Iterator Loops

Turning back to our image processing code, you can see that the pointer `pete` points to the first byte in our image once it is loaded into memory. Of course, **`pete`** is a **`const`** pointer, so it can't be changed. To process the image we need to create **a pair of pointers** like this:

- **`beg`** will be a non-const pointer which will move through all of the pixel data (using address arithmetic), so we can modify the image.
- **`end`** will be a **`const`** pointer that will contain the address just past the end of the data that **`stbi_load()`** has placed in memory. We can calculate this address also by using address arithmetic.

Here's the code you should add to **`main.cpp`** to create these two pointers.

```
unsigned char *beg = pete;    // beginning of the image
unsigned char * const end = pete + width * height * channels;
```

Notice that the expression **`width * height * channels`** is the total number of bytes in the image. By adding it to the pointer **`pete`**, we get a new address that is pointing at the first byte **following the image** in memory.

With these pointers, we can "visit" every byte in the image by using this **iterator loop**:

```
while (beg != end)
{
    // process the byte here
    beg++;    // move to the next byte
}
```