

```
program_sym program
identifier exampleGood
semicolon ;
var_sym var
litchar x
comma ,
litchar y
colon :
integer_sym integer
semicolon ;
identifier result
colon :
integer_sym integer
semicolon ;
procedure_sym procedure
identifier printNumMinusThree
lparen (
litchar x
colon :
integer_sym integer
rparen )
semicolon ;
var_sym var
identifier temp
colon :
integer_sym integer
semicolon ;
begin_sym begin
identifier temp
assign :=
litchar x
minus -
number 3
semicolon ;
writeln_sym writeln
lparen (
quotestring "Number minus three is: "
rparen )
semicolon ;
write_sym write
lparen (
identifier temp
rparen )
semicolon ;
end_sym end
semicolon ;
begin_sym begin
litchar y
```

```
assign :=  
number 10  
semicolon ;  
write_sym write  
lparen (  
quotestring "Enter an integers: "  
rparen )  
semicolon ;  
read_sym read  
lparen (  
litchar x  
rparen )  
semicolon ;  
identifier result  
assign :=  
litchar x  
times *  
litchar y  
semicolon ;  
write_sym write  
lparen (  
identifier result  
rparen )  
semicolon ;  
end_sym end  
period .
```