

C++ part 4

Ned Lowe

ned.lowe@ic.ac.uk

“In theory, there is no difference between theory and practice; In practice, there is”

Arrays

- `int person1score = 73;`
`int person2score = 38;`
`int person3score = 45;`
`int person4score = 91;`
- Rather than continuing like this, we can create an ‘array’
 - Just places many identically typed variables into a single variable
 - You can refer to each variable using an index
 - Can treat each block of the array as a normal variable of that type

Creating and using arrays

- *type varname[size];*
- `int myIntArray[10];`
- `char ReallyString[75];`

- First block is referenced by '0'
- Nth block is referenced by 'n-1'
- `myIntArray[4] = 35;`
- `ReallyString[0] = 'N';`

'Off by one' error

- Writing to a location outside the array is permissible, with disastrous results
- ```
int myIntArray[10];
 myIntArray[15] = 5;
```
- The 'off by one' error is a very common occurrence of this problem
- ```
int myIntArray[10];  
    myIntArray[10] = 5;
```

```
#include <iostream.h>
int main() {
int classScores[10];
int i;
for (i=0; i<10; i++) {
    cout << "Enter Score number " << i << endl;
    cin >> classScores[i]; }
int MaxNum = 0;
int MinNum = 100;
float Average = 0;
for (i=0; i<10; i++) {
    if (classScores[i] > MaxNum)
        MaxNum = classScores[i];
    if (classScores[i] < MinNum)
        MinNum = classScores[i];
    Average = Average + classScores[i]; }
cout << "Max Score was: " << MaxNum <<endl;
cout << "Min Score was: " << MinNum <<endl;
cout << "Average Score was: " << Average/10 <<endl; }
```

Two Dimensional Arrays

- Sometimes it helps to visual data like a chessboard
- A two dimensional array allows us to place items into memory in this manner
- Any block can be referenced using two indexes

- `int MyTwoDArray[5][5];`
- `MyTwoDArray[1][2] = 15;`
- Index starts from '0' as before, through to [n-1]
- Referencing every block in the 2D array can be achieved by using a double for loop

```
#include <iostream.h>
int main() {
int Board[5][5]
int Counter = 0;
for (int i = 0; i<5; i++)
    for (int j=0; j<5; j++) {
        Board[i,j] = Counter;
        Counter++;
    }
for (i = 0; i<5; i++) {
    for (j=0; j<5; j++)
        cout << Board[i][j] << " ";
    cout << endl;
}
}
```

Multi-Dimensional Arrays

- The concept can be taken to as many dimensions as you wish
- Just add additional indexes each time
 - `myArray[i][j][k][x][y][z];`
- Becomes difficult to visualise
 - therefore difficult to apply

Strings

- An array of characters is a string
- We have printed strings before:
 - `cout << "This is a string!";`
- When dealing with user input and variable strings we have to deal with the string as an array
- BUT because we do this so often we have some shortcuts

Strings as Variables

- `char MyString[] = "Hello World!";`
 - creates a string called MyString and makes it long enough to hold the string
 - can't reassign to variable
- `char buffer[50];`
`cout << "Enter your name: ";`
`cin >> buffer;`
`cout >> "Your name is " >> buffer >> endl;`
- Problem if you enter a space

```
#include <iostream.h>
```

```
int main() {
```

```
    char buffer[50];
```

```
    cout << "Enter your name: ";
```

```
    cin >> buffer;
```

```
    cout >> "Your name is " >> buffer >> endl;
```

```
    return 0;
```

```
}
```

cin.get

- Instead of using normal cin with user inputted strings we can use cin.get
- It takes three parameters
 - The variable to use as a buffer
 - Maximum number of characters to get
 - Delimiter that terminates input
 - by default 'newline'
- `cin.get(buffer,49);`

```
#include <iostream.h>
```

```
int main() {
```

```
    char buffer[50];
```

```
    cout << "Enter your name: ";
```

```
    cin.get(buffer,49);
```

```
    cout >> "Your name is " >> buffer >> endl;
```

```
    return 0;
```

```
}
```

Assigning strings - strcpy

- We use `#include <string.h>`
- `strcpy(String2, String1);`
 - Sets String2 the same as String1
- String2 must be big enough to allow String1 to fit into it, otherwise it will overwrite the end of the buffer

```
#include <iostream.h>
```

```
#include <string.h>
```

```
int main() {
```

```
    char buffer[50];
```

```
    char buffer2[50];
```

```
    cout << "Enter your name: ";
```

```
    cin.get(buffer,49);
```

```
    strcpy(buffer2,buffer);
```

```
    cout << "Your name is " << buffer2 << endl;
```

```
    return 0;
```

```
}
```