

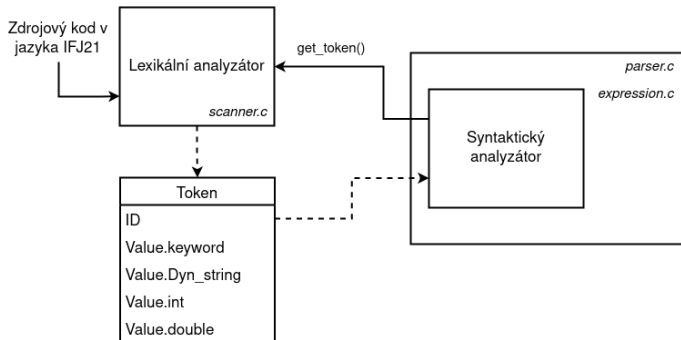
Implementace překladače imperativního jazyka IFJ21

Bártů Tomáš, Janeček Vít,
Pham Tony, Vacek Šimon



15. prosince 2021

- Lexikální analýza: `scanner.c`
- Syntaktická analýza: `parser.c`
- Syntaktická analýza: `expression.c`
- Tabulka symbolů: `symtable.c`
- Sémantická analýza: `parser.c` a `expression.c`
- Generování kódu: `generate_code.c` a `builtin.c`



- KA implementovaný přepínači
- Tokeny na vyžádání syntaktické analýzy
- Načtená data v attributech tokenu
- Řetězce uložené v **dynamickém stringu**

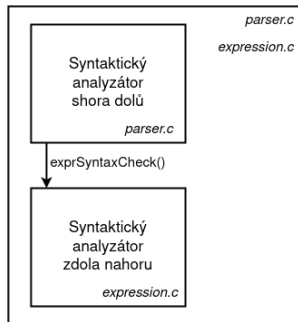
- Rekurzivní sestup
- Funkce vrací hodnotu odpovídající chybě
- Uložení pozice čtení před rozhodovacím tokenem
- Předání řízení analýze zdola nahoru u výrazů

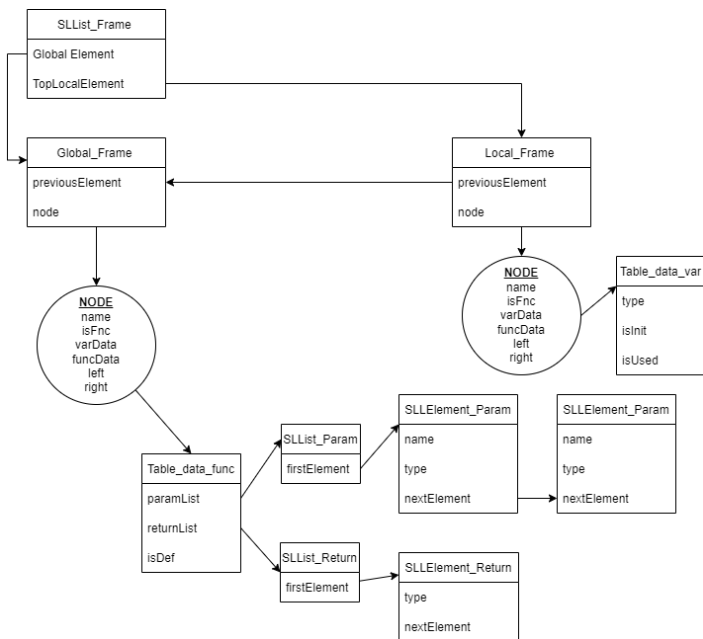
```
pos_t pos = fgetpos(srcFile)
```

načtení tokenu a rozhodnutí

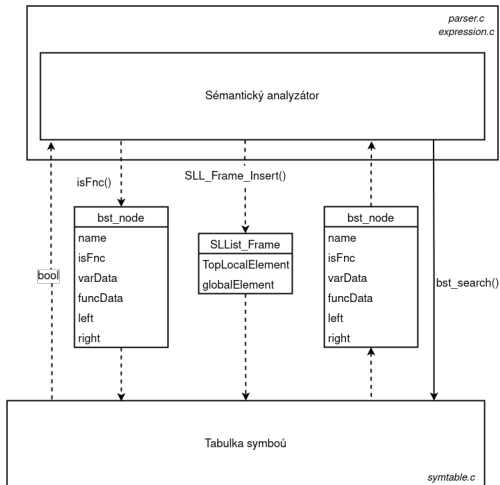
```
fsetpos(pos)
```

- Precedenční přístup
- Zásobník terminálů a neterminálů
- Precedenční tabulka implementovaná 2D polem
- Výraz se vyhodnotí při přijetí neočekávaného tokenu



