

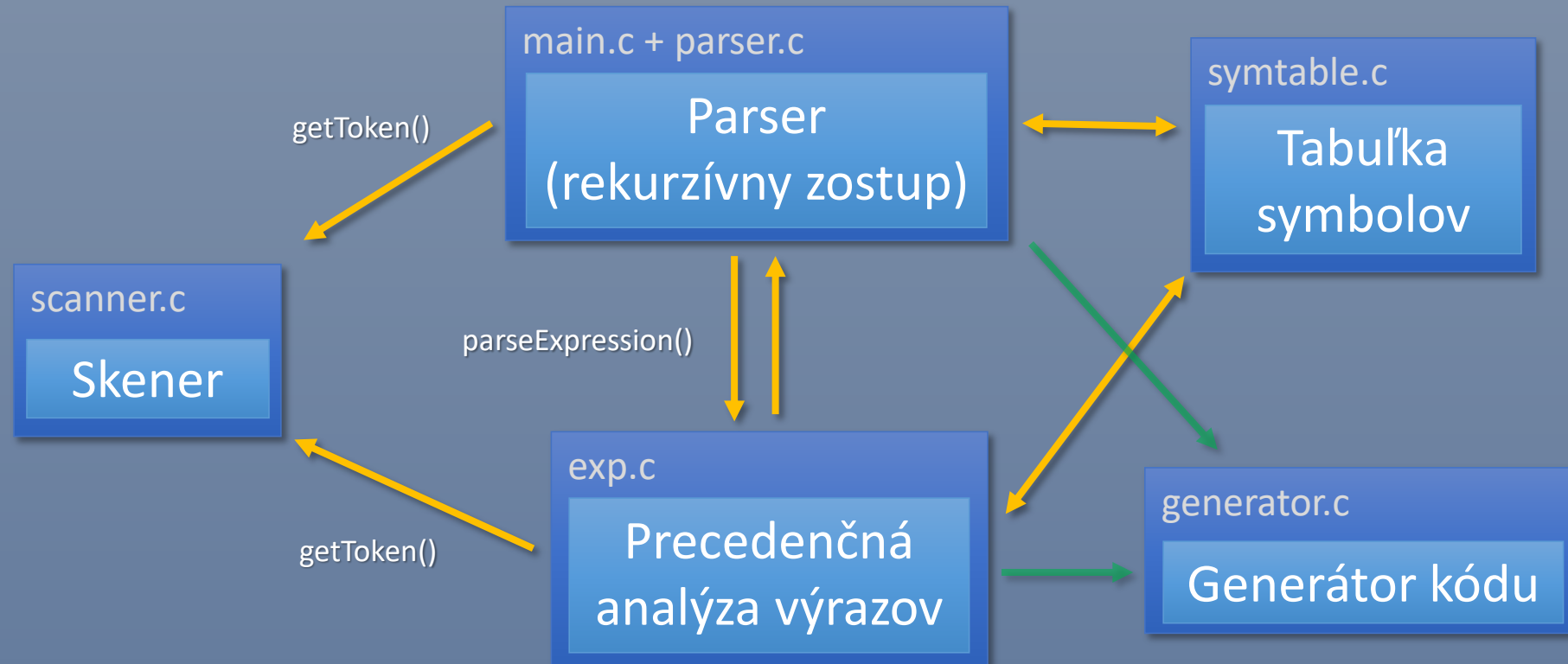
Implementace překladače imperativního jazyka IFJ23

tým xkruli03

varianta TRP-izp

Boris Hatala (xhatal02)
František Holář (xholan13)
Michal Krulich (xkruli03)
Stanislav Letaši (xletas00)

Štruktúra prekladača



Lexikálna analýza

- Získava nové tokeny z stdin
- Konečný automat
- getToken()
- storeToken()



