

Obhajoba projektu IFJ12
2012/2013

Interpreter jazyka IFJ12

Martin Maga, Vít Mojžíš, Viktor Malík, Vojtěch Meca, Jiří Macků

Rozdelenie projektu

- Lexikálny analyzátor

Rozdelenie projektu

- Lexikálny analyzátor
- Syntaktický analyzátor

Rozdelenie projektu

- Lexikálny analyzátor
- Syntaktický analyzátor
- Sémantický analyzátor

Rozdelenie projektu

- Lexikálny analyzátor
- Syntaktický analyzátor
- Sémantický analyzátor
- Interpreter

Lexikálny analyzátor

- Konečný automat pre rozpoznanie lexém
- Identifikátory, čísla(numeric), reťazce, escape sekvencie
- Rozpoznávanie kľúčových a rezervovaných slov pomocou poľa reťazcov
- Identifikácia lexikálnych chýb

Syntaxou riadený preklad

- Pozostáva: 1. syntaktický analyzátor 2.sémantický analyzátor
- LL-gramatika, zásobník a postfix notácia
- Princíp: tabuľka na základe, ktorej je riadená redukcia a vyhodnocovanie výrazov
- Tabuľka funkcií, ktoré neboli definované pri volaní
- Ukladanie tokenov do zásobníku a následné vyhodnocovanie
- Analýza zhora nadol
- Súčasná syntaktická a sémantická kontrola

Interpreter

- Kód inštrukcií: ukazateľ na operand, ukazateľ na funkciu, typ inštrukcie
- Ukladanie inštrukcií na zásobník
- Funkcia prog errors interpreter
- Neznalosť problematiky problému

Find

- Vstavaná funkcia, ktorá slúži na vyhľadanie podreťazca v reťazci
- Implementácia algoritmom: *Knuth-Morris-Pratt*
- Princíp: Konečný automat s 3 uzlami: 1. STOP 2. READ 3. START
- Tvorba pomocnej tabuľky na porovnávanie
- Naplnenie tabuľky hodnotami
- Porovnávanie reťazca s tabuľkou v prípade nezhody pokračuje ďalej
- Algoritmus končí nájdeným podreťazca a vrátením indexa
- V prípade neúspechu vracia hodnotu -1
- Zložitosť: lineárna $O(n+m)$

Sort

- Vstavovaná funkcia, ktorá slúži k zoradeniu znakov v reťazci
- Implementácia algoritmom: *Shell-sort*
- Priama metóda, princíp bubble sort-u
- Princíp: Postupná aplikácia bubble sort-u s rôznymi hodnotami kroku
- V poslednej fáze je aplikovaný klasický bubble sort
- Nestabilný, „in situ“, kvadratická zložitosť
- Najlepší radiaci algoritmus

Tabuľka symbolov

- Implementácia hashovacou tabuľkou s jednosmerne viazanými zoznamami synonym
- Údaj o počte prvkoch v zoznamoch z dôvodu rekurzie
- Pri vyhľadávaní použitie najnovšej inštancie hľadanej premennej
- Pri návrate z funkcie su kópie najbližšie vrcholu zásobníku odstránené
- Zložitosť je daná prístup k nadlhšej postupnosti zreťazených synonym

Použité zdroje

- Opora z Predmetu IAL
- Opora z Predmetu IFL
- <http://falconpl.org/>

Ďakujem za pozornosť