Lecture 7

Devendra Ghate

21/12/2020

"Always code as if the guy who ends up maintaining your code will be a violent psychopath who knows where you live" — John Woods (Professor of Logic, University of British Columbia)

List Initialisation

```
const int code = 66;
int x = 66;
char c1 {31325}; //narrowing, not allowed
char c2 {66}; //allowed because char can hold 66
char c3 {code}; //ditto
char c4 {x}; //not allowed, x is not constant
```

Type conversion

```cpp
#include <iostream>

using namespace std;

int main(){
    int guess(1.23456);
    int debt = 1.23456E12;
    cout << "guess: " << guess << endl;
    cout << "debt: " << debt << endl;
    return 0;
}
```

Typecast

Listing 3.14 **typecast.cpp**

```
// typecast.cpp -- forcing type changes
#include <iostream>
int main()
{
    using namespace std;
    int auks, bats, coots;

    // the following statement adds the values as double,
    // then converts the result to int
    auks = 19.99 + 11.99;

    // these statements add values as int
    bats = (int) 19.99 + (int) 11.99;   // old C syntax
    coots = int (19.99) + int (11.99);  // new C++ syntax
    cout << "auks = " << auks << ", bats = " << bats;
    cout << ", coots = " << coots << endl;

    char ch = 'Z';
    cout << "The code for " << ch << " is ";     // print as char
    cout << int(ch) << endl;                     // print as int
    cout << "Yes, the code is ";
    cout << static_cast<int>(ch) << endl;       // using static_cast
```

Typecast

- `static_cast<int>(variableName)`
- `int(variableName)`

Automatic type allocation

- `auto i=12` declares `i` as `int`
- `auto i=1.2` declares `i` as `double`

Arrays

- Array input manipulation (./code/11-arrayInputManipulation.cpp)
- Array memory access (./code/11a-arrayMemoryAccess.cpp)