

Implementace překladače jazyka IFJ22

Tým xstrel03
Varianta TRP

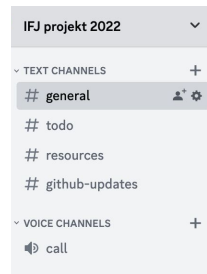


Matyáš Strelec	xstrel03
Ondřej Seidl	xseidl06
Maxmilián Nový	xnovym00
Dominik Klon	xklond00

Týmová spolupráce

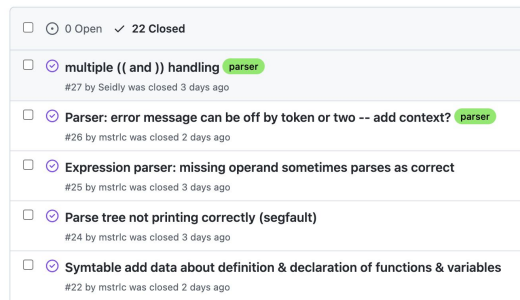


- platforma **Discord**
- pravidelné **schůzky a hovory**
- verzovací systém **Git a GitHub**

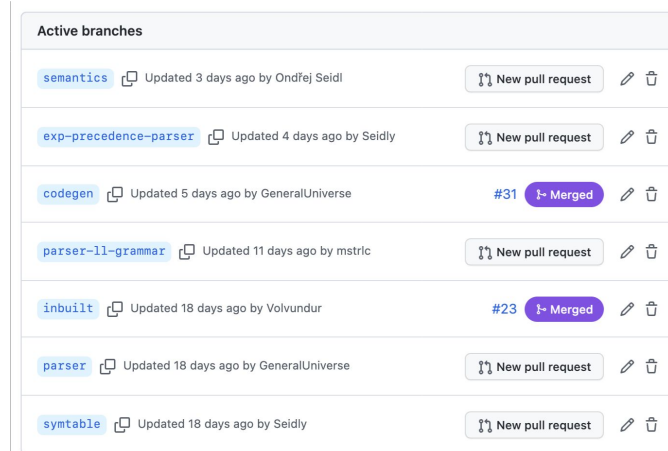


