

EDUKAČNÉ PROSTREDIE NA PROGRAMOVANIE HUDBY PRÍSTUPNÉ PRE NEVIDIACICH ŽIAKOV SEKUNDÁRNEHO VZDELÁVANIA

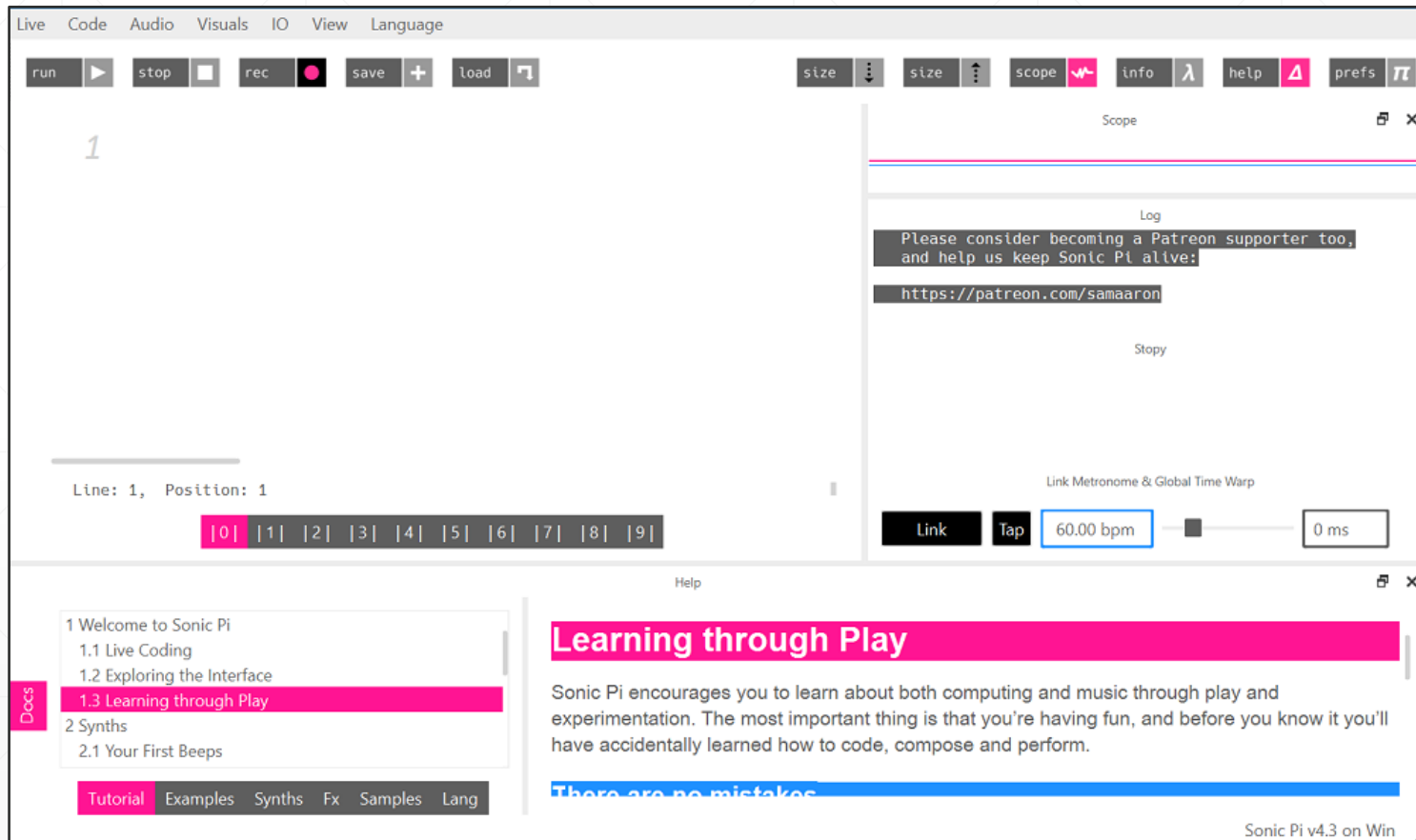
Jakub Švorc

Detaily k práci

- Školiteľka: doc. RNDr. Ľudmila Jašková, PhD.
 - Cieľ práce: Vytvoriť programovacie prostredie umožňujúce programovať hudbu pozostávajúcu z viacerých paralelne znejúcich melódií. Dôraz bude kladený na zabezpečenie plnej prístupnosti a efektívnej práce s editorom kódu pre žiakov so zrakovým postihnutím
 - Cieľová skupina: 2. stupeň ZŠ
-

