MaxBlock Get size of largest block (without compacting)

#include <Memory.h> **Memory Manager**

long **MaxBlock( );** [128K ROMs]
returns size of largest allocatable block

MaxBlock returns the size of the largest block that would be available if heap compaction were to be performed, but does not actually perform the time-consuming compaction.

Returns: a long integer; the size of largest block available after compaction.

Notes: There is another routine, **MaxBlockSys**, which performs the equivalent operation to **MaxBlock**, but in the System heap zone. It takes the same parameters as **MaxBlock**.

This works a little like **CompactMem**, except that it does not actually force heap compaction (compaction will be performed, if needed, when the handle gets allocated anyway).

Use this before attempting an operation that will require allocation of large blocks of memory. Note that a block larger than *returned* may be available; i.e., if a memory purge takes place.

If **MaxBlock** does not yield a large enough value, you could try **PurgeMem**, (another fast operation), followed by **MaxBlock** and check the result.