Thomas Fiorilla : trf40

My metadata is implemented such that it takes up 3 bytes.
   Rather than a struct, which would be padded and thus 4 bytes, I went with implementing it as a series of two types:
          Char = the in-use byte; written and read directly from the array
          Short = stores the size of the following data block

   I cast the memory block at a specific location as a dereferenced short pointer in order to read those two bytes as a short, rather than directly reading the array byte by byte.
   It was possible for me to make my metadata 50% smaller by instead implementing the short's signed nature, making positive shorts "empty" and negative shorts "in-use", negating the need for the in-use byte entirely;
          however, I felt the in-use byte might be a bit more consistent and easier to work with.