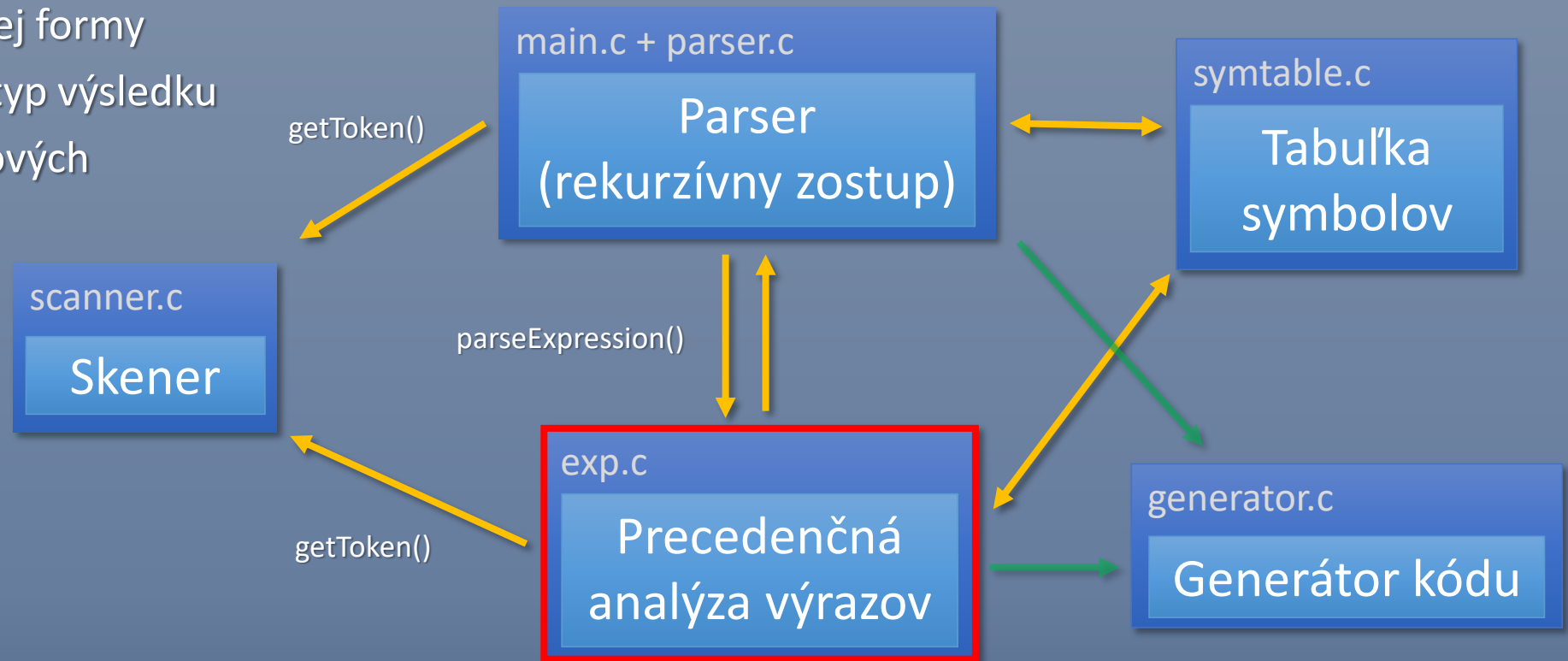
Parser (rekurzívny zostup)

- Hlavné riadenie prekladača
- LL-gramatika
- Syntaktická aj sémantická analýza
- Syntaxou riadený preklad



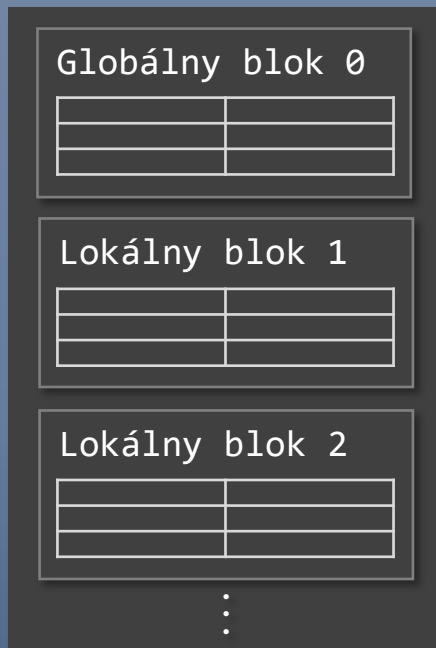
Precedenčná analýza výrazov

- Volaná parserom
- Kontrola a preklad výrazov
- Konverzia do postfixovej formy
- Vracia parseru dátový typ výsledku
- Generovanie zásobníkových inštrukcií



