



Projektová dokumentace IPP

Interpret nestrukturovaného imperativního jazyka IPPcode23

Nikita Kotvitskiy (xkotvi01)

Obsah

1	Návrh a implementace I. části projektu	i
2	Konečné automaty použité v parseru	i

1 Návrh a implementace I. části projektu

Hlavní částí implementace je třída "Parser", jejíž rozhraní se skládá z jediné metody ParseProgram s jedním parametrem \$source (obsahuje zdroj vstupu). Tato třída je schopná provést lexikální a syntaktickou analýzu kódu a zároveň vytvořit XML reprezentaci celého programu.

Analýza programu je kontextově závislá. Kód se čte řádek po řádku. Před analýzou řádek se upravuje: odstraňují se sekvence bílých znaků a komentáře.

Samotná analýza se skládá ze dvou částí: zaprvé se provádí kontrola hlavičky. Špatná hlavička způsobuje výskyt chyby 21. Pak se kontrolují příkazy. Při zpracování každého příkazu se vytváří nový XML elementy "instruction"s příslušnými argumenty. Neznámé příkazy vedou na chybu 22. Každý příkaz se rozděluje na pole lexémů. První prvek pole je identifikátor příkazu, ostatní jsou jeho argumenty. Podle identifikátoru lze rozpoznat typ. Typ reprezentuje počet a typ argumentu příkazu. Například typ "var_symb_symb" znamená, že příkaz má tři argumenty: jeden identifikátor proměnné a dva symboly. Špatný počet argumentu vede na chybu 23. Pak se kontrolují jednotlivé argumenty (a zároveň se vytváří příslušné XML elementy). Kontrola správnosti argumentů je implementovaná v podobě konečných automatů. Špatný formát argumentu vede na chybu 23.

Jakmile všechny řádky programu jsou zpracované, XML dokument se převádí na řetězec a odesílá se na standardní výstup STDOUT.

2 Konečné automaty použité v parseru

V parseru se používají tři různé konečné automaty: Var analys, Name analys a String analys.

