Please replace the paragraphs that start with “The stack is fast” and “Data with a size unknown” in the box on page 58 with this paragraph:

All data stored on the stack must have a known, fixed size. Data with an unknown size at compile time or a size that might change must be stored on the heap instead. The heap is less organized: when you put data on the heap, you request a certain amount of space. The operating system finds an empty spot in the heap that is big enough, marks it as being in use, and returns a pointer, which is the address of that location. This process is called allocating on the heap and is sometimes abbreviated as just allocating. Pushing values onto the stack is not considered allocating. Because the pointer is a known, fixed size, you can store the pointer on the stack, but when you want the actual data, you must follow the pointer.

Then please add this paragraph between the paragraph that starts with “Think of being seated at a restaurant” and the paragraph that starts with “Accessing data in the heap” on page 59:

Pushing to the stack is faster than allocating on the heap because it never has to search for a place to store new data; that location is always at the top of the stack. Comparatively, allocating space on the heap requires more work, because the operating system must first find a big enough space to hold the data and then perform bookkeeping to prepare for the next allocation.