsahuje ešte niekoľko podporných kompenzačných aplikácií, z ktorých na záver krátko opíšeme ešte jednu. Program *ltd* je zápisník, kalendár a hodiny v jednom. Obsahuje mená:

1. bežného slovenského kalendára,
2. bežného českého kalendára,
3. evanjelického kalendára,
4. katolíckeho kalendára.

Navyše obsahuje rozdelenie roka na znamenia a indikáciu priestupných rokov.

Obsahuje poznámkový pripomienkový systém, v ktorom si môže používateľ definovať pripomienky podľa stanoveného dátumu. V nasledujúcom príklade je ukázaná syntax zápisu pripomienok.

### 8.1.1 Syntax pripomienkového systému

Mená a dátumy narodení osôb v uvedenom príklade sú fiktívne. Hviezdička a číslo označuje mesiac. Za hviezdikou sa po vynechaní riadka zapíše hneď na začiatok riadka deň v mesiaci, bodka a rok narodenia. Ďalej sa napíše meno človeka, ktorého narodenie sa má pripomínať.

Ak sa namiesto roku narodenia napíše rok nula, program očakáva krátku poznámku na jeden riadok, ktorá sa bude pripomínať daný deň v mesiaci.

Ak sa namiesto roku nula uvedie aktuálny rok, program raz upozorní na danú pripomienku a potom celý riadok vymaže.

Ak je v tomto súbore syntaktická chyba, program vygeneruje hlasný gong a vráti sa späť do editora na riadok s chybou.

\*1

28.1960 Ing. Jana Bohúňová

\*2

12.2004 spoznania našich milých Kolesárovcov

16.1989 Bohuš Jablonka

\*3

9.1985 Eva Doležalová

30.1940 Katarína Gajdošová

31.1937 PhDr. Juraj Snopko

\*4

2.1970 Ing. Lucia Kollárová

7.1937 Mgr. Ján Chalúpka

\*5

2.1936 Magdaléna Kováčová  
19.1962 PhDr. Blanka Hudecová  
26.1941 Soňa Gábrišová  
31.1966 Štefan Bielik

\*6

12.1968 Ing. Ľuboš Prištiak  
15.1967 Ing. Emil Sivák  
22.1970 Mgr. Daniel Filadelfi  
23.1995 Michal Dobrota

\*7

19.1947 PhDr. Anna Maková, CSc.

\*8

\*9

\*10

\*11

12.1994 spoločného manželstva Mihálikovcov  
14.1969 Mgr. Vladimír Seeman  
28.1973 Mgr. Jana Černáková

\*12

20.1934 Ing. Marián Bula  
30.1969 Nina Kurčíková










## 8.2 Programovateľný kalkulátor

Kalkulátor sa rozbehne pomocou príkazu *lc*, alebo programovým volaním modulu *lcalc.pp*.

Prostredie kalkulátora obsahuje niekoľko okien. Okno vo vrchnej časti displeja ukazuje 25 položiek zásobníkovej pamäte. Do zásobníka sa vkladá číslo tak, že sa za ním napíše operátor push. Ak sa namiesto čísla zapíše operátor POP, vyberie sa číslo z vrcholu zásobníka. V strednej časti displeja sa nachádza okno, ktoré sa používa ako displej kalkulátora. Pod ním sa nachádza:

- okno s hodnotami pamäti 0-15,
- okno aktuálneho stavu,
- okno programových výpisov a opisu funkcií.

Výber čísla z pamäte N sa vykoná, keď sa na displej zapíše číslo N nasledované operátorom m/o. Číslo z displeja sa vloží do pamäte, keď za ním nasleduje operátor m/i, za ktorým musí nasledovať číslo pamäte. Ďalšie bežné ovládacie klávesy sú

	vykoná príkaz alebo výpočet.
	návrat do predchádzajúceho stavu, rušenie čísel a funkcií na displeji.
	editácia čísel a funkcií.
	vloženie čísla alebo funkcie. Špeciálne príkazy, ktoré sa zapisujú namiesto čísel: POP, PI, E, zobrazí číselne.
	ukončenie kalkulátora.
	pomocné okno.
	prehliadanie v pomocnom okne.
	ukončenie behu programu.
	vytváranie programov.

Kalkulátor sa používa prirodzeným zadávaním čísel a funkcií, ako napr:

```

3 + 4 =
5 * 8 =
9 : 3 =
5 * ( 3 - ( 8 + 2 ) ) =
30 cos
100 log

```

### 8.2.1 Binárne operácie

Prvý argument je a1, druhý argument je a2.



Číslo pred operáciou je jej priorita. Operácia s najväčšou prioritou sa vykoná najskôr. Pre zmenu priorít sa používajú zátvorky (). Maximálny počet otvorených zátvoriek je štyri. Ak je meno funkcie nasledované hviezdíčkom, znamená to, že daná funkcia je implementovaná ako externý program.

Zvláštne čísla:

PI                    Ludolfovo číslo, 3, 141592654...

E                    základ prirodzeného logaritmu,

POP                číslo z vrcholu zásobníka,

READ              načíta číslo do bežiacého programu,

RUN>????????? kde ???????? sa nahrádza menom programu, uloženým v súbore ???????? .big. Namiesto zápisu RUN>????????? ostane na displeji číslo alebo časť výpočtu, ktorý je výstupom daného programu.

Priorita	Meno	Syntax	Opis
3	+	a1 + a2	súčet
3	-	a1 - a2	rozdiel
4	*	a1 * a2	súčin
4	:	a1 : a2	podiel
4	/	a1 / a2	podiel
2	div	a1 div a2	celočíselné delenie, operácia berie celé časti argumentov
2	mod	a1 mod a2	zvyšok po celočíselnom delení, operácia berie celé časti argumentov
1	or	a1 or a2	logické bitové sčítanie, operácia berie celé časti argumentov
1	xor	a1 xor a2	logické bitové disjunktné sčítanie, operácia berie celé časti argumentov
2	and	a1 and a2	logické bitové násobenie, operácia berie celé časti argumentov
0	min	a1 min a2	minimum z a1, a2
0	max	a1 max a2	maximum z a1, a2
5	%	a1 % a2	a2 percent z a1
5	prom	a1 prom a2	a2 promile z a1
5	sqx	a1 sqx a2	číslo a1 sa odmocní číslom a2
5	lg	a1 lg a2	vypočíta logaritmus o základe a2, z a1