Discord

- sledování problémů
- rozdělení úkolů
- větve pro části
- možnost vrácení změn

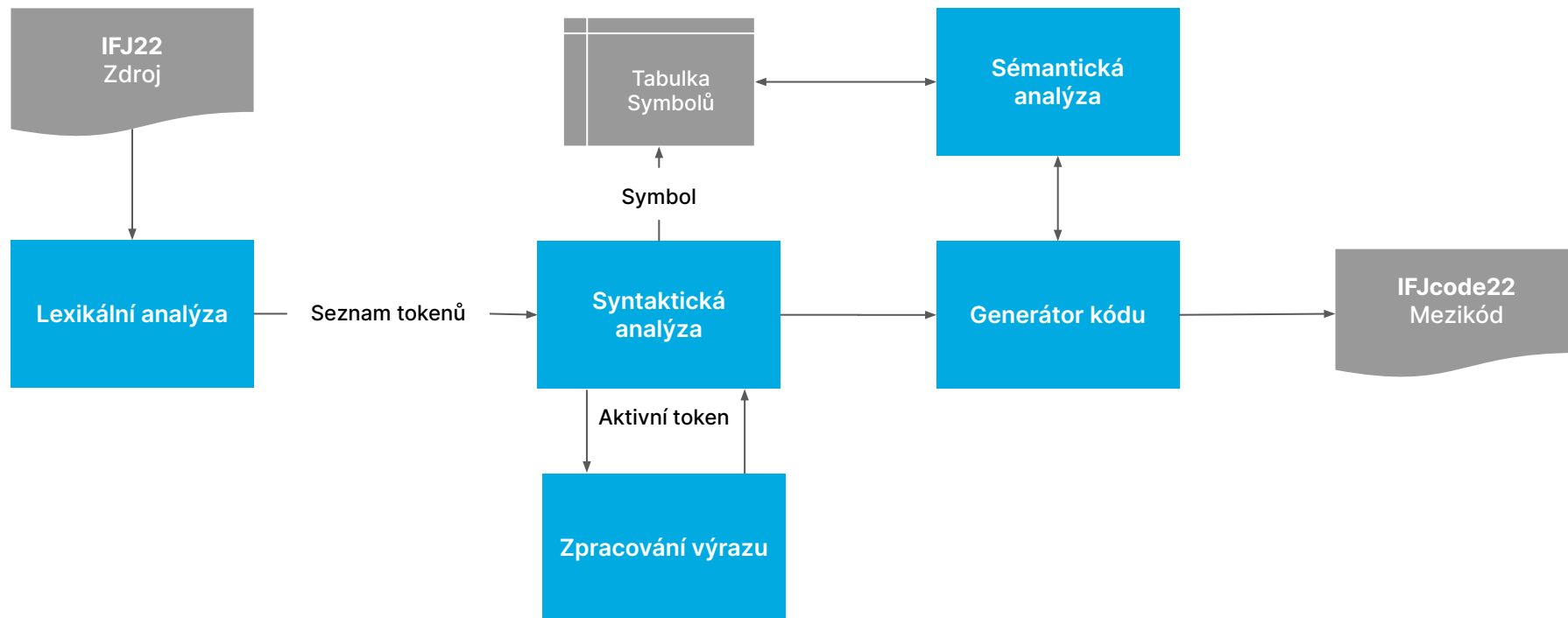


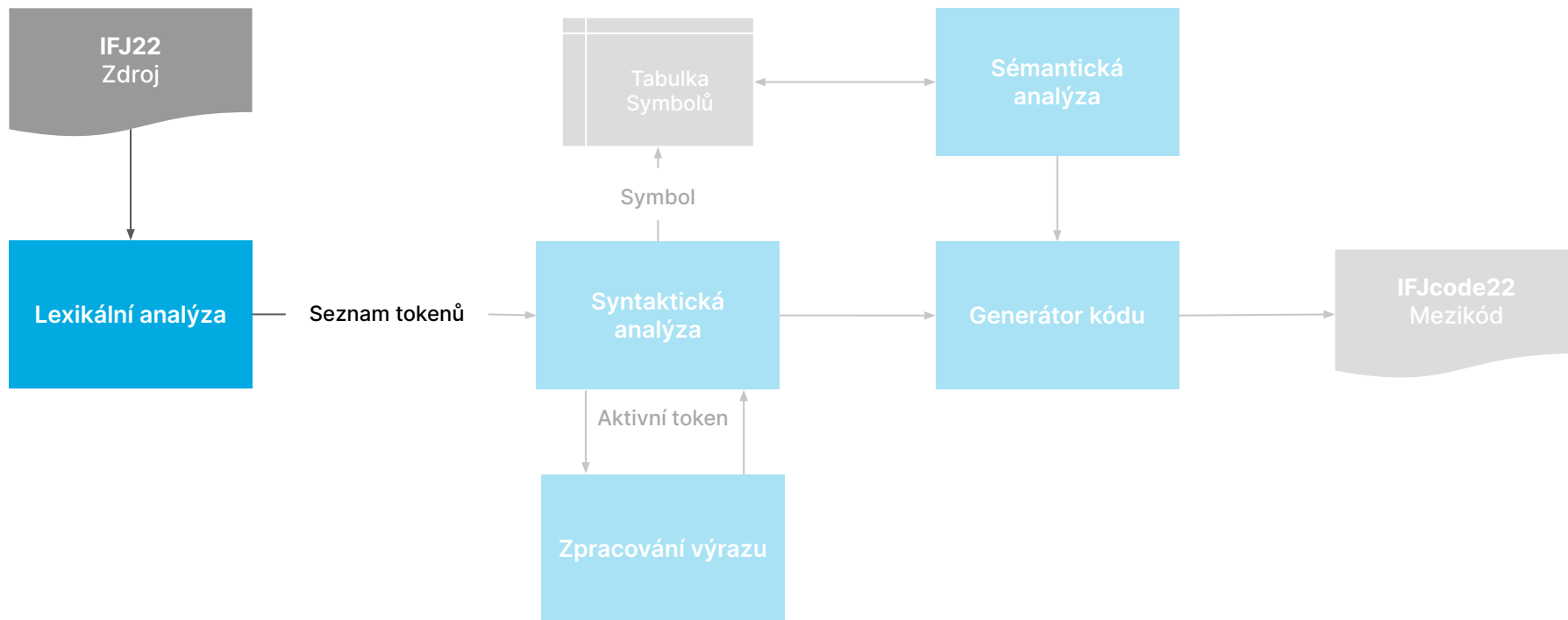
GitHub Issues



GitHub větve

Koncept fungování překladače





- **konečný automat**
 - rozpoznání typu tokenu
- naplnění **obousměrně vázaného seznamu**
 - datová struktura **token_list_t**
