

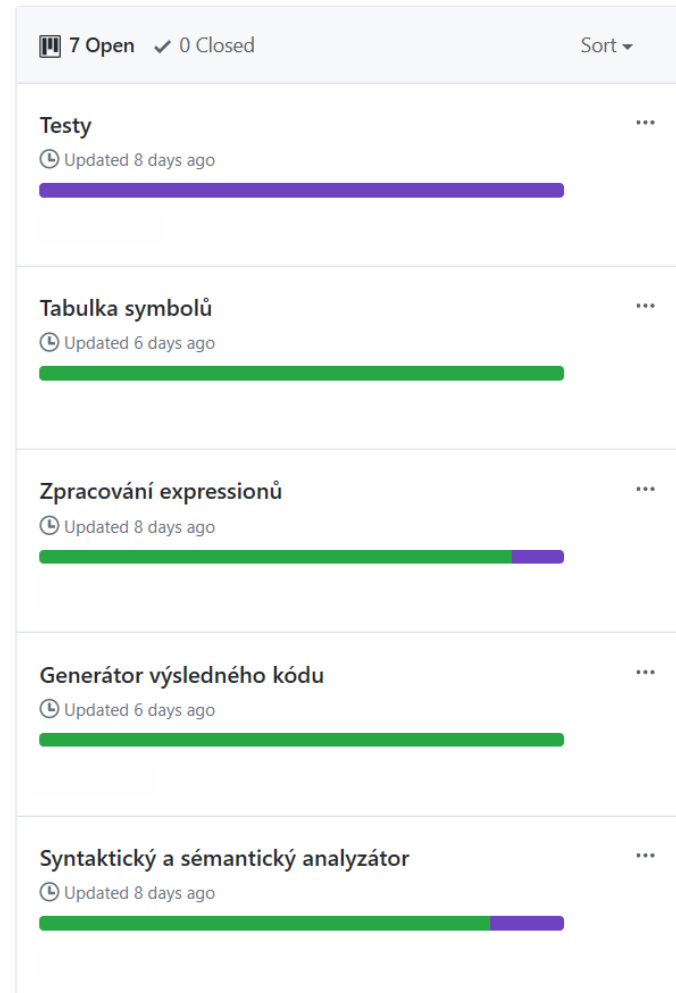
Implementace překladače imperativního jazyka IFJ18

Tým 013, varianta I

Klára Formánková	xforma14
Jan Láncoš	xlanco00
Vít Šebela	xsebel04
Jan Chaloupka	xchalo16

Práce v týmu

- GitHub
 - Verzování (git)
 - Rozdělení úkolů
 - Sledování postupu
- Facebook
- Schůzky



Organizace práce (GitHub funkce „Projects“)

Struktura překladače

Struktura překladače



Lexikální analyzátor

- Konečný automat
 - Při načtení ID – detekce klíčového slova
 - Implementace rozšíření BASE
- Vrací seznam lexémů (tokeny)
 - Hodnota tokenu není nijak upravená (10e2 → „10e2“)
 - Převod pro IFJcode18 probíhá při generaci kódu

Datová struktura Token	
Typ	
Hodnota	
Předchozí token	
Následující token	
Číslo řádku	
Pozice na řádku	