*Pokračovanie na nasledujúcej strane*

*Pokračovanie z predchádzajúcej strany*

Priorita	Meno	Syntax	Opis
0	snd	a1 snd a2	tón s frekvenciou a1 Hz bude znieť a2 msec vypočíta sa absolútna hodnota z celej časti a1, pokiaľ je menšia ako 20, nahradí sa 20, pokiaľ je väčšia ako 16000, nahradí sa 16000

### 8.2.2 Práca s pamäťou

Priorita	Meno	Syntax	Opis
5	m/i	a1 m/i a2	číslo a1 sa vloží do pamäte a2
5	m/x	a1 m/x a2	číslo a1 sa vymení s číslom z pamäte a2
5	m/o	a1 m/o	číslo a1 sa nahradí číslom z pamäte a1
5	push	a1 push	číslo a1 sa vloží do zásobníka - POP; namiesto POP sa nahradí číslo zo zásobníka
5	pop*	a1 pop	a1 sa nahradí číslom z a1-tého miesta v zásobníku a dané číslo sa v zásobníku zruší, tiež aj tie pred ním
5	sum	a1 sum a2	číslo a1 sa pričíta k číslu z pamäti a2
5	prd	a1 prd a2	číslo z pamäti a2 sa vynásobí číslom a1
5	mpus	a1 mpus	a1 je bitová mapa 16 bitov – pamätí, ktoré sa majú uložiť do zásobníka
5	mpop	a1 mpop	a1 je bitová mapa pamätí, do ktorých sa uložia hodnoty zo zásobníka

### 8.2.3 Unárne operácie

Priorita	Meno	Syntax	Opis
5	sqr	a1 sqr	umocní a1
5	sqt	a1 sqt	odmocní a1
5	ln	a1 ln	vypočíta prirodzený logaritmus z a1
5	log	a1 log	vypočíta dekadický logaritmus z a1
5	e^x	a1 e^x	vypočíta exponenciálu z a1

*Pokračovanie na nasledujúcej strane*

\*externý program

*Pokračovanie z predchádzajúcej strany*

Priorita	Meno	Syntax	Opis
5	++	a1 ++	prehodí znamienko číslu a1
5	1/x	a1 1/x	vypočíta 1/a1
5	!	a1 !	vypočíta a1-faktoriál
5	int	a1 int	celá časť z a1
5	frac	a1 frac	vypočíta a1 mínus celá časť a1
5	rnd	a1 rnd	zaokrúhli a1 na celé číslo
5	abs	a1 abs	absolútna hodnota z a1
5	inc	a1 inc	inkrementuje a1
5	dec	a1 dec	dekrementuje a1
5	dly	a1 dly	počká a1 msec, vypočíta sa absolútna hodnota z celej časti a1, pokiaľ je väčšia ako 10000 nahradí sa 10000

### 8.2.4 Goniometrické funkcie

Priorita	Meno	Syntax	Opis
5	sin	a1 sin	sínus a1 v stupňoch
5	sir	a1 sir	sínus a1 v radiánoch
5	sig	a1 sig	sínus a1 v grádoch
5	cos	a1 cos	kosínus a1 v stupňoch
5	cor	a1 cor	kosínus a1 v radiánoch
5	cog	a1 cog	kosínus a1 v grádoch
5	tn	a1 tn	tangens a1 v stupňoch
5	tr	a1 tr	tangens a1 v radiánoch
5	tg	a1 tg	tangens a1 v grádoch
5	ctn*	a1 ctn	kotangens a1 v stupňoch
5	ctr*	a1 ctr	kotangens a1 v radiánoch
5	ctg*	a1 ctg	kotangens a1 v grádoch
5	asin*	a1 asin	arcus sínus do stupňov
5	asir*	a1 asir	arcus sínus do radiánov
5	asig*	a1 asig	arcus sínus do grádov
5	acsn*	a1 acsn	arcus kosínus do stupňov
5	acsr*	a1 acsr	arcus kosínus do radiánov
5	acsg*	a1 acsg	arcus kosínus do grádov
5	atn	a1 atn	arcus tangens do stupňov
5	atr	a1 atr	arcus tangens do radiánov
5	atg	a1 atg	arcus tangens do grádov
5	actn*	a1 actn	arcus kotangens do stupňov
5	actr*	a1 actr	arcus kotangens do radiánov
5	actg*	a1 actg	arcus kotangens do grádov
5	hsin*	a1 hsin	hyperbolický sínus a1 v stupňoch

*Pokračovanie na nasledujúcej strane*

*Pokračovanie z predchádzajúcej strany*

Priorita	Meno	Syntax	Opis
5	hsir*	a1 hsir	hyperbolický sínus a1 v radiánoch
5	hsig*	a1 hsig	hyperbolický sínus a1 v grádoch
5	hcos*	a1 hcos	hyperbolický kosínus a1 v stupňoch
5	hcsr*	a1 hcsr	hyperbolický kosínus a1 v radiánoch
5	hcsg*	a1 hcsg	hyperbolický kosínus a1 v grádoch
5	htn*	a1 htn	hyperbolický tangens a1 v stupňoch
5	htr*	a1 htr	hyperbolický tangens a1 v radiánoch
5	htg*	a1 htg	hyperbolický tangens a1 v grádoch
5	hctn*	a1 hctn	hyperbolický kotangens a1 v stupňoch
5	hctr*	a1 hctr	hyperbolický kotangens a1 v radiánoch
5	hctg*	a1 hctg	hyperbolický kotangens a1 v grádoch

### 8.2.5 Prevody uhlov

Priorita	Meno	Syntax	Opis
5	d>r	a1 d>r	stupne na radiány
5	d>g	a1 d>g	stupne na grády
5	r>d	a1 r>d	radiány na stupne
5	r>g	a1 r>g	radiány na grády
5	g>d	a1 g>d	grády na stupne
5	g>r	a1 g>r	grády na radiány

### 8.2.6 Prevody číselných sústav

Tieto prevody sa netýkajú číselnej hodnoty v pamäti, ide len o jej zobrazenie na displeji.

