**Lexikální analýza**

První fází překladače je lexikální analyzátor (scanner). Nejdůležitější funkcí je zde *scannerGetToken*, což je vlastně implementace deterministického konečného automatu, který čte jednotlivé znaky ze standartního vstupu, dokud nedojde do koncového stavu (návratová hodnota 0), nebo nenarazí na lexikální chybu (návratová hodnota 1). To je zajištěné pomocí *switche* umístěného v nekonečném *while* cyklu, kde každý *case* reprezentuje právě jeden stav automatu, a při každém opakování cyklu je čten právě jeden znak ze vstupu.

Funkce má jediný parametr, a to ukazatel na předem alokovaný prvek typu *Token*, jehož typ (a případně atribut) se nastaví podle přečtených hodnot a dosaženého koncového stavu. Mezi typy tokenů bez atributů patří například relační operátory, aritmetické operátory, přiřazení, klíčová slova, odřádkování (EOL), konec souboru (EOF), závorky a další povolené znaky.

Typy tokenů s atributy jsou pouze identifikátor, celé číslo, desetinné číslo a řetězec. Atribut tokenu je typu *union*, v případě identifikátoru nebo řetězce se tedy využívá pouze atribut *string*, u celého čísla *integer* a u desetinného čísla *real.* Když je zřejmé že načítáme nějaký z těchto 4 typů tak se přečtené znaky nejprve ukládají do bufferu (globální pole), a ve chvíli kdy je dosaženo koncového stavu, se do následující pozice v bufferu vloží nulový znak. Obsah bufferu až po tento znak je poté převeden na celé/desetinné číslo, nebo zkopírován do nově alokovaného stringu (jehož velikost pro alokaci je index pozice prvního nulového znaku v bufferu + 1). Tento postup je velice praktický, protože i když se obsah bufferu nemaže ale pouze přepisuje, cokoli za prvním nulovým znakem je ignorováno, například pozůstatky nějakého delšího dříve načteného řetězce.

Všechny načtené tokeny se ukládají do obousměrného lineárního seznamu, který je předáván dalším částem překladače. Z pracujících se seznamem stojí za zmínku *ScannerGetTokenList*, která plní seznam tokeny (voláním dříve popsané funkce), dokud nenarazí na token typu EOF, nebo volaná funkce nenahlásí lexikální chybu.