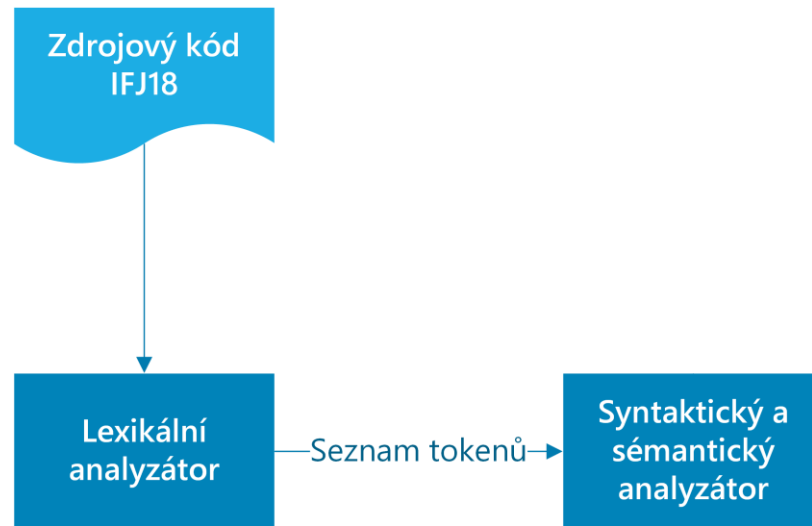


Příklad provázání tokenů – „a+3.5“

Struktura překladače



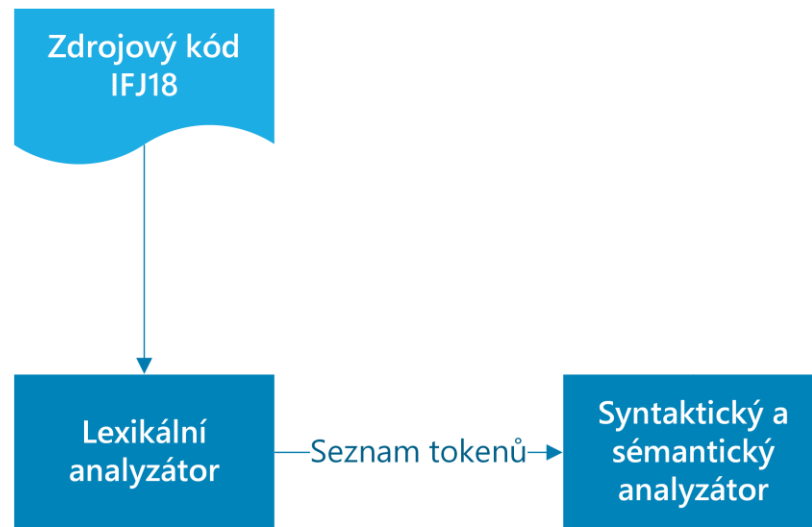
Struktura překladače



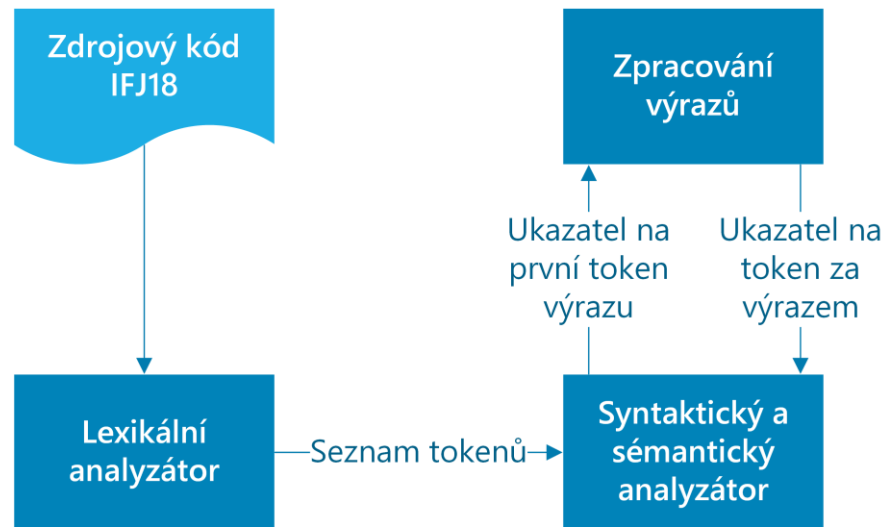
Syntaktická analýza

- Dvouprůchodová
 - Naplnění tabulky symbolů funkcemi
 - Samotná analýza kódu
- Rekurzivní sestup
- Derivace neterminálů na zásobníku
 - Precedenční LL(1) tabulka
- Současně sémantická analýza
 - Předpoklad, že kód neobsahuje syntaktické chyby
 - Syntaktická chyba má přednost
 - Ukládá ID do tabulky symbolů