Priorita	Meno	Syntax	Opis
5	binr	a1 binr	zobrazí a1 binárne <sup>†</sup>
5	oktl	a1 oktl	zobrazí a1 oktalovo <sup>†</sup>
5	dcm	a1 dcm	zobrazí a1 dekadicky
5	hexd	a1 hexd	zobrazí a1 hexadecimálne <sup>†</sup>

Kalkulátor dovoľuje tiež zapisovať čísla vo všetkých uvedených sústavách. Binárne číslo sa musí končiť znakom B, oktalové znakom O a hexadecimálne znakom H.

Zobrazované zaokrúhlenie sa netýka vnútornej hodnoty čísla, ide len o zobrazovací formát.

<sup>†</sup>operácia berie z argumentu len celú časť

Priorita	Meno	Syntax	Opis
5	,x	a1 ,x	bude sa zobrazovať so zaokrúhlením na a1 miest

### 8.2.7 Štandardne dodávané unárne programy

Priorita	Meno	Syntax	Opis
5	demo	a1 demo	demonštrácia pomalého automatického, výpočtu a1 - fiktívny počet percent zo zárobku. Demonštrácia pamätí, cyklu a práce so zásobníkom
5	stup	a1 stup	demonštruje zvukové funkcie, a1 je dĺžka tónu v msec
5	okru	a1 okru	obvod kruhu s polomerom a1
5	pkru	a1 pkru	plocha kruhu s polomerom a1
5	pgul	a1 pgul	povrch gule s polomerom a1
5	ogul	a1 ogul	objem gule s polomerom a1
5	pop	a1 pop	viď 8.2.2
5	mhzv	a1 mhzv	prevedie frekvenciu v MHz na vlnovú dĺžku
5	vmhz	a1 vmhz	prevedie vlnovú dĺžku na frekvenciu v MHz

Ďalšie unárne programy implementujú goniometrické funkcie, viď 8.2.4.

### 8.2.8 Štandardne dodávané binárne programy

Priorita	Meno	Syntax	Opis
5	pyt	a1 pyt a2	kde a1=A, a2=B a výsledok je C zo vzorca Pytagorovej vety
5	tj	a1 tj a2	obsah trojuholníka, a1=základňa, a2=výška
5	oval	a1 oval a2	objem valca, a1=polomer podstavy, a2=výška
5	pval	a1 pval a2	povrch valca, a1=polomer podstavy, a2=výška
5	oku	a1 oku a2	objem kužeľa, a1=polomer podstavy, a2=výška
5	pku	a1 pku a2	povrch kužeľa, a1=polomer podstavy, a2=výška
5	nad	a1 nad a2	kombinačné číslo, $\binom{n}{k} = \frac{n!}{(n-k)!k!}$ , kde a1=n, a2=k
5	f&r	a1 f&r a2	pre frekvenciu a1 a odpor a2 vypočíta kapacitu v pF

*Pokračovanie na nasledujúcej strane*

*Pokračovanie z predchádzajúcej strany*

Priorita	Meno	Syntax	Opis
5	f&cv	a1 f&c a2	pre frekvenciu a1 a kapacitu a2 v pF vypočíta odpor
5	c&rf	a1 c&rf a2	pre kapacitu a1 v pF a odpor a2 vypočíta frekvenciu
5	t&r	a1 t&r a2	pre dĺžku kmitu a1 a odpor a2 vypočíta kapacitu v pF
5	t&c	a1 t&c a2	pre dĺžku kmitu a1 a kapacitu a2 v pF vypočíta odpor
5	c&rt	a1 c&rt a2	pre kapacitu a1 v pF a odpor a2 vypočíta dĺžku kmitu

### 8.2.9 Štandardne dodávané ternárne programy

Priorita	Meno	Syntax	Opis
3	lich	a1 + a2 lich a3	obsah lichobežníka, a1 je základňa 1, a2 je základňa 2, a3 je výška
5	dp	a1 % a2 dp a3	výpočet úrokov, a1 je uložený vklad, a2 - úrok v percentách, a3 - daň z úrokov v percentách
5	ozkr ozkv	a1 ozkr a2 ozkv a3	objem zrezaného kužeľa, a1 je polomer dolnej podstavy, a2=polomer hornej podstavy, a3=výška
5	pzkr pzkv	a1 pzkr a2 pzkv a3	povrch zrezaného kužeľa, a1=polomer dolnej podstavy, a2=polomer hornej podstavy, a3=výška
5	kvab kvc	a1 kvab a2 kvc a3	vypočíta kvadratickú rovnicu s koeficientami: a=a1, b=a2, c=a3, korene uloží do pamätí 00 a 01, diskriminant ostane na displeji. Ak je záporný, komplexné výsledky budú v pamätiach 00+01i a 02+03i. V tomto prípade sa nastaví pamäť 08 a 09 na jednotku,

*Pokračovanie na nasledujúcej strane*

*Pokračovanie z predchádzajúcej strany*

Priorita	Meno	Syntax	Opis
			čo znamená, že prvé dve dvojice pamätí sú komplexné čísla. Ak má rovnica len reálne korene, pamäť 08 sa nastaví na nulu.

### 8.2.10 Počítanie v komplexnom obore

Pre výpočty v komplexnom obore je preddefinovaných desať binárnych a päť unárnych programov. Pritom platia tieto konvencie:

1. pamäte 00 a 01, 02 a 03, 04 a 05, 06 a 07 tvoria dvojice algebraického zápisu komplexného čísla  $x_1 + x_2i$ ,
2. dvojice v pamätiach sa určujú číslom prvej pamäte, teda 0,2,4 alebo 6,
3. ak je v prvej dvojici komplexné číslo, pamäť 08 je potrebné nastaviť na jednotku, pre druhú dvojicu pamätí 09, pre tretiu pamäť 10 a pre štvrtú pamäť 11,
4. komplexné operácie používajú pamäte 08-11 na testovanie, či sa jedná o komplexné číslo,
5. pokiaľ používateľ pracuje len s komplexnými operáciami, pamäte 08-11 sú nastavované automaticky.