- rozpoznávání **datového typu, klíčových slov**
- zpracování **dat tokenu**
 - odstranění uvozovek z řetězce
 - změna escape sekvence na znaky

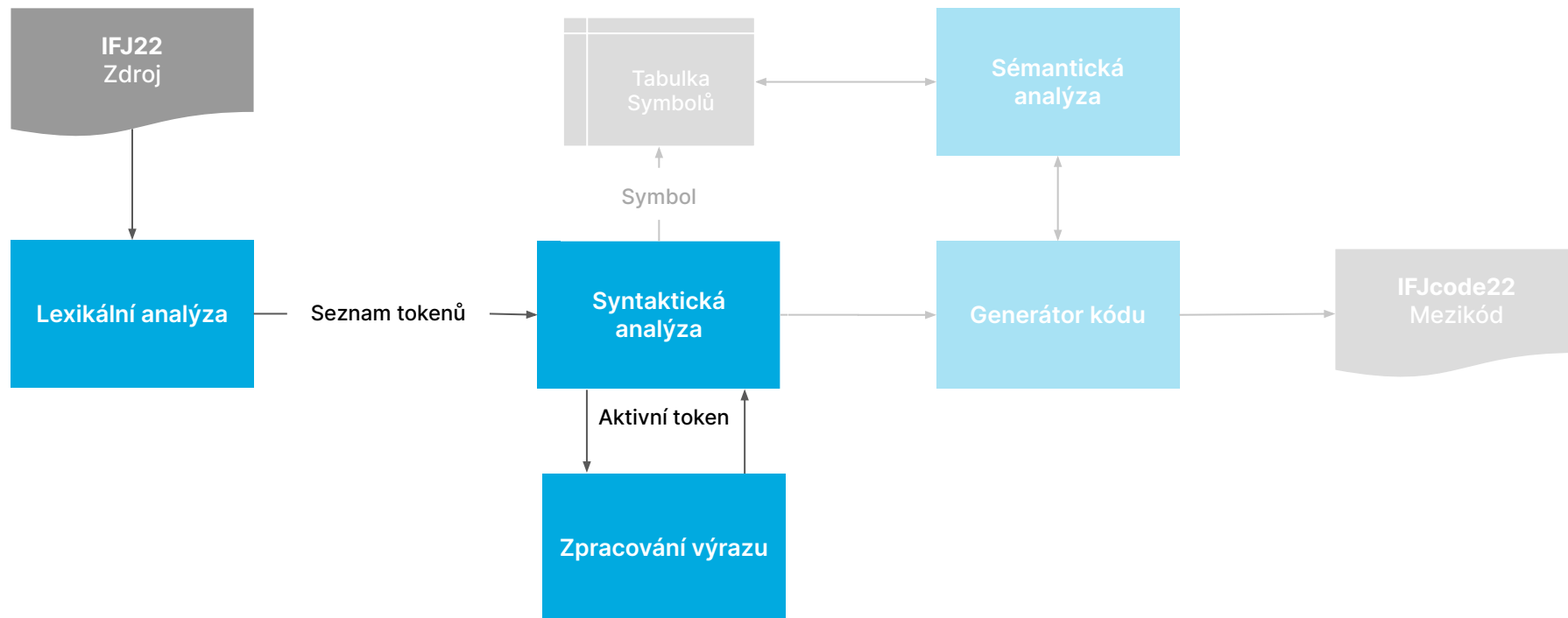
struktura seznamu tokenů

ukazatel na strukturu tokenu	firstToken
	activeToken
	lastToken

struktura tokenu

výčtový typ tokenu	type
ukazatel na řetězec	data
celé číslo	line
	length
ukazatel na strukturu tokenu	prev
	next