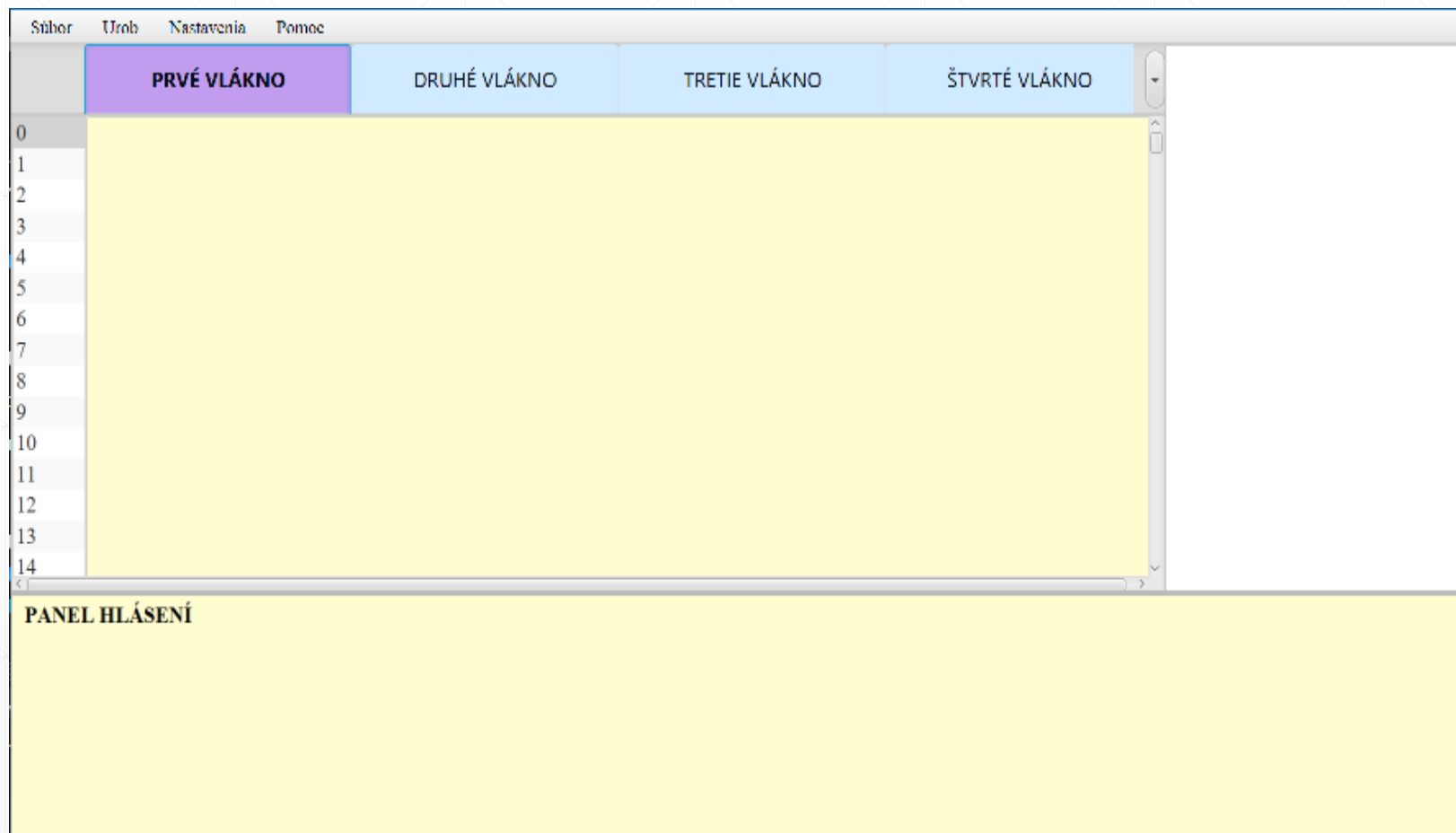
Podobné aplikácie

SonicPi



- Sam Aaron
- Podpora štandardného aj zvukového programovania
- Jednoduchá aj komplexná syntax (veľa parametrov) a základné programové konštrukcie
- IDE s vlastným jazykom
- Bohatá škála vytvárania zvukov a ich rôzne úpravy