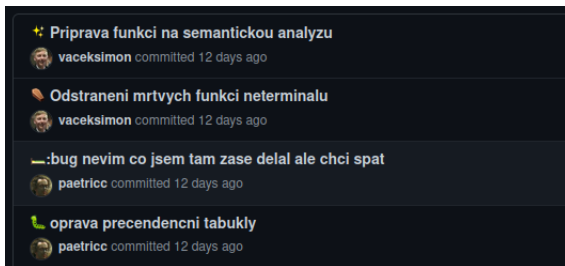


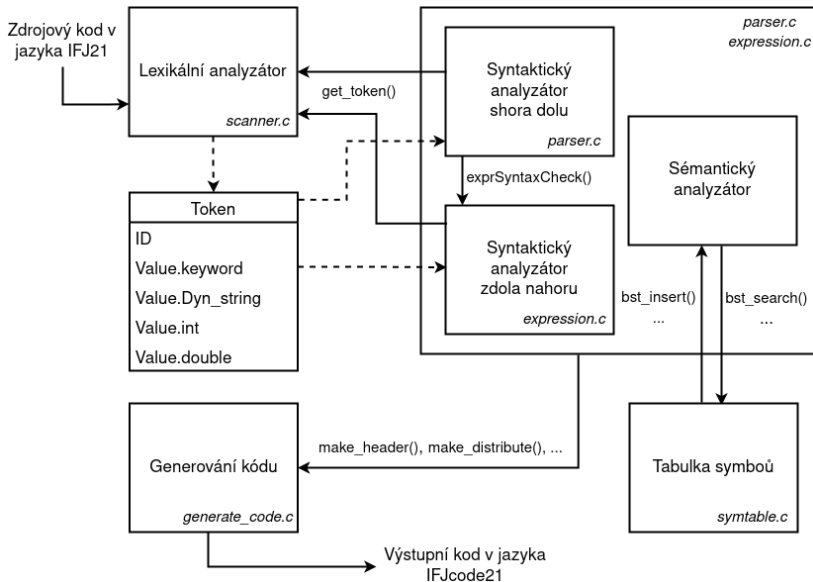
- Doplněny k syntaktické analýze
- Zásobník na datové typy operandů



- Vestavěné funkce v `builtin.c`
- Výpis instrukcí ve funkcích v `generate_code.c`
- Stínění řešeno kopírováním proměnných
- Změna této proměnné se vrátí do nižšího rámce

- Komunikace přes messenger a discord
- Github pro hostování repozitáře
- Systém forků a pullrequestů
- Prezenční schůzky na fakultě





| | require | global | id_fce | function | return | end | (|) | integer | number | string | id_var | , | : | int_value | num_value | str_value | nil | local | = | # | if | else | while | \$ |
|----------------|---------|--------|--------|----------|--------|-----|-----|-----|---------|--------|--------|--------|-----|-----|-----------|-----------|-----------|-----|-------|-----|-----|-----|------|-------|-----|
| <start> | 1. | | | | | | | | | | | | | | | | | | | | | | | | |
| <program> | | 2. | 3. | 4. | | | | | | | | | | | | | | | | | | | | | 5. |
| <fnc_dec> | | 6. | | | | | | | | | | | | | | | | | | | | | | | |
| <params_dec> | | | | | | | | 7. | 8. | 8. | 8. | | | | | | | | | | | | | | |
| <params_dec2> | | | | | | | | 9. | | | | | 10. | | | | | | | | | | | | |
| <return_type> | | 12. | 12. | 12. | | | | | | | | | | 11. | | | | | | | | | | | 13. |
| <data_type> | | | | | | | | | 13. | 14. | 15. | | | | | | | 16. | | | | | | | |
| <fnc_call> | | | 17. | | | | | | | | | | | | | | | | | | | | | | |
| <value> | | | | | | | | 18. | | | | 19. | | | 19. | 19. | 19. | 19. | | | | | | | |
| <value2> | | | | | | | | 21. | | | | | 20. | | | | | | | | | | | | |
| <value_last> | | | | | | | | | | | | 22. | | | 23. | 24. | 25. | 26. | | | | | | | |
| <fnc_def> | | | | 27. | | | | | | | | | | | | | | | | | | | | | |
| <fnc_head> | | | | 28. | | | | | | | | | | | | | | | | | | | | | |
| <fnc_def2> | | | 29. | | 29. | 29. | | | | | | 29. | | 30. | | | | | 29. | | | 29. | | 29. | |
| <params_def> | | | | | | | | 31. | | | | 32. | | | | | | | | | | | | | |
| <params_def2> | | | | | | | | 33. | | | | | 34. | | | | | | | | | | | | |
| <var_def> | | | | | | | | | | | | 35. | | | | | | | | | | | | | |
| <return> | | | | | 36. | | | | | | | | | | | | | | | | | | | | |
| <return2> | | | | | | 37. | 38. | | | | | 38. | | | 38. | 38. | 38. | 38. | | | 38. | | | | |
| <fnc_body> | | | 39. | | 40. | 41. | | | | | | 39. | | | | | | | 39. | | | 39. | | 39. | |
| <statements> | | | 44. | | | | | | | | | 43. | | | | | | | 42. | | | 45. | | 46. | |
| <var_dec> | | | | | | | | | | | | | | | | | | | 47. | | | | | | |
| <var_dec_init> | | | 49. | | 49. | 49. | | | | | | 49. | | | | | | | | | | 49. | 49. | 49. | |
| <var_assign> | | | 51. | | | | 50. | | | | | 50. | | | 50. | 50. | 50. | 50. | | 48. | | 50. | | | |
| <if> | | | | | | | | | | | | | | | | | | | | | | 52. | | | |
| <loop> | | | | | | | | | | | | | | | | | | | | | | | | 53. | |
| <statements> | | | 55. | | 56. | 54. | | | | | | 55. | | | | | | | 55. | | | 55. | 54. | 55. | |