Syntaktická analýza a zpracování výrazu

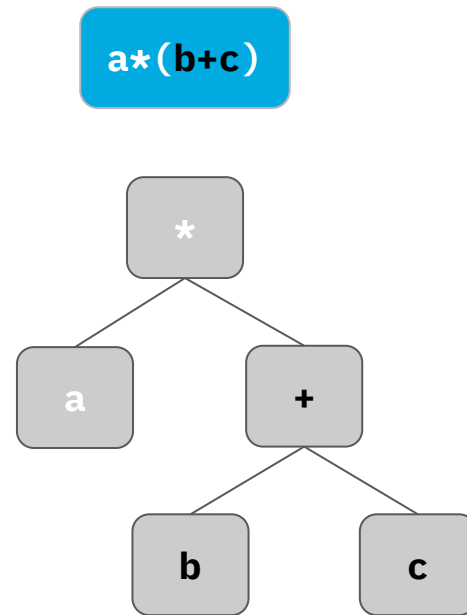


- kontrola **prologu**
- dle **LL-gramatiky** a **LL-tabulky**
- rekurzivní zpracování tokenů
 - **neterminály:** `rule_Stat()`, `rule_Expr()`, ...
 - **terminály:** `parseTerminal()`
- neočekávaná posloupnost → **chyba**

```
<prog> -> <stat> <prog>
...
<stat> -> $id = <assign> ;
<stat> -> return <expr> ;
...
<assign> -> <expr>
<assign> -> func-id ( <args> )
...
```

