

```

program >> globPromenne funAProc
globPromenne >> globPromenne modifikator typ IDENTIFIKATOR = matVyras; |
globPromenna modifikator typ IDENTIFIKATOR = logVyras | zadne
lokPromenne >> lokPromenne typ IDENTIFIKATOR = matVyras; |
lokPromenne typ IDENTIFIKATOR = logVyras; | zadne
modifikator >> konst | zadne
funAProc >> funAProc funkce | funAProc procedura | zadne
funkce >> funkce typ IDENTIFIKATOR(parametry) { vnitrekFunkce }
procedury >> procedura IDENTIFIKATOR(parametry) { vnitrekProcedury }
parametry >> parametry, parametr | parametr | zadne
parametr >> typ IDENTIFIKATOR
vnitrekFunkce >> lokPromenna viceAkci vracHodnoty
vnitrekProceury >> lokPromenna viceAkci
akce >> cyklus | rozhodnuti | vyras | volaniFunkce | vraceniHodnoty | zastaveni |
IDENTIFIKATOR = logVyras; | IDENTIFIKATOR = matVyras;
viceAkci >> viceAkci akce | zadne
cyklus >> zatimco(sloz_podminka) { viceAkci }
rozhodnuti >> pokud(sloz_podminka) { vice_akci } | pokud(sloz_podminka) { vice_akci }
pokud ne { vice_akci }
volaniFunkce >> IDENTIFIKATOR(vstupni_hodnoty);
vstupHodnoty >> vstupni_hodnoty, IDENTIFIKATOR | vstupni_hodnoty, “hodnota” |
IDENTIFIKATOR | “hodnota” | zadne
vracHodnoty >> vrat “hodnota”; | vrat IDENTIFIKATOR;
zastaveni >> zastav;
typ >> cislo | logicky
podmOperator >> > | < | <= | >= | == | !=
matVyras >> term | matVyras2
matVyras2 >> + term matVyras2 | - term matVyras2 | zadne
term >> faktor term2
term2 >> * faktor term2 | / faktor term2 | zadne
faktor >> (matVyras) | IDENTIFIKATOR | volaniFunkce | “hodnota”
logVyras >> “hodnota” | volaniFunkce | IDENTIFIKATOR
slozPodm >> podmTerm slozPodm2
slozPodm2 >> || podmTerm slozPodm2 | zadne
podmTerm >> podmFaktor podmTerm2
podmTerm2 >> && podmFaktor term2
podmFaktor >> (slozPodm) | !(podmFaktor) | IDENTIFIKATOR podmOperator
IDENTIFIKATOR | IDENTIFIKATOR podmOperator “hodnota” |
“hodnota” | “hodnota” podmOperator “hodnota” |
“hodnota” podmOperator IDENTIFIKATOR

```