AOZ Studio Beta - Bug #325

Functions with floating point results lock up AOZ. Also, Integer functions return floating point results!

03/31/2020 04:21 AM - Brian Flanagan

Status:	Feedback	**Start date:**	03/31/2020
Priority:	Normal	**Due date:**	
Assignee:	Francois Lionet	**% Done:**	0%
Category:		**Estimated time:**	0:00 hour
Target version:			
Affected version:	0.9.9.4-RC1		

Description

No error displays, but AOZ locks up.
You can't even break out with Ctrl-C!

Example 1:

```
'
' This fails
'
Function "mult#", _num1#, _num2#
End Function ( _num1# * _num2# )
```

Result is that AOZ locks up at black screen. Can't break out.

Example 2: (Even the simplest floating point function will fail at runtime.)

```
'
' Return self
'
Function "slf#", _num#
End Function ( _num# )
```

Example 3:

```
'
' This integer function works (but gets wrong result)
'
Function "mult", _num1#, _num2#
End Function ( _num1# * _num2# )
```

Result: (What the floating point function SHOULD have returned!)

```
29.20999999999997
```

Result SHOULD be:

```
29
```

Perhaps something is reversed between the integer and floating point results?

History

#1 - 04/16/2020 04:10 AM - Francois Lionet

- Status changed from New to Resolved

- Assignee set to Francois Lionet

- Target version set to 0.9.8.1

There were problems in Functions written in Basic, AND for the moment (I can correct this but it needs a bit of work), you CANNOT use # in the name of a function: the transpiler thinks that you are using an array and generates an "Array not dimensioned" error...

THIS works now:

```
Print slf( 2.2 )
Wait key

Function "slf", _num#
End Function ( _num# )
```

#2 - 04/19/2020 09:19 PM - Brian Flanagan

- Status changed from Resolved to Feedback

re-tested in 0.9.8.1 Test 2 Yes, the example you gave works, but I have questions:

1. Are you planning on making the # work for floating point functions in the future?
2. Wouldn't your (temporary?) solution cause a problem if you want to define a function with Integer results?
   I realize I could always just use:

```
End Function(Int(result))
```

...to *work-around* , but **for consistency, shouldn't we let the function's type define the results?**

#3 - 07/03/2020 11:31 PM - Brian Flanagan

- Affected version changed from 0.9.7 to 0.9.9.3

Re-tested in 0.9.9.3. Problems still exist.

#4 - 09/07/2020 08:40 AM - Brian Flanagan

- Target version deleted (0.9.8.1)

- Affected version changed from 0.9.9.3 to 0.9.9.4-RC1

Re-tested in 0.9.9.4-RC1
Although it no longer locks up AOZ, it still gets the non-dimensioned array error.