Priorita	Meno	Syntax	Opis
5	k#a	a1 k#a a2	vloží do pamätí 00 a 01 číslo $a_1+a_2i$ a na displeji nechá číslo 0, ktoré určuje dvojicu pamätí s komplexným číslom
5	k#b	a1 k#b a2	vloží do pamätí 02 a 03 číslo $a_1+a_2i$ a na displeji nechá číslo 2, ktoré určuje dvojicu pamätí s komplexným číslom
5	k#c	a1 k#c a2	vloží do pamätí 04 a 05 číslo $a_1+a_2i$ a na displeji nechá číslo 4, ktoré určuje dvojicu pamätí s komplexným číslom
5	k#d	a1 k#d a2	vloží do pamätí 06 a 07 číslo $a_1+a_2i$ a na displeji nechá číslo 6, ktoré určuje dvojicu pamätí s komplexným číslom
1	kplu	a1 kplu a2	komplexné sčítanie, kde $a_1$ a $a_2$ určujú komplexné čísla v pamätiach. Výsledok bude uložený do prvej dvojice pamätí a na displeji

*Pokračovanie na nasledujúcej strane*

*Pokračovanie z predchádzajúcej strany*

Priorita	Meno	Syntax	Opis
1	kmin	a1 kmin a2	ostane číslo 0, ktoré túto dvojicu určuje. komplexné odčítanie, kde a1 a a2 určujú komplexné čísla v pamätiach. Výsledok bude uložený do prvej dvojice pamätí a na displeji ostane číslo 0, ktoré túto dvojicu určuje.
2	kkra	a1 kkra a2	komplexné násobenie, kde a1 a a2 určujú komplexné čísla v pamätiach. Výsledok bude uložený do prvej dvojice pamätí a na displeji ostane číslo 0, ktoré túto dvojicu určuje.
2	kdel	a1 kdel a2	komplexné delenie, kde a1 a a2 určujú komplexné čísla v pamätiach. Výsledok bude uložený do prvej dvojice pamätí a na displeji ostane číslo 0, ktoré túto dvojicu určuje.
3	kmoc	a1 kmoc a2	komplexné umocňovanie na a2, kde a1 určuje komplexné čísla v pamätiach. Výsledok bude uložený do prvej dvojice pamätí a na displeji ostane číslo 0, ktoré túto dvojicu určuje.
3	kodm	a1 kodm a2	komplexná a2-há odmocnina, kde a1 určuje komplexné čísla v pamätiach. Výsledok bude uložený do prvej dvojice pamätí a na displeji ostane číslo 0, ktoré túto dvojicu určuje.
5	kabs	a1 kabs	absolútna hodnota z komplexného čísla uloženého v pamätiach a1.
5	kag	a1 kag	Výsledok je reálny, ostane na displeji. prevedie komplexné číslo v pamätiach a1 z algebraického tvaru na goniometrický, na displeji ostane a1
5	kg	a1 kg	prevedie komplexné číslo v pamätiach a1 z goniometrického tvaru na algebraický, na displeji ostane a1
5	kpus	a1 kpus	vloží komplexné číslo v pamätiach a1 do zásobníka
5	kpop	a1 kpop	vyberie zo zásobníka komplexné číslo a uloží ho do pamätí a1

### 8.2.11 Funkcie skokov v programoch

Priorita	Meno	Syntax	Opis
5	jmp	a1 jmp	skočí na prvý nasledujúci znak
5	jmxx	a1 jmxx a2	ak a2>0 skočí na a2-tý nasledujúci znak   ak a2<0 skočí na abs(a2)-hý predchádzajúci znak

*Pokračovanie na nasledujúcej strane*



Pokračovanie z predchádzajúcej strany

Priorita	Meno	Syntax	Opis
0	j>=	a1 j>= a2	ak $a1 \geq a2$ skočí na príslušný predchádzajúci znak \
0	j<=	a1 j<= a2	ak $a1 \leq a2$ skočí na príslušný predchádzajúci znak \
0	j>>	a1 j>> a2	ak $a1 > a2$ skočí na príslušný predchádzajúci znak \
0	j<<	a1 j<< a2	ak $a1 < a2$ skočí na príslušný predchádzajúci znak \
0	j=	a1 j= a2	ak $a1 = a2$ skočí na príslušný predchádzajúci znak \
0	j<>	a1 j<> a2	ak $a1 \neq a2$ skočí na príslušný predchádzajúci znak \

### 8.2.12 Tvorba programov

Znak & v programe ruší literál na displeji, nahrádza teda stlačenie klávesu `Esc`. Znak ~ zapne pomalý výpočet, používa sa pri kontrole a prezentácii. Po opätovnom zadaní ~ sa pomalý výpočet vypne. Znak \, | sa používajú pri skokoch a cykloch. Text v zložených zátvorkách sa berie ako komentár, kalkulátor ho ignoruje. Text v hranatých zátvorkách sa vypisuje do okna *Funkcia*, čím sa môžu komentovať jednotlivé kroky. Pri pomalom výpočte sa takto vypisuje prezentačný komentár. Znak , pozastaví výpočet, stlačením klávesu `Space` sa bude pokračovať. Ak sa stlačí kláves `Esc`, beh programu skončí. Znak ' vygeneruje akustický signál. Znak ; znamená koniec riadku; je ekvivalentný so stlačením klávesu `Enter`. Znak \$ zruší posledné číslo v zásobníku, ak je zásobník prázdny, nič nevykoná. Znak ? ukončí program, v jednej funkcii môže byť viac ukončení.

### 8.2.13 Programové súbory

Mená programov môžu mať maximálne štyri znaky, medzi ktorými môžu byť len malé písmená bez diakritiky. Programy sú textové súbory s príponou .un pre unárne programy a .bn? pre binárne programy. Unárne programy majú prioritu stanovenú implicitne na 5. Priorita binárnych programov je definovaná posledným znakom prípony: priorita 0 - .bn0, priorita 1 - .bn1... Maximálna priorita je 5.

Majme binárny program *plus* s prioritou 5. Operácia bude v súbore plus.bn5 s obsahom

+ POP =

Na displej kalkulátora sa potom zapíše  $3 \text{ plus } 3 =$ , výsledok bude podľa očakávania 6.

Preddefinované programy sa nachádzajú v adresári `~/rows/hcp`, volajú sa tak isto ako bežné operácie kalkulátora.

#### 8.2.14 Jazyk CALCULUS

Princíp jazyka CALCULUS je jednoduchý. Do definície sa zapisujú znaky tak, ako by ich používateľ zapisoval pri výpočte na displej kalkulátora. Rozdiely:

1. namiesto klávesu Enter sa zadáva `;`,
2. namiesto klávesu Esc sa zadáva `&`,
3. je povolený ľubovoľný počet medzier.

