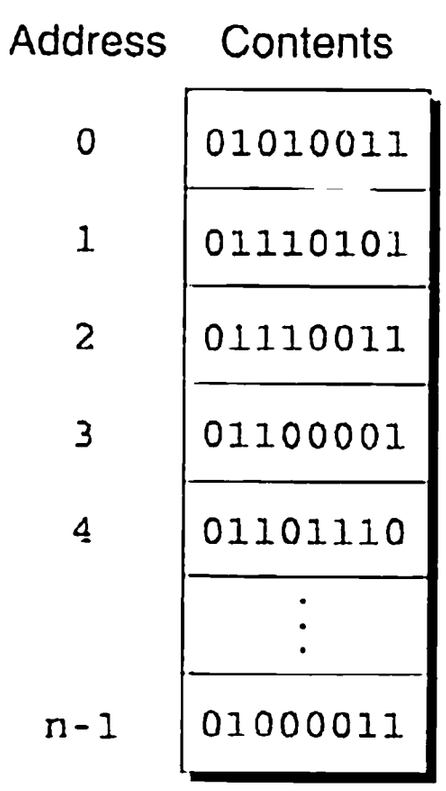
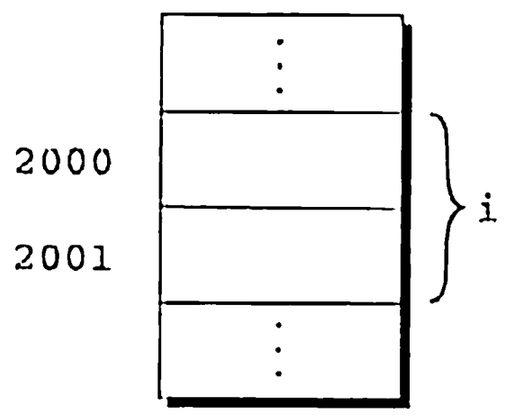


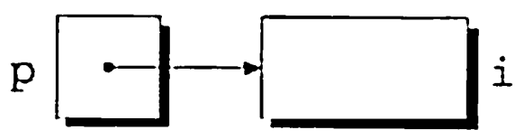
the address of the first byte is said to be the address of the variable. In the following figure, the variable `i` occupies the bytes at addresses 2000 and 2001, so `i`'s address is 2000:



Here's where pointers come in. Although addresses are represented by numbers, their range of values may differ from that of integers, so we can't necessarily store them in ordinary integer variables. We can, however, store them in special ***pointer variables***. When we store the address of a variable `i` in the pointer variable `p`, we say that `p` "points to" `i`. In other words, a pointer is nothing more than an address, and a pointer variable is just a variable that can store an address.

Q&A

Instead of showing addresses as numbers in our examples, I'll use a simpler notation. To indicate that a pointer variable `p` stores the address of a variable `i`, I'll show the contents of `p` as an arrow directed toward `i`:



Declaring Pointer Variables

A pointer variable is declared in much the same way as an ordinary variable. The only difference is that the name of a pointer variable must be preceded by an asterisk:

```
int *p;
```