Struktura překladače



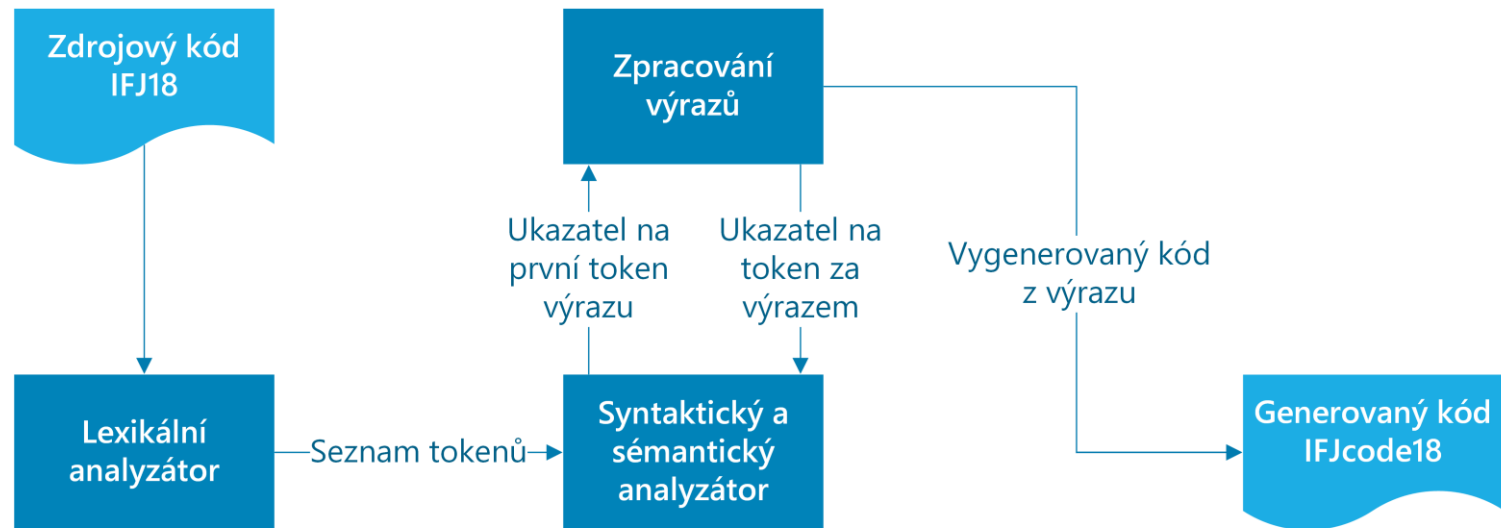
Struktura překladače



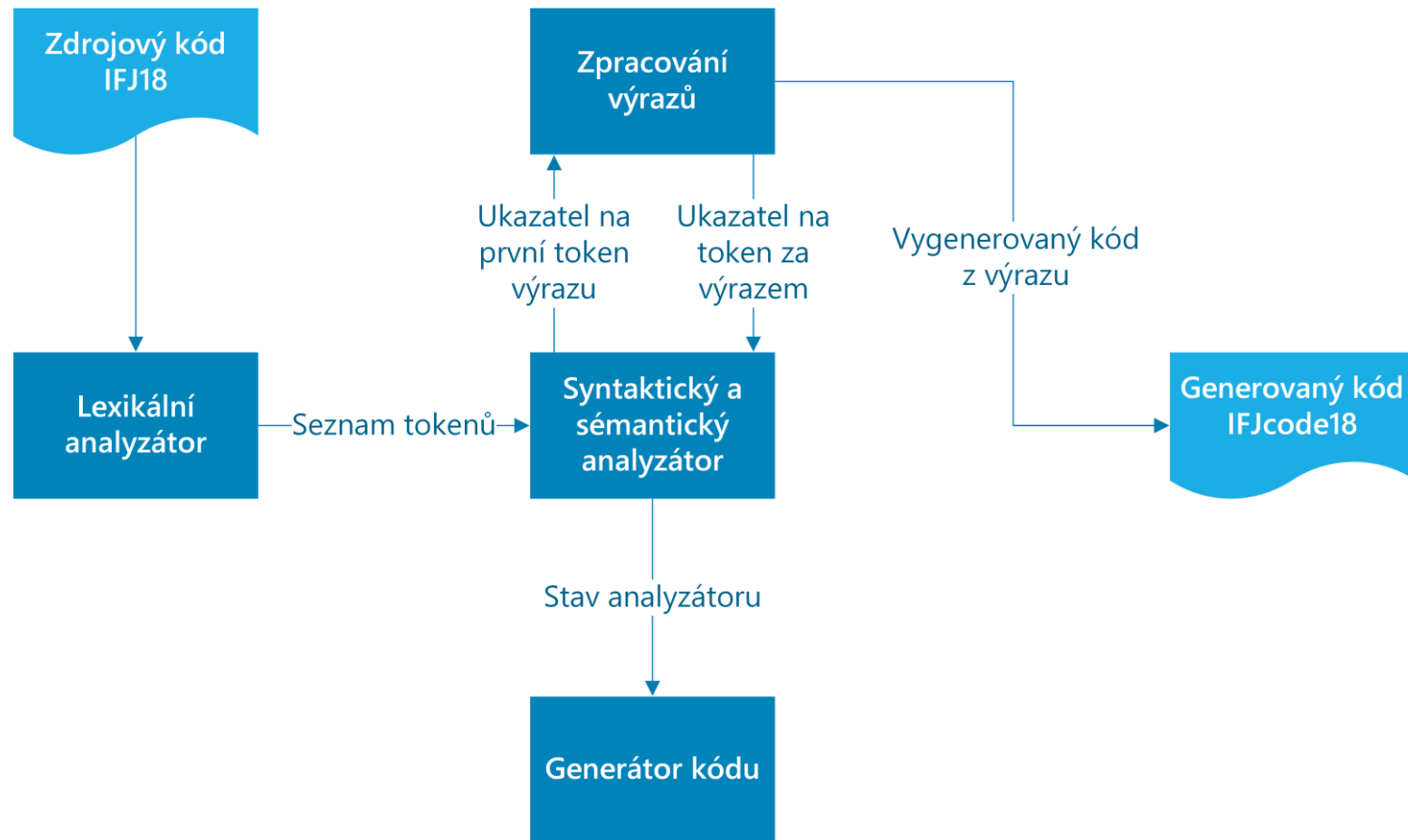
Zpracování výrazů

- Precedenční analýzou
 - Využití precedenční tabulky
- Syntaktická kontrola
- Sémantická kontrola
 - Známé typy operandů – kontrola typů
 - Jinak za běhu
- Generování kódu
 - Při převodu terminál → neterminál na zásobníku
 - Využití zásobníku IFJcode18 (*PUSHS*, *POPS*, ...)
- Syntaktickému analyzátoru se poté vrací ukazatel na první token nepatřící do daného výrazu

Struktura překladače



Struktura překladače



Generování kódu

- Probíhá zároveň se syntaktickou analýzou
 - Aktuálně načtený token
 - Token na vrcholu zásobníku
 - Lokální tabulku proměnných
- Předem generované funkce
 - Kontrola a převod typů
 - Základní funkce IFJ18
- Generace struktur funkčně odpovídajících strukturám vstupního kódu