##### Unárne programy

Na začiatku unárneho programu sa vyžaduje zadanie operácie. Táto operácia na displeji nahradí identifikátor unárneho programu. Zapíše sa teda za číslo na displeji, za ktorým sa volal unárny program. Ďalšie operácie sa budú vykonávať tak, ako sú definované v súbore unárneho programu. Rovnítko v definícii unárneho programu vykoná všetky vnútorné operácie v jazyku CALCULUS. Po skončení unárneho programu očakáva kalkulátor zadanie ďalšej operácie alebo čísla podľa toho, čo ponechal na displeji unárny program.

##### Binárne programy

Pracujú podobne ako unárne, ale majú navyše k dispozícii ešte druhý operand. Tento operand je programu poskytnutý v zásobníku, ako je vidieť v podkapitole 8.2.13: príklad `plus.bn5`.

##### Nepodmienené skoky

Príkaz `jmp` hľadá v nasledujúcom zdrojovom kóde prvý výskyt znaku `|`. Výpočet bude pokračovať hneď za týmto znakom.

##### Podmienené skoky

Podľa podmienok, ktoré už boli opísané vyššie sa rozhodne, či sa pôjde v programe ďalej alebo sa vráti na príslušný predchádzajúci znak `\`. Podmienené skoky môžu byť do seba vnorené, potom sa zachová príslušnosť znakov `\` a podmieňovacích skokov tak, aby sa zachovalo vnorenie.

Kombináciou znakov ,, & a textu v hranatých zátvorkách sa umožní zobrazovanie medzivýsledkov aj s ich opisom v okne *Funkcia*.

### 8.2.15 Programy typu .big

Programy typu .big sú odlišené od unárnych a binárnych svojim volaním - nezadávajú sa medzi čísla ako operácie. Napíšu sa namiesto čísla s prefixom RUN> za ktorým nasleduje meno programu.

Výsledkom spustenia veľkého programu musí byť číslo, ktoré sa nahradí za identifikátor RUN>cislo. Ak je výsledkom viac čísel, je potrebné použiť zásobník alebo pamäte. Programy typu .big umožňujú využitie nového typu vstupu, pomocou načítavacej operácie READ. Táto operácia sa zapisuje v programe namiesto čísla. Operácia čaká na zadanie čísla z klávesnice.

Unárne, binárne programy a programy .big sa môžu navzájom v sebe volať, je povolená aj rekurzia. V rámci systému ROWS sú štandardne dodávané programy nasledujúce programy typu big: kv-rov, ktorý počíta reálne alebo komplexné kvadratické rovnice a tehly, ktorý počíta počet tehál v stene.

## 8.3 Práca s nahrávkami

Súčasťou systému ROWS je program *wp*, ktorý sa používa na spracovanie a orientáciu v súboroch .wav. *wp* dokáže pracovať s piatimi stopami, ktoré môžu byť rozdelené na bloky. *wp* takto umožňuje porovnávanie nahrávok a priamy prístup na označené časti. Ďalej popíšeme klávesové skratky programu *wp*. Klávesové skratky vyvolávajú v rôznych kontextoch používania rôzne akcie, preto sa k danej kombinácii kláves viaže viacero opisov.

Esc	koniec
Esc	zrušiť editáciu mena
Enter	prehrávanie
Enter	stop
Enter	vložiť meno
Shift + Enter	na akt. index
Ctrl + Enter	nahrávanie
Ctrl + Alt + Enter	na akt. blok

Space	pauza
Space	zrušiť pauzu
Shift + Space	označ koniec bloku
Ctrl + Space	označ index
PgUp / PgDn	prepínač stôp
PgUp / PgDn	vrchol/koniec
Up / Down	typy nastavení
Up / Down	prehliadať mená
Shift + Up / Down	naspäť/dopredu k indexu
Ctrl + Up / Down	typ pretáčania
Ctrl + Shift + Up / Down	naspäť/dopredu k medzere
Ctrl + Shift + Up / Down	stop/blok/indx
Left / Right	nastavovanie
Left / Right	posun kurzora
Shift + Left / Right	naspäť/dopredu pomalšie
Ctrl + Left / Right	naspäť/dopredu
Ctrl + Shift + Left / Right	naspäť/dopredu rýchlejšie
Ctrl + Alt + Up / Down	naspäť/dopredu k bloku
Home / End	hlasitosť
Home / End	začiatok/koniec
Ins	meno stopy
Ins	meno bloku
Ins	meno indexu
Ins	vkladať/prepisovať
Ctrl + Ins	meno adresára
Del	utlmiť/hlas
Del	vymazať znak
Ctrl + Shift + Del	vymazať index
Ctrl + Shift + Del	vymazať blok
Backspace	vymazať znak vľavo
F1	premenovať blok
F1	premenovať index
F2	uložiť zmeny

---

F3		zmeniť poradie blokov
F3		zmeniť poradie indexov
F4		bloky na stopu
F5		indexy na bloky
Shift	+	F8
Shift	+	F8

vymazať bloky  
vymazať indexy



## 9. Závery

Vytvorená softvérová vrstva modulu ROWS poskytuje služby pre používateľské rozhranie zamerané na podporu práce nevidiacich ľudí. Tieto služby boli testované päť rokov v implementovanom AmI systéme RUDO. Využitelnosť služieb bola overená v štyroch úrovniach:

1. práca na príkazovom riadku OS LINUX Debian,
2. služby používané aplikáciami so semigrafickým prostredím pre nevidiacich,
3. služby ponúkané pre celkové semigrafické príkazové prostredie HANIBAL – komandér pre nevidiacich,
4. služby zamerané špeciálne na obsluhu zariadení AmI systému RUDO.

Súčasťou AmI systému RUDO je modul ROWS, ktorý je využiteľný aj mimo ambientu prostredia systému RUDO. ROWS nainštalovaný na počítač vytvorí z neho kompenzačnú pomôcku pre nevidiacich, ktorá je použiteľná v bežnom, všeobecnom zmysle. Keď sa takýto počítač dostane do dosahu ambientu systému RUDO (napr. do dosahu WiFi), pripojí sa na AmI systém RUDO a ponúka všetky služby tohoto ambientného systému.