TonIK2



- Práca s hotovými príkazmi
- Output panel
- Práca s premennými
- Programové konštrukcie
- Plne ovládateľné klávesnicou

Implementácia

Postup pri riešení práce

- Spracovanie článkov
 - Prehľad vhodných technológií
 - Zadefinovanie syntaxe
 - Aplikácia na test s čítačom obrazovky
 - Porovnanie kompilácie proti interpretácií jazyka
 - Začatie implementácie
-

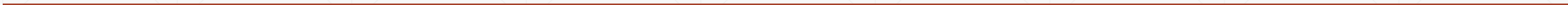
Implementácia

- Voľba vhodného nástroja
 - Python vs. JavaFx vs. .NET WPF + knižnice
 - Malá testovacia aplikácia
 - Tvorba GUI
 - Interpreter
 - naivný spôsob spracovania programu
 - Kompilátor a virtuálna mašina
 - zjednodušený bez programovacích konštrukcií
-

Implementácia - pokračovanie

- Prehrávanie tónu
 - Spracovanie dodatočných parametrov
 - Zmena nástroja
 - Cyklus s počtom opakovaní (for loop)
 - Kontroly syntaxe počas kompilácie
 - Terminál na debugovanie a hlásenie chýb
-

Ukážka

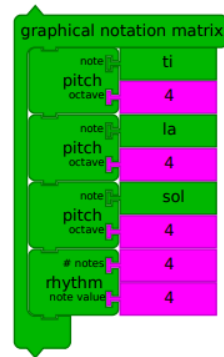


Zdroje

Music Blocks: A Musical Microworld

- Walter Bender
- Devin Ulibarri
- Yash Khandelwal

```
mouse = {
  g'4 a'4 b'4 g'4 g'4 a'4 b'4 g'4 b'4 c''4 d''2 b'4 c''4 d''2 d''8 e''8 d''8 c''8
  b'4 g'4 d''8 e''8 d''8 c''8 b'4 g'4 g'4 d'4 g'2 g'4 d'4 g'2}
\score { << \new Staff = "treble"
{ \clef "treble" \set Staff.instrumentName = #"mouse" \mouse } >> \layout { } }
```



Solfa	⏵	⏴	⏶	⏷
ti ₄				
la ₄				
sol ₄				
rhythmic note values	1/4 ↓	1/4 ↓	1/4 ↓	1/4 ↓

Solfa	⏵	⏴	⏶	⏷
ti ₄				
la ₄				
sol ₄				
rhythmic note values	1/4 ↓	1/4 ↓	1/4 ↓	1/4 ↓



Designing educational programming tools for the blind: mitigating the inequality of coding in schools

- Clarissa Correa de Oliveira

2.5.5 Hello Ruby

Created for children aged 5 and up, Hello Ruby is a playful exploration of computers, technology and programming, through games, exercises and apps. Ruby, the main character, meets friends along her adventures and together they solve problems. Activities range from designing computers out of paper by fitting bits and pieces that constitute a computer, to activities that use only the body as a tool for interaction, with concepts such as 'loops' embodied by repeating a sequence of instructions that include hand clapping and jumping.