Generování kódu

Problém definice funkce

Generace kódu zároveň s
analýzou

Tabulka proměnných není
kompletní

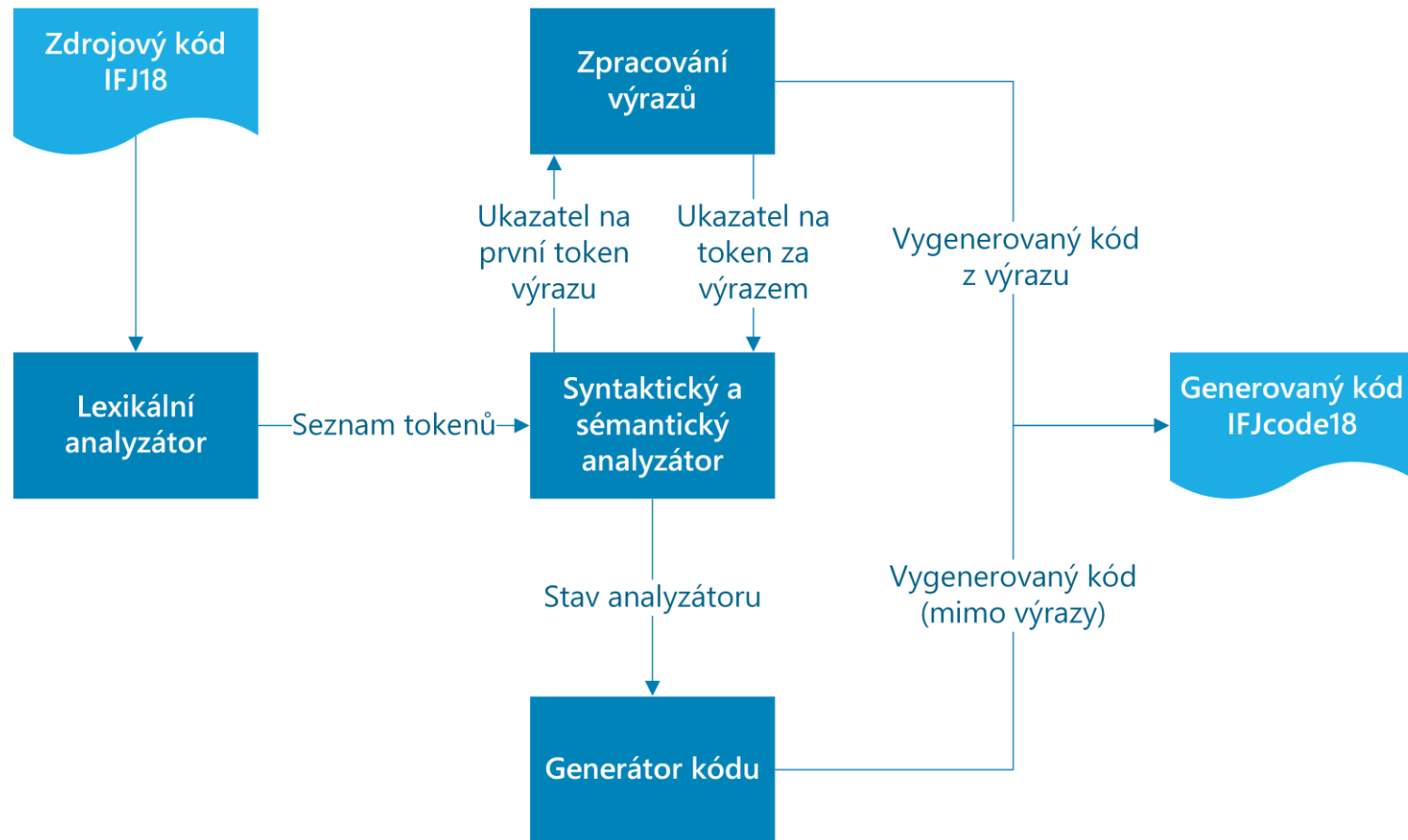
Řešení

Vstupní návěští funkce
generovat na konci

Po definici proměnných
se přesunout na začátek
těla funkce

```
... hlavní tělo programu ...  
JUMP fn$end  
LABEL fn$start  
... vnitřní kód funkce ...  
MOVE LF@$return TF@$return  
RETURN  
LABEL fn ← začátek funkce  
PUSHFRAME  
DEFVAR LF@a  
DEFVAR LF@$return  
MOVE LF@$return nil@nil  
MOVE LF@a LF@%1  
JUMP fn$start  
LABEL fn$end  
... hlavní tělo programu ...
```

Struktura překladače



Děkujeme za pozornost