**Pointers to structures.**

Like any other type, structures can be pointed to by its own type of pointers:

|  |  |  |
| --- | --- | --- |
| 1 2 3 4 5 6 7 | struct movies\_t {  string title;  int year;  };  movies\_t amovie;  movies\_t \* pmovie; |  |

Here *amovie* is an object of structure type *movies\_t*, and *pmovie* is a pointer to point to objects of structure type *movies\_t*. Therefore, the following code would also be valid:

|  |  |  |
| --- | --- | --- |
|  | pmovie = &amovie; |  |

The value of the pointer *pmovie* would be assigned the address of object *amovie*.

Now, let's see another example that mixes pointers and structures, and will serve to introduce a new operator: the *arrow* operator *(->)*:

|  |  |  |  |
| --- | --- | --- | --- |
| 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 | // pointers to structures  #include <iostream>  #include <string>  #include <sstream>  using namespace std;  struct movies\_t {  string title;  int year;  };  int main ()  {  string mystr;  movies\_t amovie;  movies\_t \* pmovie;  pmovie = &amovie;  cout << "Enter title: ";  getline (cin, pmovie->title);  cout << "Enter year: ";  getline (cin, mystr);  (stringstream) mystr >> pmovie->year;  cout << "\nYou have entered:\n";  cout << pmovie->title;  cout << " (" << pmovie->year << ")\n";  return 0;  } | Enter title: Invasion of the body snatchers  Enter year: 1978    You have entered:  Invasion of the body snatchers (1978) | [Edit & Run](https://www32.cplusplus.com/doc/tutorial/structures/) |

The arrow operator (->) is a dereference operator that is used exclusively with pointers to objects that have members. This operator serves to access the member of an object directly from its address. For example, in the example above:

|  |  |  |
| --- | --- | --- |
|  | pmovie->title |  |

is, for all purposes, equivalent to:

|  |  |  |
| --- | --- | --- |
|  | (\*pmovie).title |  |

Both expressions, *pmovie->title* and *(\*pmovie).title* are valid, and both access the member title of the data structure pointed by a pointer called *pmovie*. It is definitely something different than:

|  |  |  |
| --- | --- | --- |
|  | \*pmovie.title |  |

which is rather equivalent to:

|  |  |  |
| --- | --- | --- |
|  | \*(pmovie.title) |  |

This would access the value pointed by a hypothetical pointer member called title of the structure object *pmovie* (which is not the case, since title is not a pointer type). The following panel summarizes possible combinations of the operators for pointers and for structure members:

|  |  |  |
| --- | --- | --- |
| **Expression** | **What is evaluated** | **Equivalent** |
| a.b | Member b of object a |  |
| a->b | Member b of object pointed to by a | (\*a).b |
| \*a.b | Value pointed to by member b of object a | \*(a.b) |