Opodstatnenosť vývoja AmI systému RUDO a jeho využiteľnosť pre nevidiacich vo všeobecnosti je ukázaná v článku [10].

Modul ROWS ako bezplatná kompenzačná pomôcka pre nevidiacich sa ponúka všetkým nevidiacim používateľom na stránke [www.systemrows.eu](http://www.systemrows.eu).

## 9.1 Pokrytie práce na príkazovom riadku

Nevidiaci človek vzhľadom na svoj hendikep prijíma informácie s ťažkosťami a omnoho pomalšie ako vidiaci. Preto je potrebné uvedomiť si, že spätný popis ikon grafických prostredí a čítanie rozsiahlych ponúk časovo znevýhodňuje nevidiaceho človeka, čo sa prejaví jeho schopnosťou uplatniť sa na trhu práce.

Na druhej strane nevidiaci človek môže písať rovnako rýchlo ako aj vidiaci.

Toto je veľmi dôležitý aspekt, na základe ktorého je možné tvrdiť, že pre nevidiacich ľudí môže mať práca na príkazovom riadku zmysel a môže sa stať časovo veľmi efektívnou.

V rámci nášho výskumu bola táto myšlienka uvedená do praxe realizáciou vývoja špeciálneho používateľského rozhrania AmI systému RUDO.

Služby príkazového riadku modulu ROWS boli testované päť rokov nevidiacim človekom, ktorý nepoužíval iné používateľské rozhranie. Počítač využíval ako:

1. bežný používateľ,
2. administrátor,
3. bežný používateľ AmI systému RUDO,
4. administrátor AmI systému RUDO.

Služby príkazového riadku modulu ROWS, tak ako sú navrhnuté v kapitolách 5 a 6, spolu s vybranými príkazmi OS LINUX z kapitoly 4, plne pokryli používateľské potreby nevidiaceho človeka vo všetkých vyššie uvedených štyroch bodoch.

## 9.2 Pokrytie služieb pre aplikácie

Semigrafické výstupy sú pre nevidiaceho používateľa veľmi zaujímavé tým, že na jednej strane môže ľahko čítať text (keďže ide o textovú konzolu), na druhej strane má takéto prostredie výhody výberu v krátkych ponukách.



Semigrafické prostredia môžu byť zároveň dôležitým nástrojom komunikácie od nevidiaceho smerom k vidiacemu človeku. Je možné použiť ich s dataprojektorom - nevidiacemu počítač rozpráva syntetickým hlasom a vidiaci vidia premietaný obraz. Nevidiaci človek môže takýmto spôsobom prezentovať svoju prácu.

Aplikácie so semigrafickým výstupom pre nevidiacich majú na softvérovú vrstvu modulu ROWS ďalšie požiadavky:

1. potreba vypnutia čítacieho servra a potreba priameho použitia syntetizéra GOBLIN (viď schému 1.3),
2. potreba obmedzenia čítania z displeja na menšie súradnicami špecifikované okno (viď schému 1.3),
3. potreba generovania orientačných zvukov, ktoré v aplikáciách symbolizujú typ alebo úspešnosť vykonávanej operácie.

Vyššie uvedené tri služby sú poskytované v rámci sieťového protokolu softvérového servra syntetizéra GOBLIN a čítacieho softvérového servra (viď schému 1.3). Podrobný popis týchto sieťových protokolov presahuje rámec tejto publikácie.

Niektoré príklady semigrafických aplikácií sú uvedené v kapitolách 7 a 8.

Pri vývoji semigrafických aplikácií počas piatich rokov testovania boli poskytované sieťové služby softvérovej vrstvy modulu ROWS postačujúce pre všetky vyvíjané programy so semigrafickým rozhraním.

### 9.3 Pokrytie služieb pre komandér

Komandér HANIBAL - semigrafické používateľské rozhranie pre nevidiacich - vyžaduje použitie služieb modulu ROWS v sieťovom aj príkazovom zmysle. V tejto verzii AmI systému RUDO je komandér HANIBAL už vytvorený, ale zatiaľ ešte neprebehla časovo primeraná fáza empirického testovania v praxi.

Na základe testovania, ktoré trvalo šesť mesiacov, môžeme predpokladať, že budú aj pre túto aplikáciu poskytované služby postačujúce.

Aby mohol byť HANIBAL – komandér pre nevidiacich naozaj efektívny používateľský nástroj, je nutné takýto produkt dlhší čas empiricky testovať v praxi a na základe skúseností modifikovať služby jeho semigrafického používateľského rozhrania. Pre tento

účel bol navrhnutý rovnomenný jazyk HANIBAL, pomocou ktorého môže nevidiaci používateľ samostatne meniť služby prostredia HANIBAL v plnom rozsahu.

## 9.4 Pokrytie služieb pre AmI systém RUDO

Pri bežnej a administrátorskej obsluhu zariadení AmI systému RUDO sa používajú sieťové aplikácie so semigrafickým používateľským rozhraním a aplikácie príkazového charakteru.

Softvérová vrstva modulu ROWS môže preto v zmysle služieb plne pokryť všetky požiadavky AmI systému podľa podkapitol 9.1 a 9.2.

AmI systém RUDO môžu obsluhovať aj vidiaci ľudia, keď použijú sieťovo orientované semigrafické aplikácie. Toto obmedzenie je istým nedostatkom, ktorý bude v budúcnosti riešený vývojom plne grafických klientov.

Môžeme odhadnúť, že služby softvérovej vrstvy modulu ROWS budú postačujúce aj pre tieto aplikácie, tento záver však nie je potvrdený.

## 9.5 Záver

AmI systémy v inteligentných budovách ponúkajú veľmi zaujímavé aplikácie aj v oblasti asistovaného bývania pre nevidiacich ľudí. Jedná sa o oblasť informatiky, v ktorej sa pomaly otvárajú dvere do sveta malých zázrakov.

Vytvorená softvérová vrstva modulu ROWS a špeciálna inštalácia operačného systému Linux Debian na domový server a klientské stanice sú dôležitou prípravou na prevádzku AmI systému RUDO.

Prototyp tohoto systému bude na Fakulte prírodných vied Univerzity Mateja Bela ďalej vyvíjaný a empiricky testovaný. Výsledky tohoto vývoja môžu v budúcnosti uľahčiť bývanie pomerne veľkej skupiny nevidiacich ľudí.

