## How would I create my own CBL\_ALLOC\_MEM and CBL\_FREE\_MEM routines?

## Here are some examples:

	IDENTIFICATION DIVISION.	PROGRAM-ID.	CBL_ALLOC_MEM.	ENVIRONMENT DIVI	
SION.	DATA DIVISION.	LINKAGE SECTION.	01 LNK-MEM-E	POINTER POINTER	
	01 LNK-MEM-SIZE	UNSIGNED-INT.	01 LNK-FLAGS	UNSIGNED-IN	
т.	PROCEDURE DIVISION USI	NG LNK-MEM-POINTER		LNK-ME	
M-SIZE		LNK-FLAGS.	MAIN.	IF LNK-MEM-SI	
ZE GREA	ATER THAN ZERO	CALL "calloc'	" USING BY VALUE 1		
	BY VALUE LNK-MEM-SIZE		RETURN	ING LNK-MEM-	
POINTE	R END-IF.	GOBACK.			

	IDENTIFICATION DIVISION.	PROGRAM-ID.	CBL_FREE_MEM.	ENVIRONMENT DIVIS	
ION.	DATA DIVISION.	LINKAGE SECTION.	01 LNK-MEN	M-POINTER POINTER.	
	PROCEDURE DIVISION USING	LNK-MEM-POINTER.	MAIN.	IF LNK-MEM-POIN	
TER NO	T EQUAL TO ZERO	CALL "free"	USING BY VALUE	LNK-MEM-	
POINTE	R END-IF.	GOBACK.			

Note that both of the above programs must be compiled with the -cp POINTER compatibility option.

On Windows the calloc() and free() functions are in MSVCRT.DLL which must be preloaded by specifying iscobol.shared\_library\_list=msvcrt.dll.

For example, the following test program allocates and frees 100 bytes:

```
id division.program-id. test.data division.working-storage section.77 mem-ptr usage pointer.procedure division.main-logic. call "CBL_ALLOC_MEM" using mem-ptr, by value 100, by value 0. call "CBL_FREE_MEM" using by value mem-ptr.
```

To compile and run:

iscc -cp (ed_library				CBL_FREE	_MEM.cblisc	c -cp	test.cbljava	-Discobol.shar
Online UR	L: <u>https://</u>	support.ve	eryant.co	m/phpkb	/article.php?i	id=15		