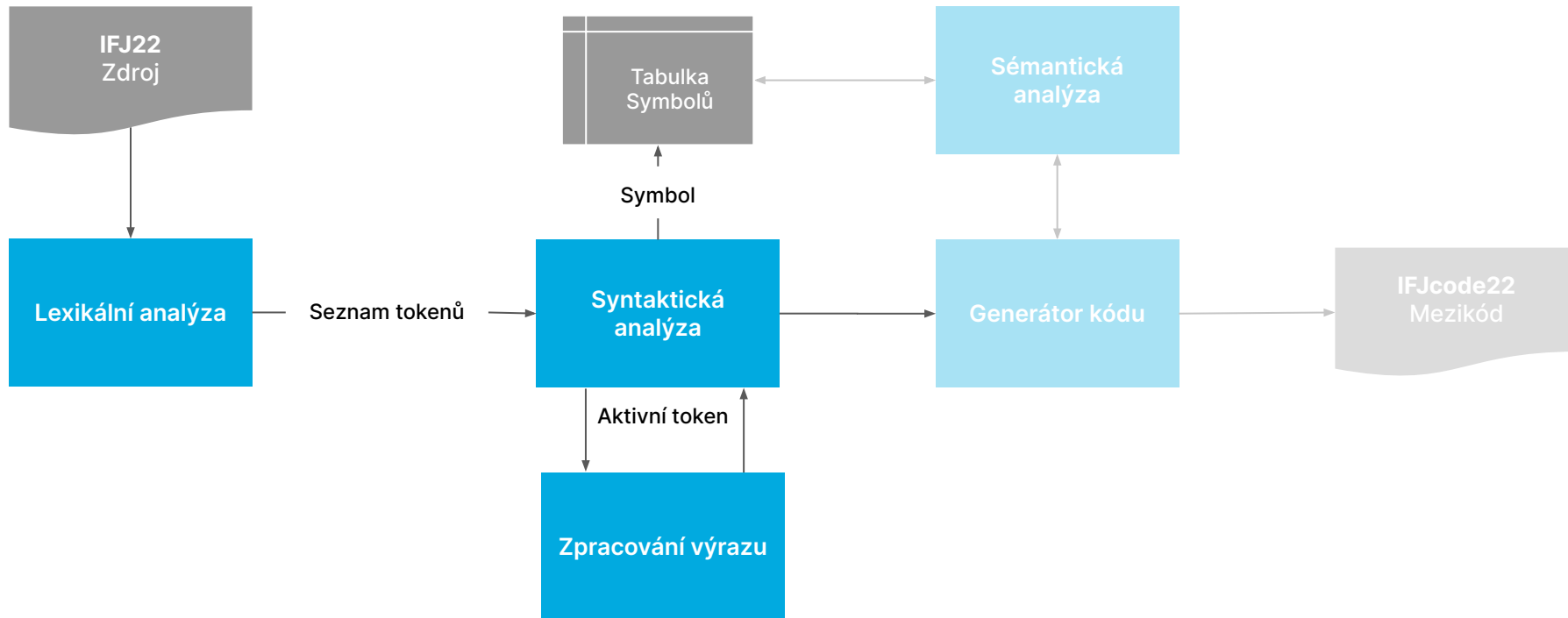
ukázka pravidel gramatiky

- voláno syntaktickou analýzou
- založené na precedenční tabulce
- **rekurzivní**
- výstupem je **syntaktický** strom výrazu
 - myšlenka jednoduchých stromů



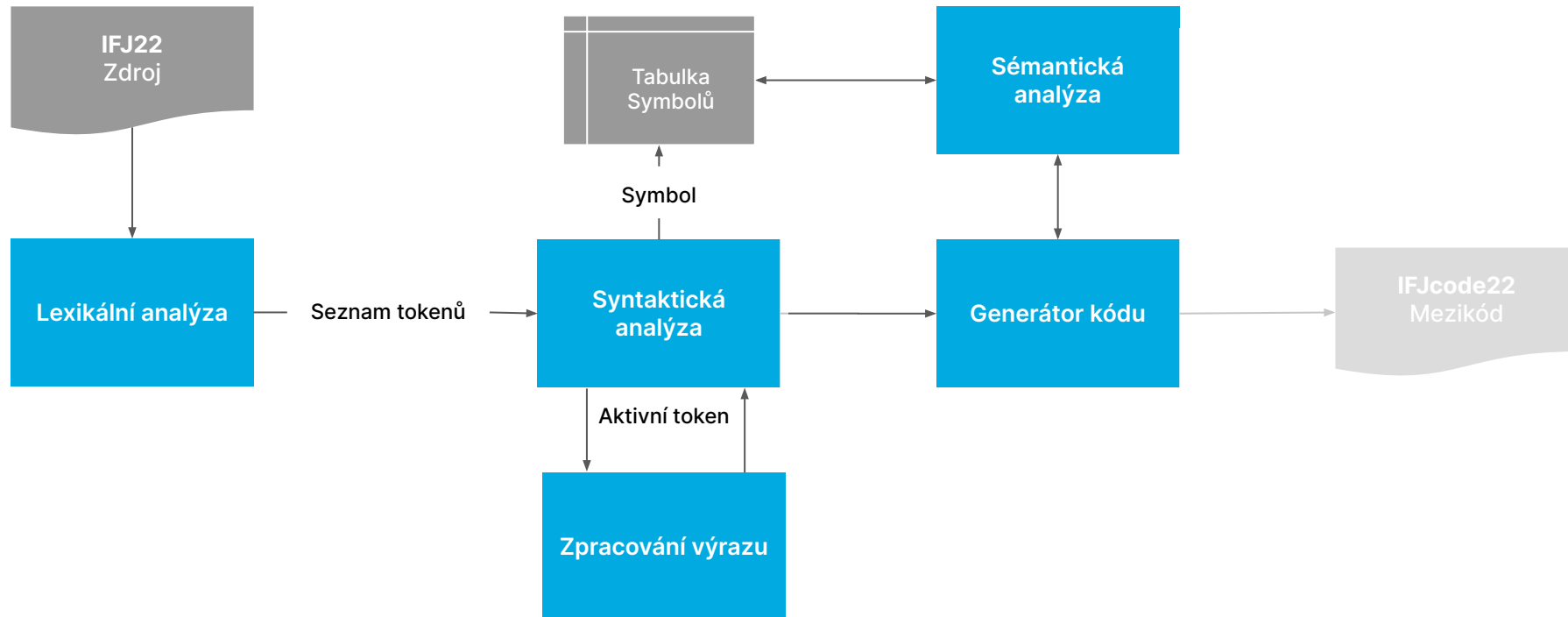
syntaktický strom pro výraz

Tabulka symbolů



- implementovaná jako soubor tabulek s rozptýlenými prvky
- obsahuje
 - tabulky proměnných pro jednotlivé rámce
 - tabulku funkcí
- jednotlivé tabulky jsou **dynamicky** alokované
- používána v sémantické analýze a při generování definic proměnných

