You Have Not Received Any Emails In The Last Hour
What Do You Do?

Step 1: PANIC
Is your email broken? How long has it been down? Are there actually important emails that you're NOT GETTING???

Step 2: Send yourself a test email
Did it arrive?
Yes
No

Step 2: Phone
Well, what if it only works for YOU??

Step 3: Annoy Your Friends
Ask them to send you a test email. Surely, this is more important than anything they're doing.
Did it arrive?
Yes
No

There is an indeterminate period in time in which you may or may not have been able to receive important emails

Step 4: Do Something Stupid
E-mail everyone you know and ask if they tried to send you an e-mail in the last hour.
if/else if/else

```java
if (<expression>) {
    <statement>
}
...
else if (<expression>) {
    <statement>
}
...
else {
    <statement>
}
```
Relational and Logical Operators

• == Equality
• >= Greater than or equal to
• <= Less than or equal to
• > Greater than
• < Less than
• != Not equal to
• || OR
• && AND
• ! NOT
Example

• if(1+2)
• if(2-4)
• if(2-2)
• if(4==4)
• if((2+1) == 4)
• if(4.1 == 4)
• if(3 <= 4)
• if(4 >= 4)
• if(3.5 > 4)
• if(4 < 4)
• if(3+2*2 > 9)
• if((3+2)*2 > 9)
Examples Continued

- **AND**: if((1>2) && (2<5))
- **OR**: if((1>2) || (2<5))
- **NOT**: if(!(1>2) && (2<5))

| p  | q  | p && q | p || q | !p |
|----|----|--------|--------|----|
| T  | T  | T      | T      | F  |
| T  | F  | F      | T      | F  |
| F  | T  | F      | T      | T  |
| F  | F  | F      | F      | T  |
if/else Example

if (x > y) {
    cout << "X is greater than Y" << endl;
} else {
    cout << "Y is less than X" << endl;
}

//Are these print statements always true?
if/else if/else Example

if (x > y) {
    cout << "X is greater than Y" << endl;
}
else if (x == y) {
    cout << "X is equal to Y" << endl;
}
else {
    cout << "Y is less than X" << endl;
}
Nested Decisions

if (confused_on_class_procedure == True) {
    if (procedure_in_syllabus == False) {
        cout << "Email TA or Shannon" << endl;
    } else {
        bool still_confused = read_syllabus();
        if (still_confused == True) {
            cout << "Email TA or Shannon" << endl;
        } else {
            cout << "Good job!" << endl;
        }
    }
} else {
}
Notes on Scope

• Block Scope: variables declared exist until the end of the block
• Blocks = {}
int main () {
    int a = 0;
    if (a == 0) {
        int b = a; //a still exists in here
    }
    b = 1; //error: b does not exist out here
    return 0;
}
Alternative: Switch Statements

```java
switch (<expression>) {
    case <const-expression>:
        <statement>;
        ...
        break;
    case <const-expression>:
        <statement>;
        ...
        break;
    default:
        <statement>;
        ...
        break;
}
```
Switch Statement Details

• Tests equality: `<expression> == <const-expression>`
  • If it’s true, execute that code

• Need to have break statements otherwise it will fall through and execute everything else

• Default case is optional but a good idea
```cpp
#include <iostream>
using namespace std;

int main () {
    if (1+2) {
        cout << "1+2 worked" << endl;
    }
    if (2-4) {
        cout << "2-4 worked" << endl;
    }
    if (2-2) {
        cout << "2-2 worked" << endl;
    }
    if (4 == 4) {
        cout << "4 == 4 worked" << endl;
    }
    if ((2+1) == 4) {
        cout << "(2+1) == 4 worked" << endl;
    }
    if (4.1 == 4) {
        cout << "4.1 == 4 worked" << endl;
    }
    if (3 <= 4) {
        cout << "3 <= 4 worked" << endl;
    }
}
```
cout << "4.1 == 4 worked" << endl;
}
if (3 <= 4) {
    cout << "3 <= 4 worked" << endl;
}
if (4 >= 4) {
    cout << "4 >= 4 worked" << endl;
}
if (3.5 > 4) {
    cout << "3.5 > 4 worked" << endl;
}
if (4 < 4) {
    cout << "4 < 4 worked" << endl;
}
if (3+2*2 > 9) {
    cout << "3+2*2 > 9 worked" << endl;
}
if ((3+2)*2 > 9) {
    cout << "(3+2)*2 > 9 worked" << endl;
}
if ((1 > 2) && (2 < 5)) {
    cout << "(1 > 2) && (2 < 5) worked" << endl;
}
if ((1 > 2) || (2 < 5)) {
    cout << "24,2-9 worked" << endl;
}
cout << "(3+2)*2 > 9 worked" << endl;

}  
if((1 > 2) && (2 < 5)) {
    cout << "(1 > 2) && (2 < 5) worked" << endl;
}
if ((1 > 2) || (2 < 5)) {
    cout << "(1 > 2) || (2 < 5) worked" << endl;
}
if (!(1 > 2) && (2 < 5)) {
    cout << "!(1 > 2) && (2 < 5) worked" << endl;
}
return 0;
Vimrc -> in home directory vim .vimrc

```
1 set smartindent
2 set mouse=a
3 set nu
4 colorscheme default
```
Feedback

https://tinyurl.com/yau3323k