



Dokumentace k semestrální práci z KIV/FJP

# Překladač jazyka Pascal0Like

**Student:** Martin Kružej, Jakub Šmaus  
**St. číslo:** A17N0079P, A17N0089P  
**E-mail:** kruzej@students.zcu.cz, smaus@students.zcu.cz  
**Datum:** TODO

# Obsah

<b>1</b>	<b>Zadání</b>	<b>1</b>
<b>2</b>	<b>Analýza</b>	<b>2</b>
<b>3</b>	<b>Implementace</b>	<b>3</b>
3.1	Překladač . . . . .	3
3.2	Interpret . . . . .	3
3.3	TABULKY PROZATIM . . . . .	3
<b>4</b>	<b>Uživatelská dokumentace</b>	<b>8</b>
4.1	Postup přeložení a sestavení . . . . .	8
<b>5</b>	<b>Závěr</b>	<b>9</b>

# 1 Zadání

Tématem semestrální práce je vytvoření

## 2 Analýza

## 3 Implementace

### 3.1 Překladač

### 3.2 Interpret

### 3.3 TABULKY PROZATIM

Tabulka 3.1: Instrukce

Kód	Instrukce	Popis
1	LIT 0, M	Vloží konstantní celou hodnotu ( <b>literál</b> ) M do zásobníku
2	LRT 0, M	Vloží konstantní reálnou hodnotu ( <b>literál</b> ) M do zásobníku
3	OPR 0, M	<b>Operace</b> , která se provede nad vrcholem zásobníku <b>pro celé čísla</b>
	OPR 0, 1	<b>Negace</b> ; vybere vrchol a vrátí negativní hodnotu
	OPR 0, 2	<b>Sčítání</b> ; vybere dvě hodnoty, sečte a vrátí
	OPR 0, 3	<b>Odečítání</b> ; vybere dvě hodnoty, odečte druhou první a vrátí výsledek
	OPR 0, 4	<b>Násobení</b> ; vybere dvě hodnoty, vynásobí a vrátí výsledek
	OPR 0, 5	<b>Dělení</b> ; vybere dvě hodnoty, vydělí druhou první
	OPR 0, 6	<b>Lichost</b> ; vybere vrchol a vloží 1 když liché, 0 když sudé
	OPR 0, 7	<b>Modulo</b> ; vybere dvě hodnoty, vydělí druhý prvním a vloží zbytek
	OPR 0, 8	<b>Rovnost</b> ; vybere dvě hodnoty, a vloží 1 pokud se rovnají, jinak 0
	OPR 0, 9	<b>Nerovnost</b> ; vybere dvě hodnoty a vloží 0 pokud se rovnají, jinak 0

	OPR 0, 10	<b>Menší než;</b> vybere dvě hodnoty a vloží 1 pokud je první menší než druhá, jinak 0
	OPR 0, 11	<b>Větší nebo rovno než;</b> vybere dvě hodnoty a vloží 1 pokud je první větší nebo rovno než druhá, jinak 0
	OPR 0, 12	<b>Větší;</b> vybere dvě hodnoty a vloží 1 pokud je první větší nebo rovno než druhá, jinak 0
	OPR 0, 13	<b>Menší nebo rovno než;</b> vybere dvě hodnoty a vloží 1 pokud je první menší nebo rovno než druhá, jinak 0
4	LOD L, M	<b>Načtení;</b> načte hodnotu vrcholu z umístění dané offse-tem M od L lexikografických úrovní dolů
5	STO L, M	<b>Uložení;</b> uloží hodnotu vrcholu z umístění dané offse-tem M od L lexikografických úrovní dolů
6	CAL L, M	<b>Volání procedury</b> v kódovém indexu <b>M</b>
7	RET 0, 0	<b>Návrat z procedury;</b> vrátí se z procedury do volající procedury
8	INT 0, M	<b>Alokování</b> místa pro M hodnot na vrcholu zásobníku
9	JMP 0, M	Provede skok do instrukce <b>M</b>
10	JMC 0, M	Vybere vrchol a skočí k instrukci M pokud je rovna 0, <b>podmíněný skok</b>
11	REA L, M	<b>Načte celé číslo</b> ze vstupu a uloží jej na zásobník
12	WRI L, M	Odebere celé číslo z vrcholu zásobníku a <b>vypíše jej na vstup</b>
13	RER L, M	<b>Načte reálné číslo</b> ze vstupu a uloží jej na zásobník
14	WRR L, M	Odebere reálné číslo z vrcholu zásobníku a <b>vypíše jej na vstup</b>
15	OPF 0, M	<b>Operace,</b> která se provede nad vrcholem zásobníku <b>pro reálná čísla</b>
	OPF 0, 1	<b>Negace;</b> vybere vrchol a vrátí negativní hodnotu
	OPF 0, 2	<b>Sčítání;</b> vybere dvě hodnoty, sečte a vrátí
	OPF 0, 3	<b>Odečítání;</b> vybere dvě hodnoty, odečte druhou první a vrátí výsledek