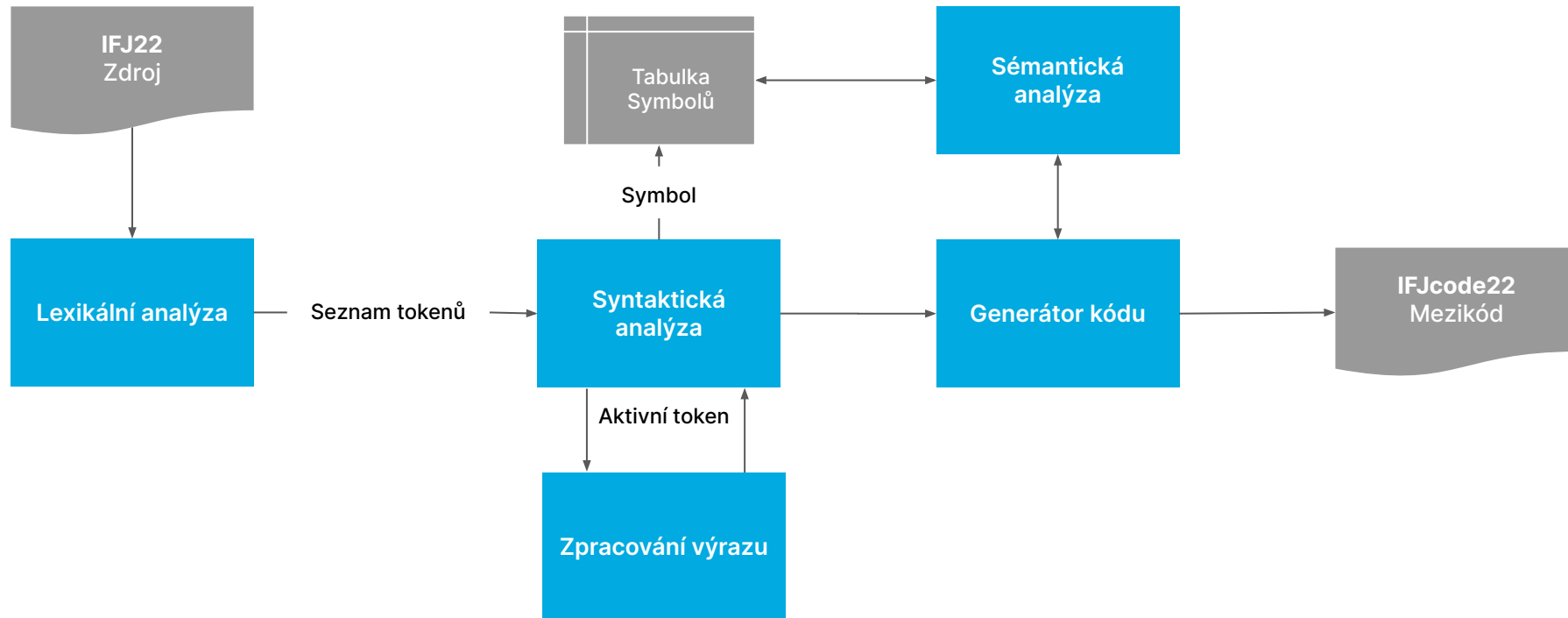
Sémantická analýza a generování kódu



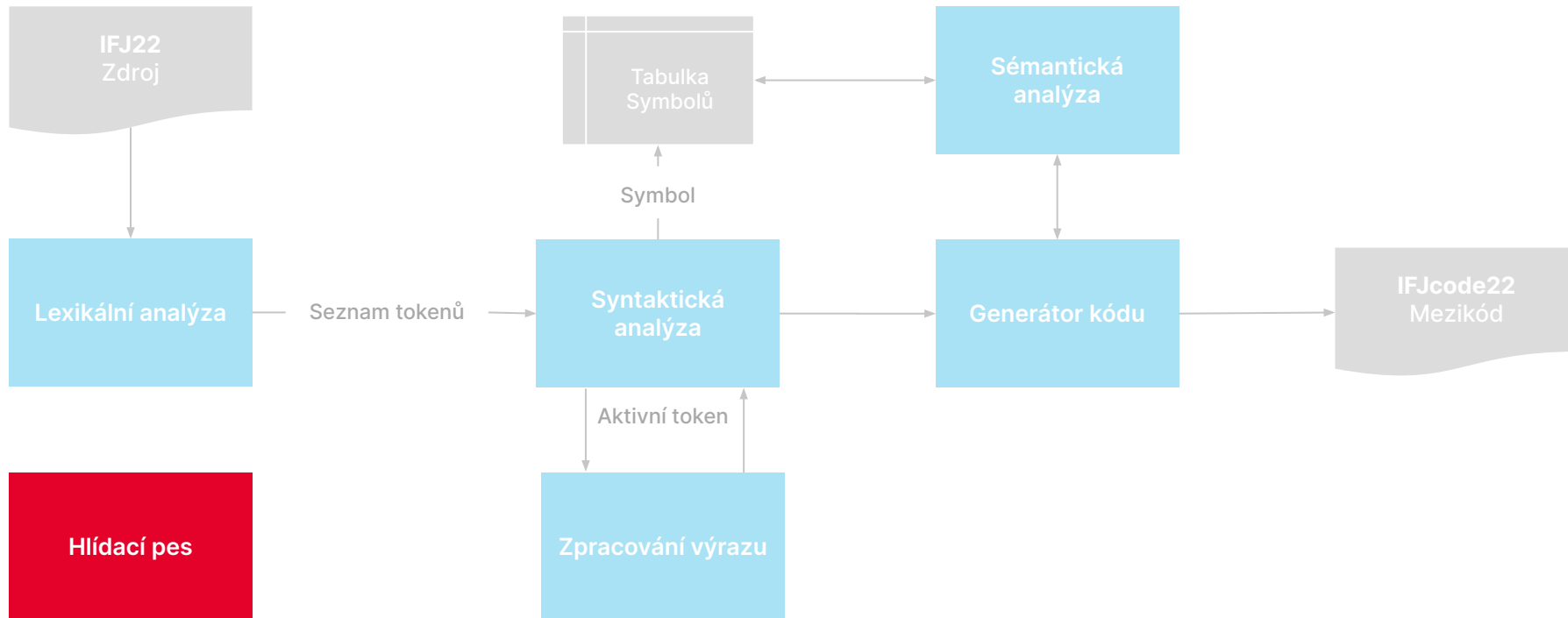
- kontroluje
 - existenci proměnných a funkcí
 - redefinice funkcí
 - počty argumentů, návratový typ funkce
 - typu operandů ve výrazech
- vyhledávání v **tabulce symbolů**, **generování kódu** v IFJcode22
- ukončení programu a **vrácení chybového kódu** v případě chyby

- napříč celou **syntaktickou analýzou**
- generovaný kód **konvertuje typy** aritmetických i logických **operací**
- řešení **rozsahu platnosti** proměnných
 - pomocí generování **jejich definice**
na začátku hlavního těla programu a každé funkce
- **předání parametrů** funkcím pomocí **zásobníků**

Koncept fungování překladače



Koncept fungování překladače



- kontrola chyb ve všech modulech
- makro **HANDLE_ERROR**
 - proměnná `error` ukládá návrat funkce
 - pokud je nalezena chyba volá `error_exit()`
- funkce **`error_exit()`**
 - volání **`exit()`** s kódem chyby
 - **chybový výpis**
 - > **Error 2 (Syntax error) on line 28 near '\$var**

Děkujeme za pozornost!

Tým xstrel03
Varianta TRP



Matyáš Strelec	xstrel03
Ondřej Seidl	xseidl06
Maxmilián Nový	xnovym00
Dominik Klon	xklond00

Děkujeme za pozornost!

