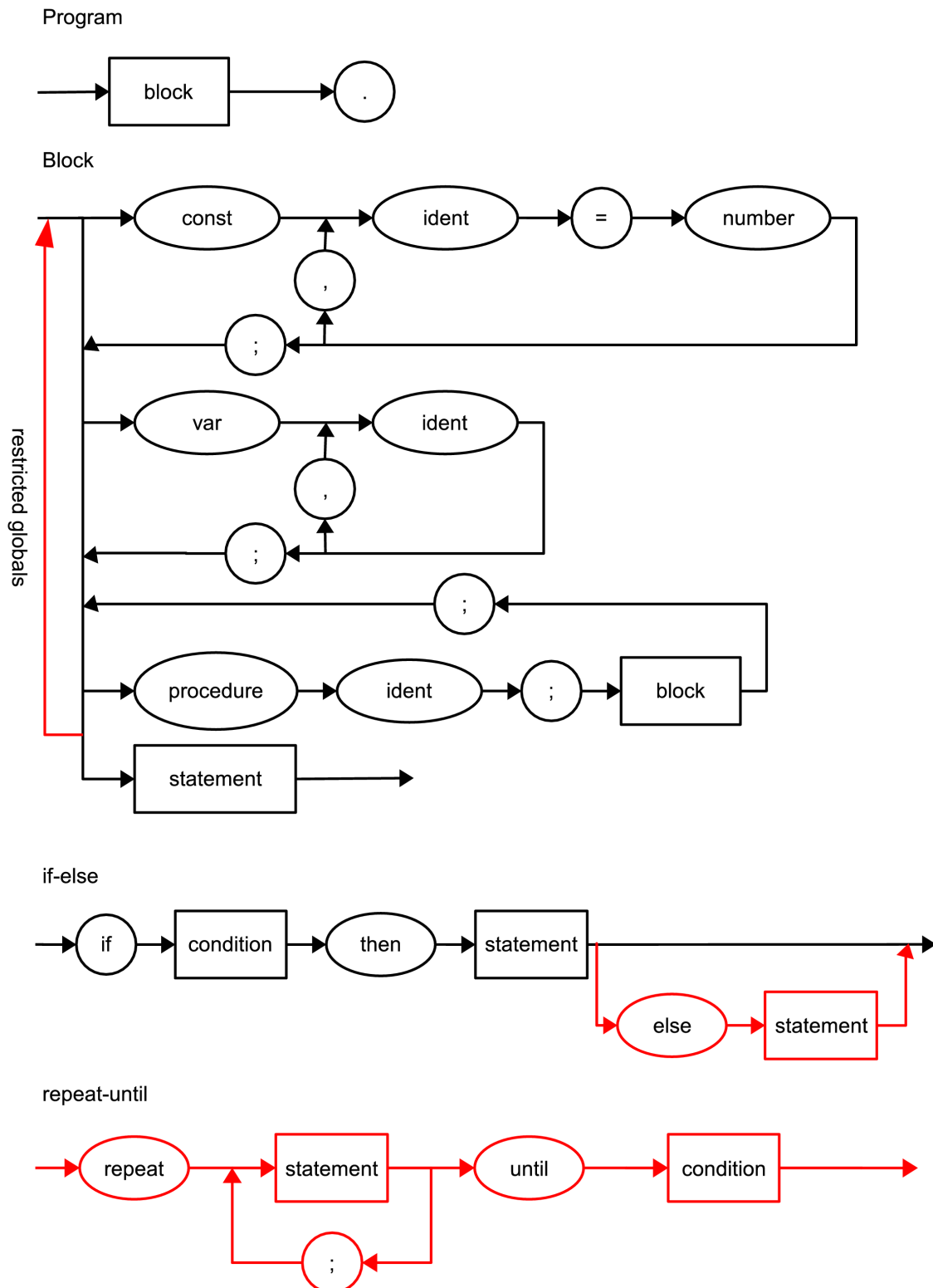


## Modified Syntax Diagrams

Adding:

restricted globals  
if-then-else  
repeat-until  
write/writeln  
for-to/downto-do  
case-of-cend



```

graph LR
    start(( )) --> write(write)
    write --> open("(")
    open --> expr1[expression]
    expr1 --> close(")")
    close --> end(( ))
    close --> comma(",")
    comma --> open

```

```

graph LR
    Start(( )) --> writeln([writeln])
    writeln --> LP("(")
    LP --> expression[expression]
    expression --> RP(")")
    RP --> End(( ))
    expression --> comma([,])
    comma --> expression
  
```

```

graph LR
    start(( )) --> for((for))
    for --> ident([ident])
    ident --> assign((:=))
    assign --> expr1[expression]
    expr1 --> choice(( ))
    choice --> to((to))
    choice --> downto([downto])
    to --> expr2[expression]
    downto --> expr2
    expr2 --> do((do))
    do --> stmt[statement]
    stmt --> end(( ))
  
```

must be a variable!

```
graph LR; Start(( )) --> case((case)); case --> expression[expression]; expression --> of((of)); of --> number((number)); of --> ident((ident)); number --> colon((:)); ident --> colon; colon --> statement[statement]; statement --> semicolon((;)); semicolon --> of; of --> cend((cend)); cend --> End(( ))
```

must be a constant!