## 10. Skratky a pojmy

~ • Špeciálny znak, ktorý sa používa pod operačným systémom Linux ako skratka domovského adresára.

**AAL systém** • Informatický systém vytvárajúci prostredie asistovaného bývania, zväčša zameraný na starostlivosť pre starších ľudí

**AmI systém** • Informatický systém vytvárajúci prostredie asistovaného bývania vo všeobecnom zmysle.

**Braillove, bodové písmo** • Písmo pozostávajúce zo šiestich alebo ôsmich hmatom rozpoznaiteľných bodov, ktoré používajú nevidiaci ľudia.

**Braillova klávesnica** • Bežná počítačová klávesnica softvérovo prispôbena na zápis pomocou Braillovho písma. Klávesy „asdf“ a „jkl;“ reprezentujú jednotlivé body Braillovho písma a je potrebné stláčať ich súčasne v príslušných kombináciách.

**Braillov, hmatový displej** • Displej zobrazujúci jediný riadok textu v Braillovom písme. Body jednotlivých Braillových znakov sa

mechanicky zasúvajú alebo vysúvajú, čím sa stávajú hmatateľnými pomocou bruška čítacieho prsta – ľavého ukazováka.

**CALCULUS** • Jazyk pomocou ktorého sa definujú funkcie v programovateľnom kalkulátore opísanom v tejto publikácii.

**FAUN** • Jazyk pomocou ktorého sa definujú melódie pozostávajúce z tónov generátora kmitov. Jeden z opísaných programových modulov ho ponúka ako prostriedok na vytváranie orientačných zvukov pre nevidiacich používateľov počítačov.

**GOBLIN** • Slovenský rečový syntetizér, ktorý je súčasťou opisovaného systému.

**HANIBAL** • Semigrafické používateľské rozhranie pre nevidiacich, ktoré je postavené na využití opisovanej softvérovej vrstvy. V súčasnosti je v štádiu testovania, existuje len vo svojej beta verzii. Rovnomenný jazyk sa používa na definíciu štruktúry okien používateľského rozhrania HANIBAL a definíciu funkcií v rámci špecifikovaných pracovných okien.

**KRO-n** • Parameter špecifikuje počet súčasne stlačených klávesov ktoré dokáže bežná počítačová klávesnica akceptovať. Klávesnica používaná na písanie Braillovoho písma musí mať parameter KRO aspoň 8.

**LAN** • Lokálna počítačová sieť s metalickými dátovými prepojeniami.

**OS** • Operačný systém.

**OCR systém** • Systém rozpoznávania textu, ktorý je uložený vo forme grafického rastra. Takýto systém je využiteľný pre nevidiacich, lebo prevádza skenovaný dokument na text, ktorý môže byť čítaný pomocou rečového syntetizéra.

**PC** • Osobný počítač.

**ROWS** • Názov programového systému opísaného v tejto publikácii, ktorý vytvorí z počítača pomôcku pre nevidiacich.

**RUDO** • Názov AmI systému, opísaného v tejto publikácii, ktorého súčasťou je aj modul ROWS.

**UEFI** • Zavádzací systém počítačov umiestnený na prvej partícii na disku. Až po jeho inicializácii sa zavádzajú operačné systémy ako Windows alebo Linux.

**WAN** • Hierarchicky vyššia počítačová sieť, v rámci ktorej je pripojená lokálna počítačová sieť.

**WiFi** • Lokálna bezdrôtová počítačová sieť.

**WORF** • Hypertextový jazyk používaný v systéme ROWS na rýchlu prípravu dokumentov nevidiacimi používateľmi.

**Z-Wave** • Bezdrôtová sieť používaná na prepojenie počítača so zariadeniami domácej automatizácie napríklad s radiátorovými hlavicami.



## Zoznam bibliografických odkazov

- [1] AALIANCE. *Summary of Standards*. 2017. URL: <http://nero.offis.de/projects/aalliance2/start> (citované 30. 06. 2017).
- [2] M. Amiribesheli, A. Benmansour a A. Bouchachia. „A review of smart homes in healthcare“. In: *Journal of Ambient Intelligence and Humanized Computing* 6.4 (2015), strany 495–517. ISSN: 1868-5145. DOI: 10.1007/s12652-015-0270-2.
- [3] J. Bhatt a H.K. Verma. „Design and Development of Wired Building Automation Systems“. In: *Energy and Buildings* 103.Supplement C (2015), strany 396–413. ISSN: 0378-7788. DOI: <https://doi.org/10.1016/j.enbuild.2015.02.054>.
- [4] M. Darwish et al. „A Comparison Between Ambient Assisted Living Systems“. In: *Smart Homes and Health Teleatics: 12th International Conference, ICOST 2014, Denver, CO, USA, June 25-27, 2014, Revised Papers*. Editované C. Boudine et al. Cham: Springer International Publishing, 2015, strany 231–237. ISBN: 978-3-319-14424-5. DOI: 10.1007/978-3-319-14424-5\_26.

- [5] A. Dasios et al. „Hands-On Experiences in Deploying Cost-Effective Ambient-Assisted Living Systems“. In: *Sensors* 15.6 (2015), strany 14487–14512. ISSN: 1424-8220. DOI: 10.3390/s150614487.
- [6] T. Gallagher et al. „Indoor navigation for people who are blind or vision impaired: where are we and where are we going?“ In: *Journal of Location Based Services* 8.1 (2014), strany 54–73. DOI: 10.1080/17489725.2014.895062.
- [7] A. I. Galushkin. *Neural Networks Theory*. 1. vydanie. Springer-Verlag Berlin Heidelberg, 2007. DOI: 10.1007/978-3-540-48125-6.
- [8] F. Gullà et al. „Design Adaptable and Adaptive User Interfaces: A Method to Manage the Information“. In: *Ambient Assisted Living: Italian Forum 2014*. Editované B. Andò et al. Cham: Springer International Publishing, 2015, strany 47–58. ISBN: 978-3-319-18374-9. DOI: 10.1007/978-3-319-18374-9\_5.
- [9] C. Hill, G. Raymond a I. Yeung. *Ambient Assisted Living Technology*. Final report of project E-project-051113-192757. 2017. URL: <https://www.wpi.edu/Pubs/Eproject/Available/E-project-051113-192757/> (citované 29. 06. 2017).
- [10] Smutný Z. Hudec M. „RUDO: A Home Ambient Intelligence System for Blind People“. In: *Sensors* 8.17 (2017), E1926. DOI: 10.3390/s17081926.
- [11] M. Hudec. „Vektorový priestor pohybu osôb“. In: *Information Technologies - Applications and Theory*. Košice: Department of Computer Science Faculty of Science Pavol Jozef Šafárik University, 2002, strany 103–111.
- [12] M. Hudec. *Informačné technológie v softwarových kompenzačných aplikáciách*. Banská Bystrica: Ústav vedy a výskumu UMB, 2006.
- [13] M. Hudec. „Vývojové prostredie na prácu s hlasom a neurónovou sieťou pre zrakovo hendikepovaných“. In: *Informatika v škole* 35 (2008), strany 37–43.
- [14] M. Hudec. *Konštrukcia neurónových sietí skladaním jednotiek stratovej kompresie*. Banská Bystrica: Inštitút matematiky a informatiky FPV UMB, 2009.