	OPF 0, 4	<b>Násobení</b> ; vybere dvě hodnoty, vynásobí a vrátí výsledek
	OPF 0, 5	<b>Dělení</b> ; vybere dvě hodnoty, vydělí druhou první
	OPF 0, 6	<b>Lichost</b> ; vybere vrchol a vloží 1 když liché, 0 když sudé
	OPF 0, 7	<b>Modulo</b> ; vybere dvě hodnoty, vydělí druhý prvním a vloží zbytek
	OPF 0, 8	<b>Rovnost</b> ; vybere dvě hodnoty, a vloží 1 pokud se rovnají, jinak 0
	OPF 0, 9	<b>Nerovnost</b> ; vybere dvě hodnoty a vloží 0 pokud se rovnají, jinak 0
	OPF 0, 10	<b>Menší než</b> ; vybere dvě hodnoty a vloží 1 pokud je první menší než druhá, jinak 0
	OPF 0, 11	<b>Větší nebo rovno než</b> ; vybere dvě hodnoty a vloží 1 pokud je první větší nebo rovno než druhá, jinak 0
	OPF 0, 12	<b>Větší</b> ; vybere dvě hodnoty a vloží 1 pokud je první větší nebo rovno než druhá, jinak 0
	OPF 0, 13	<b>Menší nebo rovno než</b> ; vybere dvě hodnoty a vloží 1 pokud je první menší nebo rovno než druhá, jinak 0
16	RTI 0, 0	<b>Reálné číslo na celé číslo</b> ; vybere jednu hodnotu ze zásobníku a vloží celou část čísla do zásobníku
17	ITR 0, 0	<b>Celé číslo na reálné číslo</b> ; vybere jednu hodnotu ze zásobníku a vloží číslo jako reálné do zásobníku
18	NEW 0, 0	<b>Alokace na haldě</b> ; alokuje se jedno místo na haldě, na zásobník vloží hodnotu představující pozici místa v haldě
19	DEL 0, 0	<b>Uvolnění místa na haldě</b> ; odebere ze zásobníku jednu hodnotu a to adresu na haldě, kterou uvolní
20	LDA 0, 0	<b>Načtení hodnoty z haldy</b> ; odebere ze zásobníku hodnotu a vloží hodnotu z haldy
21	STA 0, 0	<b>Uložení hodnoty na haldu</b> ; odebere dvě hodnoty zásobníku. Na první představující adresu uloží druhou v haldě

22	PLD 0, 0	<b>Dynamické načtení hodnoty z místa určeného L/A;</b> odebere ze zásobníku dvě hodnoty. První je úroveň zanoření a druhá je relativní pozice
23	PST 0, 0	<b>Dynamické uložení hodnoty z místa určeného L/A;</b> odebere ze zásobníku tři hodnoty. První je úroveň zanoření, druhá relativní pozice a třetí

Tabulka 3.2: Symboly

Symbol	Název	Popis
+	PLUS	sčítání ve výrazech
-	MINUS	odečítání ve výrazech
*	MULTIPLY	násobení ve výrazech
/	DIVIDE	dělení ve výrazech
%	MODULO	modulo ve výrazech
=	EQUAL	kontrola rovnosti dvou výrazů
<>	NOT_EQUAL	kontrola nerovnosti dvou výrazů
<	LT	kontrola, že levý výraz je menší než pravý výraz
<=	LE	kontrola, že levý výraz je menší nebo stejně velký jako pravý výraz
>=	GE	kontrola, že levý výraz je větší nebo stejně velký jako pravý výraz
>	GT	kontrola, že levý výraz je větší než pravý výraz
:=	ASSIGN	přiřazení proměnné
?	TERNARY_ONE	ternární operátor 1
!	TERNARY_TWO	ternární operátor 2
;	SEMI	konec příkazu
,	COMMA	oddělovač proměnných
:	COLON	inicializace návěští
(	LPAREN	levá závorka výrazu
)	RPAREN	pravá závorka výrazu
[	LBRACK	počátek výrazů paralelní deklarace
]	RBRACK	konec výrazů paralelní deklarace
.	DOT	konec programu

Tabulka 3.3: Klíčová slova

Název	Použití
-------	---------



BEGIN	počátek bloku příkazů
END	konec bloku příkazů

## 4 Uživatelská dokumentace

### 4.1 Postup přeložení a sestavení

## 5 Závěr

Výstupem práce je funkční  
Zadání bylo tímto splněno.