## Pointers Can Serve Many Purposes

In general, we can
- **add an arbitrary number of pointers**
- to any structure.

Pointers can be **used to organize groups of structures** in different ways.
- orderings
- relationships
- properties

  

## Example: Use Linked List to Maintain Ordering

For example, say that we want to sort players
- by name,
- by age, and
- by number of games played.

We **can maintain all three orderings**
- **using** three separate "next" fields (`player_t*`) in the player structure.
- Each field corresponds to a single ordering.

  

## Example: Abstract Syntax Trees (ASTs)

Another example:
- **abstract syntax trees (ASTs)**
- used as an intermediate representation (IR) of a program for compilation

**Nodes represent operators or statements**,
- **operands** are a relation to operators, and
- **initialization, tests, and updates** are a relation to statements (if, for, while, do).
- **All make use of pointers** to other nodes.

  

## Illustration of an AST Construction

```
for (p = player_list.next;
     &player_list != p;
     p = p->next) {
    // do something for all players
}
```