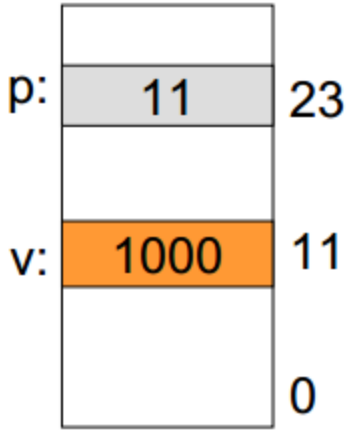
Pointers

- What is a pointer
  - A variable whose value is the address of another variable
  - **p** is a pointer to variable **v**
- Operations
  - **&:** address of (reference)
  - ***:** indirection (dereference)
- Declaration mimics use
  - **int *p;**
    p is the address of an **int**
    (dereference **p** is an integer)
  - **int v;**
    p = &v;
    p stores the address of **v**



http://www.cs.princeton.edu/courses/archive/spr04/cos217/lectures/Pointers.pdf

Pointer Operation Examples

- Examples of * and &

```
int x, y, *p;
p = &x;          /* p gets the address of x */
y = *p;          /* y gets the value point to by p */
y = *(&x);       /* same as y = x */
```

- Unary operators associate right to left

```
y = *&x;         /* same as y = *(&x) */
```

- Unary operators bind more tightly than binary ones

```
y = *p + 1;      /* same as y = (*p) + 1; */
y = *p++;        /* same as y = *(p++); */
```

http://www.cs.princeton.edu/courses/archive/spr04/cos217/lectures/Pointers.pdf

More Pointer Examples

- References (e.g., *p) are variables

```
int x, y, *px, *py;


px = &x;          /* px is the address of x     */
*px = 0;          /* sets x to 0                */
py = px;          /* py also points to x        */
*py += 1;         /* increments x to 1          */
y = (*px)++;      /* sets y to 1, x to 2        */
```

- What about the following?

```
++*px
*px++
```

http://www.cs.princeton.edu/courses/archive/spr04/cos217/lectures/Pointers.pdf