

Syntax errors

Remove a semicolon from line 12.

```
test.cpp:4:1: error: expected ';' before 'bool'
```

Just a syntax error because it's missing a semi-colon

Remove the curly brace from line 4.

```
test.cpp:5:4: error: expected initializer before 'int'
test.cpp:6:4: error: expected unqualified-id before 'for'
test.cpp:6:13: error: 'str1' does not name a type
test.cpp:6:28: error: 'i' does not name a type
test.cpp:10:4: error: expected unqualified-id before 'if'
test.cpp:12:4: error: expected unqualified-id before 'return'
test.cpp:13:1: error: expected declaration before '}' token
```

This one throws a bunch of errors. The top one gives you a good idea that something is wrong before the int declaration.

Remove the brackets from the argv arguments in the call to is_equal() on line 17.

```
test.cpp: In function 'int main(int, char**)':
test.cpp:17:24: error: 'argv1' was not declared in this scope
test.cpp:17:31: error: 'argv2' was not declared in this scope
```

It's looking for the argv1 and argv2 variables since they were typed in without brackets. It thinks they are variables instead of an array.

Remove the asterisk/splat from the argv in main on line 15.

```
test.cpp:15:5: warning: second argument of 'int main(int, char*)' should be 'char **' [-Wmain]
test.cpp: In function 'int main(int, char*)':
test.cpp:17:40: error: invalid conversion from 'char' to 'char*' [-fpermissive]
test.cpp:4:6: error: initializing argument 1 of 'bool is_equal(char*, char*)' [-fpermissive]
test.cpp:17:40: error: invalid conversion from 'char' to 'char*' [-fpermissive]
test.cpp:4:6: error: initializing argument 2 of 'bool is_equal(char*, char*)' [-fpermissive]
```

It's looking for a pointer to an array and instead just getting an array argv[]. Another option would be to change it to **argv instead of *argv[]

Remove the return type, bool, from the is_equal() function on line 4.

```
test.cpp:4:32: error: ISO C++ forbids declaration of 'is_equal' with no type [-fpermissive]
```

It's saying that you need to have a type to go with your function declaration

Remove one of the parameters in the `is_equal()` function on line 4.

test.cpp: In function 'bool is_equal(char*)':

test.cpp:6:13: error: 'str1' was not declared in this scope

test.cpp: In function 'int main(int, char**)':

test.cpp:17:40: error: too many arguments to function 'bool is_equal(char*)'

test.cpp:4:6: note: declared here

The last line shows that when the function was called it was sent too many arguments. That either means the function is wrong or the way it was called was wrong. In this case it's the function.

Comment out line 5, and declare `i` in the for loop on line 6.

Logic errors

test.cpp: In function 'bool is_equal(char*, char*)':

test.cpp:10:12: error: name lookup of 'i' changed for ISO 'for' scoping [-fpermissive]

test.cpp:10:12: note: (if you use '-fpermissive' G++ will accept your code)

The first line tells you that there is a problem in the function. The second line shows that the "`i`" variable was not declared in the scope for the IF statement on line 10

Put two asterisks/splats on the parameters of line 4, and put ampersand, `&`, in front of the arguments on line 17.

The logic error is that on line 17 we are sending the function the address of `argv[1]` and `argv[2]` and then trying to see if they are the same. It correctly shows that they're not the same but we really want to send the function

Comment out line 10, and run with `./string_equal hell hello`.

This comments out the line that check to see if the two strings are equal length by looking for a NULL character at the end of the second word. It just sees the NULL on the first string and quits.

Comment out line 16, and run with `./string_equal hello`

This gives a Seg Fault because we are trying to get the value from `argv[2]` which doesn't exist when we only enter one word.