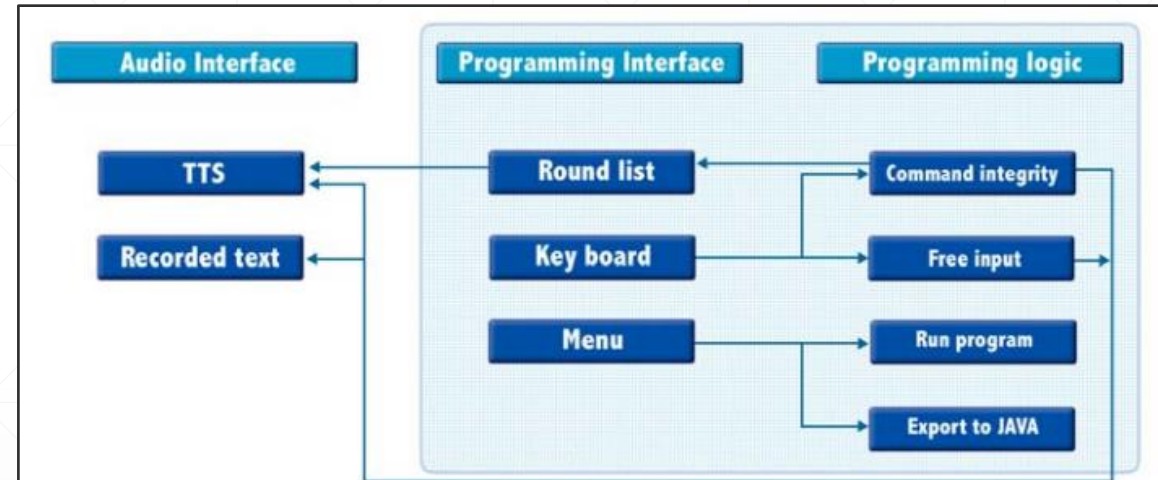
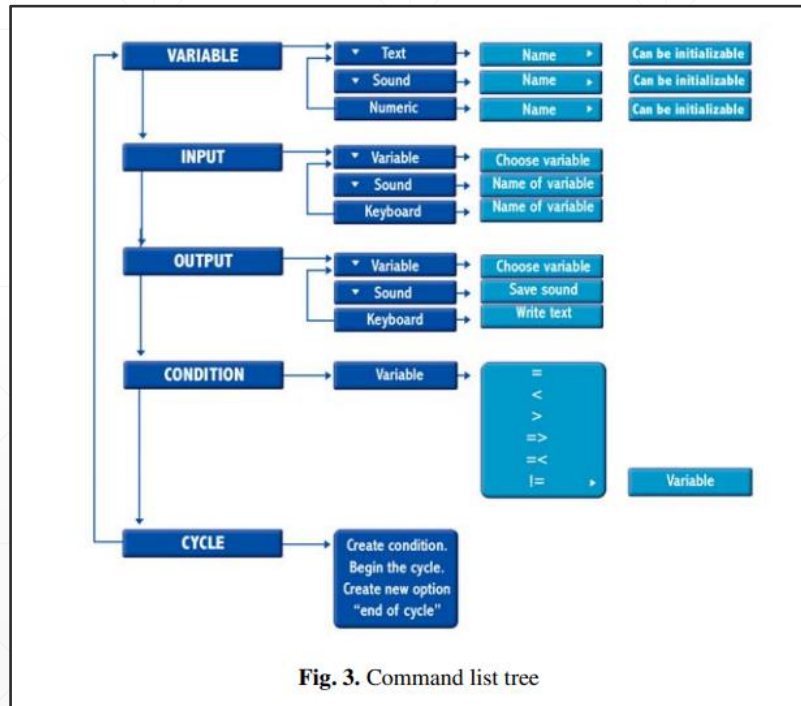
Figure 7: Children play Universal Remote Control, an exercise where children get to practice building a remote control and giving commands. Source: <http://blog.helloruby.com/post/131553872243/for-educators-lesson-plan-for-universal-remote>. (Access on April 24, 2017)

3.2.1 State-of-the-Art Review

As a first step in developing this research, and as a practice that follows the research throughout most of its process, literature review served as an informative basis. The search for published material related to equality in education, computational thinking, coding and blindness, available in media such as articles, books, online forums, videos and websites, helped to understand the current state of blind children's access to coding education and gather evidence to support this study. Together with the selection and analysis of up-to-date teaching tools intended for learning computational thinking and coding at a young age, a panorama of the tools available today served as reference for the do's and don'ts of Designing for the blind.

APL: Audio Programming Language for Blind Learners

- Jaime Sánchez



3.2 Implementation

APL was developed by using Java 1.4.1 and FreeTTS Java speech synthesizer. These are fine tools for media management and close to the machine logic. APL has the following modules: DataBase, Integrity, Kernel, CHI and generator de JAVA.

Budúce ciele

Budúce cieľe

- Práca s premennými
 - Podprogramy
 - Vetvenie
 - Vlákna a súbežné prehrávanie
 - Kompatibilita s NVDA
 - Skúmanie používania v praxi spolu so žiakmi cieľovej skupiny
-

Ďakujem
