Control structures

C++ Lecture 3

Adam Kohl

Motivation

- Need the ability to add logic to our programs
- Math calculations isn't enough
- Error checking
  - Changing the flow of the program
  - Mainly used to check if a certain part of the code should be executed

If statements

- Allows for changing code flow depending on conditions
- If (condition is true) { execute something } else {execute something else}
- Elseif

```
if (bank_account < 0){
    cout << "Huh?" << endl;
}
elseif (bank_account > 1000000){
    cout << "WHAAAAAAAAAT?" << endl;
}
else{
    cout << "Welcome to the 99%" << endl;
}
```

If statements

- Nesting means to put similar control structures within each other
- Helps in separating logic and making code more readable

```
if (bank_account > 0){
    if (bank_account > 1000000){
        cout << "WHAAAAAAAAAT?" << endl;
    }
    else{
        cout << "Welcome to the 99%" << endl;
    }
}
else{
    cout << "Huh?" << endl;
}
```

Mini Task

Compile the code given in control structures code.cpp and make sure that it runs

If there are any errors how would you fix it?

Switch – case statements

- Sometimes you can go crazy with if statements

```
srand(time(NULL));
int die = rand() % 6 + 1;
if (die == 1){
    cout << "Hello I'm Mercury" << endl;
}
if (die == 2){
    cout << "Hello I'm Venus" << endl;
}
if (die == 3){
    cout << "Hello I'm Earth" << endl;
}
if (die == 4){
    cout << "Hello I'm Mars" << endl;
}
if (die == 5){
    cout << "Hello I'm Jupiter" << endl;
}
if (die == 6){
    cout << "Hello I'm Saturn" << endl;
}
```



```
switch (die){
case(1):
    cout << "Hello I'm Mercury" << endl;
    break;
case(2) :
    cout << "Hello I'm Venus" << endl;
    break;
case(3) :
    cout << "Hello I'm Earth" << endl;
    break;
case(4) :
    cout << "Hello I'm Mars" << endl;
    break;
case(5) :
    cout << "Hello I'm Jupiter" << endl;
    break;
case(6) :
    cout << "Hello I'm Saturn" << endl;
    break;
default:
    cout << "Hello I'm want to be Pluto" << endl;
}
```

Switch – case statements

```
switch (die){
case(1):
    cout << "Hello I'm Mercury" << endl;
    break;
case(2) :
    cout << "Hello I'm Venus" << endl;
    break;
case(3) :
    cout << "Hello I'm Earth" << endl;
    break;
case(4) :
    cout << "Hello I'm Mars" << endl;
    break;
case(5) :
    cout << "Hello I'm Jupiter" << endl;
    break;
case(6) :
    cout << "Hello I'm Saturn" << endl;
    break;
default:
    cout << "Hello I'm want to be Pluto" << endl;
}
```

Works with char, int, float, double NOT string

Need to put a break statement to avoid it bleeding into other statements

Goes to this line if none of the cases match

Mini Task

Convert the if statements to switch case statements

Loops

- 3 main types
  - While
  - Do – while
  - For loop
- You can repeat blocks of code based on certain conditions

While loop

While (condition is true) { execute something }

- Will keep running till the condition is false

```
cout << "Countdown" << endl;
int count = 10;
while (count > 0){
    cout << count << endl;
    count = count - 1;
    Sleep(1000);
}
```

Do while

- While loop backwards

  do {execute something} while (condition is true);

- Will execute the body of the loop at least once

```
cout << "Countdown" << endl;
int count = 10;
do{
    cout << count << endl;
    count = count - 1;
    Sleep(1000);
} while (count > 0);
```

For loop

- Convenience function that does a lot for us

for (initialize loop variable ; check condition is true ; increment loop variable) { execute something }

C++ takes care to execute everything properly and in order

```
for (int count = 10; count > 0; count = count - 1){
    //Do something
    cout << count << endl;
}
```

For loop

```
for (int count = 10; count > 0; count = count - 1){
    //Do something
    cout << count << endl;
}
```

Initialization of loop variable run once only before entering the loop

Condition check run every time before entering the loop

Run every time after finishing the loop

Mini Task

Make modifications to while loop

Convert to do while

Convert to for loop

Questions?

Assignment

- Do the control structures assignment.cpp
- If you finish quickly enough do the advanced one