University of Hertfordshire

7COM1025
Programming for Software Engineers
Lecture 3

Dr. Renato C. de Amorim

THE ONE HUNDRED UNDER FIFTY

BASIC: CASTS

```
#include <iostream>
using namespace std;
int main()
{
    cout<<3/2<<"\n";
    cout<<(float)3/2<<"\n";
    return 0;
}
```

BASIC: IF IN DETAIL

- If-else-if ladder

```
If (condition)
      statement;
else if(condition)
      statement;
else
      statement;
```

- Nested ifs

```
If(i){
      if(j) result=1;
      if(k) result=2;
      else result=3;
}
else result=4;
```

What's the value of result in the following cases: (a) i=false, j=false, k=true; (b)i=true, j=true, k=true; (c)i=true, j=true, k=false1; (d) i=true, j=false, k=false

BASIC: SWITCH

```
#include <iostream>
using namespace std;
int main()
{
	int num;
	cout<<"Enter a number from 1 to 3";
	cin >> num;
	switch(num) {
		case 1:
			cout<<"You entered 1";
			break;
		case 2:
			cout<<"You entered 2";
			break;
		case 3:
			cout<<"You entered 3";
			break;
		default:
			cout<<"You entered something else";
	}
	return 0;
}
```

- What happens if you remove the break statement?

Problem 2.3

Help System

Let’s write a very basic help system for c++, it should have four options (1, 2, 3, 4). Each option should relate to a c++ command (say, 1 for cout, 2 for cin, 3 for for, 4 for return).

When the user chooses his number, your program should show a small piece of text that explain the c++ command.
Don’t forget to put a very simple menu so the user knows that (as an example) 1 is for cout.

BASIC: FOR IN DETAIL

```
for (initialization; expression; increment/decrement)
     statement;
```

Multiple loop control variables

```
#include <iostream>
using namespace std;
int main()
{
    int x,y;
     for(x=0, y=10; x<=y; ++x, --y)
          cout<<x<<" "<<y<<"\n";
     return 0;
}
```

BASIC: FOR IN DETAIL II (CONT)

- Note that each piece in the for loop is optional. Infinite loop:

```
for(;;)
    statement;
```

Likewise the body is optional

```
#include <iostream>
using namespace std;
int main()
{
    int i, sum=0;
    for (i=1; i<=10; sum+=i++);
    cout << "Sum is "<<sum;
    return 0;
}
```

BASIC: OTHER LOOPS

```
while(expression)
    statements;
```

If the statements should happen at least once:

```
do{
    statements;
} while (condition);
```

Notes:

Optionally, you can use “break;” to exit a loop.

To force an early iteration you can use “continue;”.

Problem 2.2

A better help system

Lets add a new option to our previous help system: 5 – Exit. Unless the user chooses 5 your program should show the menu again and allow the user to choose whatever number he wants.

BASIC: NESTED LOOPS

Just like is statements, loops can be nested (ie. put one inside the other).

```
#include <iostream>
using namespace std;
int main()
{
    int a, b;
    for (a=1; a<=10; a++)
        for (b=1; b<=10; b++)
            cout<<"a: "<<a<<" b: "<<b<<endl;
    return 0;
}
```

BASIC: MORE ON ARRAYS

- There is no bounds checking!

```
int crash[10], i;
for (i=0; i<100; i++) crash[i]=i;
```

(note: you have elements up to index 9, but try to reach up to 99)

- Transferring contents from one array to another

```
int a[10], b[10];
...
a=b; //Error !
```

You should instead:

```
for (i=0; i<10;i++) a[i] = b[i];
```

BASIC: TWO-DIMENSIONAL ARRAYS

```
#include <iostream>
using namespace std;
int main()
{
    int row, col, nums[3][4], sum=0;
    for (row=0; row<3; row++){
        for(col=0;col<4;col++){
            nums[row][col]=++sum;
            cout<<nums[row][col] << " ";
        }
        cout <<"\n";
    }
    return 0;
}
```

Problem 2.3

- Write a program that reads 10 integers from the user and then shows them on the screen sorted.

Hint: you can use bubble sort (below)

6 5 3 1 8 7 2 4