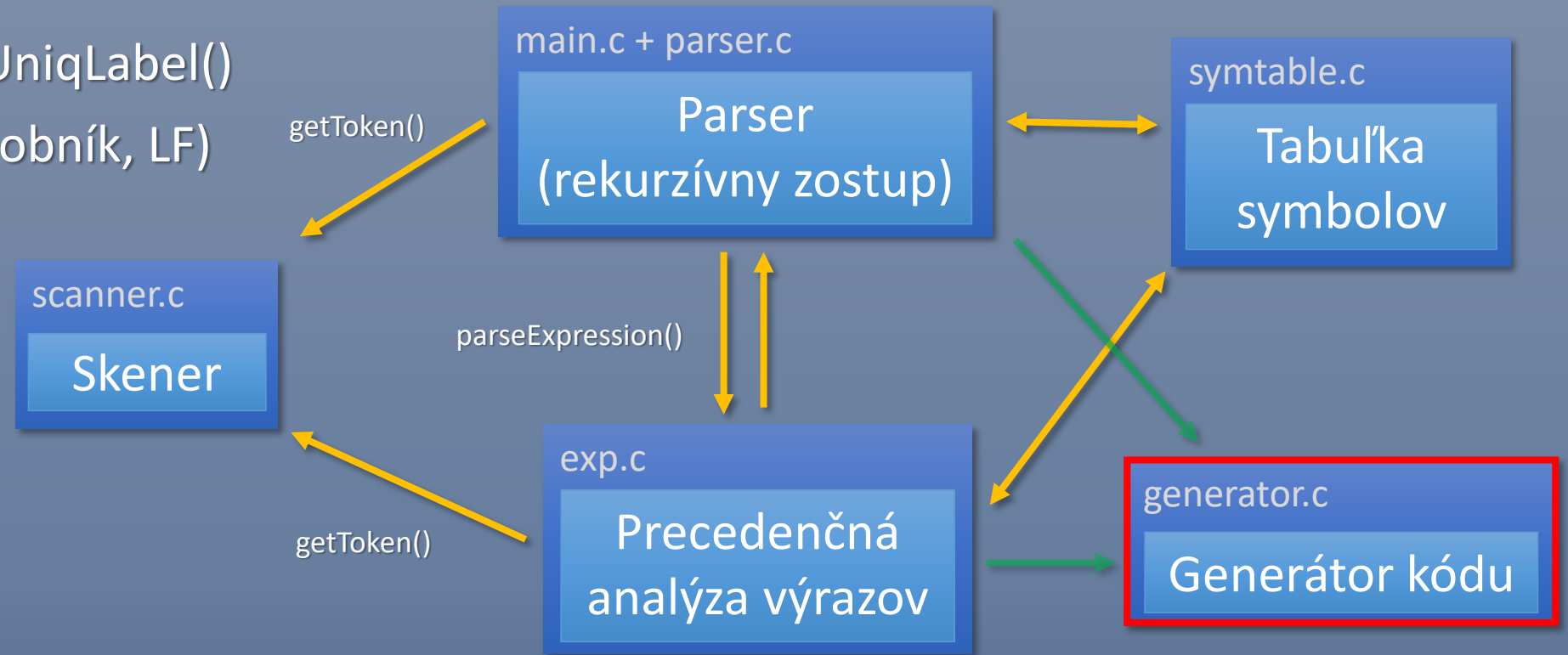
Tabuľka symbolov

- Varianta TRP-izp
- Dve rozptyľovacie funkcie
- ID premenných, dát. typy, ...
- Predpisy funkcií
- Zoznam dielčích blokov



Generovanie kódu

- Pomocou rozhrania generator.h
- Dva zoznamy
- `genUniqVar()`, `genUniqLabel()`
- Volanie funkcií (zásobník, LF)



Pomocné dátové štruktúry

StrR

- Reťazec s dynamickou dĺžkou
- Atribúty tokenov, inštrukcie, ...

```
typedef struct stringR {  
    char *data; ///  
    size_t size; ///  
} str_T;
```

DLL – dvojsmerný zoznam

- Uchováva reťazce
- Vygenerované inštrukcie, ...

```
typedef struct DLLstr_element {  
    char* string;                ///  
    struct DLLstr_element* prev; ///  
    struct DLLstr_element* next; ///  
} *DLLstr_el_ptr;  
  
typedef struct DLL_str_struct {  
    DLLstr_el_ptr first;        ///  
    DLLstr_el_ptr active;       ///  
    DLLstr_el_ptr last;         ///  
} DLLstr_T;
```