- [15] M. Hudec. „Kompenzačný softvér pre nevidiacich v oblasti vedeckej informatiky“. In: *DidInfo 2011*. Editované L. Huraj a D. Hováthová. Banská Bystrica: Katedra informatiky FPV UMB v Banskej Bystrici, 2011, strany 103–111.
- [16] M. Hudec. *Naprogramujte si svoj rečový syntetizér s neurónovou sieťou*. Banská Bystrica: Inštitút matematiky a informatiky FPV UMB, 2013.
- [17] M. Hudec. „Inteligentné budovy s asistentom pre nevidiacich“. In: *Acta Informatica Pragensia* 5 (2016), strany 4–17.
- [18] L. Jelínek. *Jádro systému Linux. Kompletní průvodce programátora*. Brno: Computer Press, 2008.
- [19] G. Karmakar, A. Kabra a K. Ramamritham. „Maintaining thermal comfort in buildings: feasibility, algorithms, implementation, evaluation“. In: *Real-Time Systems* 51.5 (2015), strany 485–525. ISSN: 1573-1383. DOI: 10.1007/s11241-015-9231-2.
- [20] K. R. Konda et al. „Smart Camera Reconfiguration in Assisted Home Environments for Elderly Care“. In: *Computer Vision - ECCV 2014 Workshops: Zurich, Switzerland, September 6-7 and 12, 2014, Proceedings, Part IV*. Editované L. Agapito, M. M. Bronstein a C. Rother. Cham: Springer International Publishing, 2015, strany 45–58. DOI: 10.1007/978-3-319-16220-1\_4.
- [21] J. R. Koza et al. *Genetic Programming IV. Routine Human-Competitive Machine Intelligence*. Springer US, 2003.
- [22] F. Leotta a M. Mecella. „PLaTHEA: a marker-less people localization and tracking system for home automation“. In: *Software: Practice and Experience* 45.6 (2015), strany 801–835. ISSN: 1097-024X. DOI: 10.1002/spe.2262.
- [23] M. L. Mekhali et al. „A Compressive Sensing Approach to Describe Indoor Scenes for Blind People“. In: *IEEE Transactions on Circuits and Systems for Video Technology* 25.7 (2015), strany 1246–1257. DOI: 10.1109/TCSVT.2014.2372371.
- [24] T. D. P. Mendes et al. „Smart Home Communication Technologies and Applications: Wireless Protocol Assessment for Home Area Network Resources“. In: *Energies* 8.7 (2015), strany 7279–7311. ISSN: 1996-1073. DOI: 10.3390/en8077279.

- [25] A. Nait Aicha, G. Englebienne a B. Kröse. „Continuous Gait Velocity Analysis Using Ambient Sensors in a Smart Home“. In: *Ambient Intelligence: 12th European Conference, AmI 2015, Athens, Greece, November 11-13, 2015, Proceedings*. Editované B. De Ruyter et al. Cham: Springer International Publishing, 2015, strany 219–235. ISBN: 978-3-319-26005-1. DOI: 10.1007/978-3-319-26005-1\_15.
- [26] T. Nef et al. „Evaluation of Three State-of-the-Art Classifiers for Recognition of Activities of Daily Living from Smart Home Ambient Data“. In: *Sensors* 15.5 (2015), strany 11725–11740. ISSN: 1424-8220. DOI: 10.3390/s150511725.
- [27] R. Neruda a J. Šíma. *Teoretické otázky neuronových sítí*. Praha: MATFYZPRESS vyd. MFF UK, 1996.
- [28] Q. Ni, A. B. García Hernando. a I. Pau de la Cruz. „The Elderly’s Independent Living in Smart Homes: A Characterization of Activities and Sensing Infrastructure Survey to Facilitate Services Development“. In: *Sensors* 15.5 (2015), strany 11312–11362. ISSN: 1424-8220. DOI: 10.3390/s150511312.
- [29] N. Nikolaev a H. Iba. *Adaptive Learning of Polynomial Networks. Genetic Programming, Backpropagation and Bayesian Methods*. Springer US, 2006.
- [30] J. Psutka et al. *Mluvíme s počítačem česky*. Praha: Academia, 2006.
- [31] I. Roper, L. Pilgrim a G. Simpson. „Guide to smart home wireless technologies“. In: *Journal of the Institute of Telecommunications Professionals* 9.3 (2015), strany 29–33.
- [32] S. Samarasinghe. *Neural Networks for Applied Sciences and Engineering. From Fundamentals to Complex Pattern Recognition*. 1. vydanie. Auerbach Publications, 2006.
- [33] M. G. Sobell. *Mistrovství v Linuxu. Příkazový řádek, shell, programování*. Brno: Computer Press, 2007.
- [34] J. Uhlíř et al. *Technologie hlasových komunikací*. Praha: Nakladatelství ČVUT, 2007.

- [35] M. Vacher et al. „Evaluation of a Context-Aware Voice Interface for Ambient Assisted Living: Qualitative User Study vs. Quantitative System Evaluation“. In: *ACM Trans. Access. Comput.* 7.2 (2015), 5:1–5:36. ISSN: 1936-7228. DOI: 10.1145/2738047.
- [36] A. F. Villan et al. „Face Recognition and Spoofing Detection System Adapted To Visually-Impaired People“. In: *IEEE Latin America Transactions* 14.2 (2016), strany 913–921.