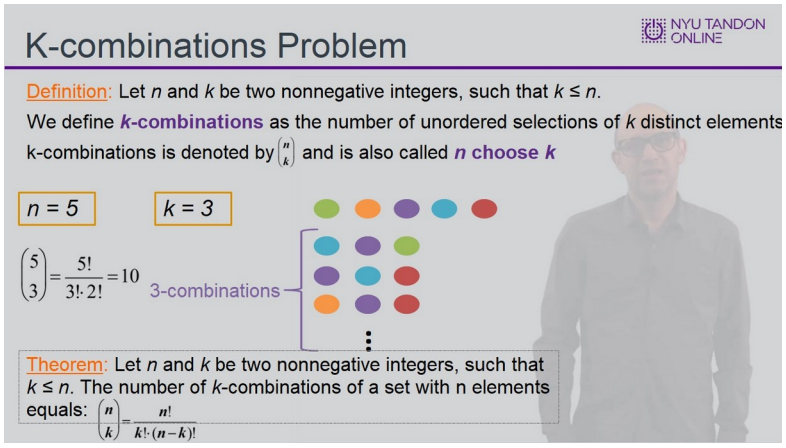
Module 8 Functions

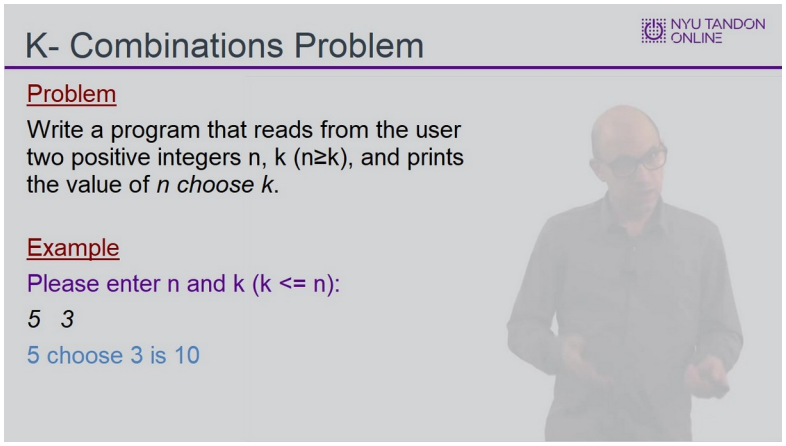
2. Function Definition & Execution Model

2.1 CS Bridge: Functions

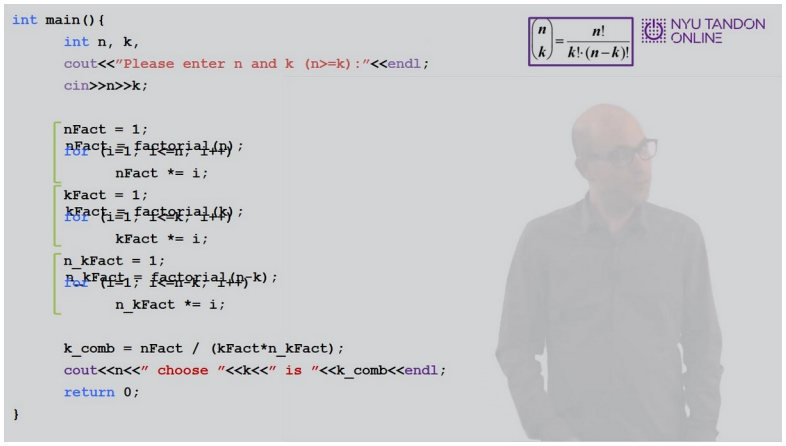
2.2 k-combinations



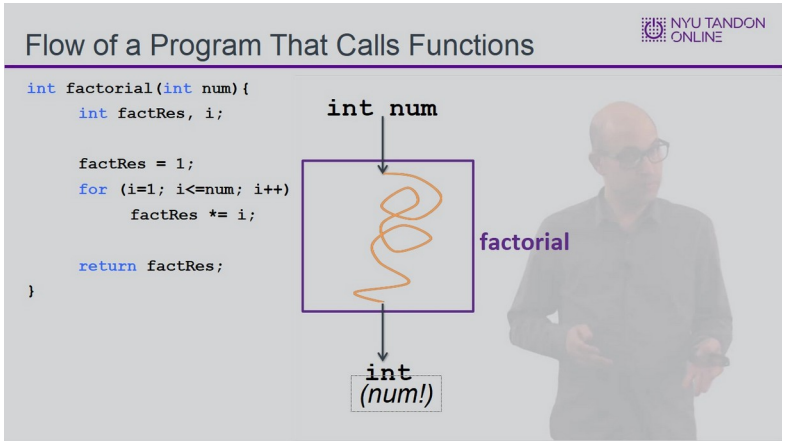
2.3 k-combination Problem



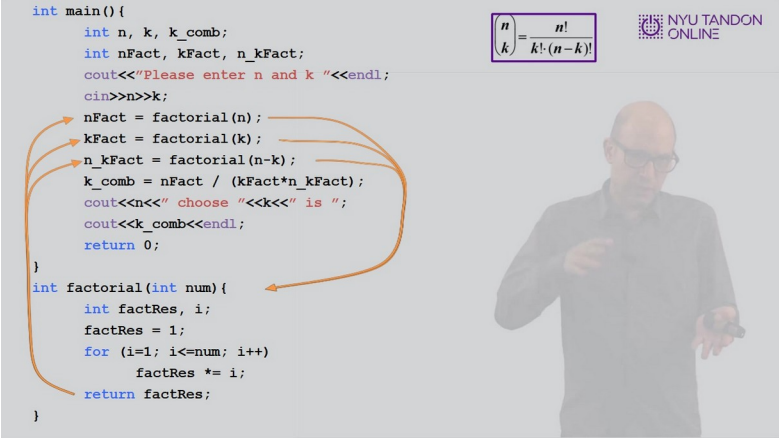
2.4 k-Combination Solution



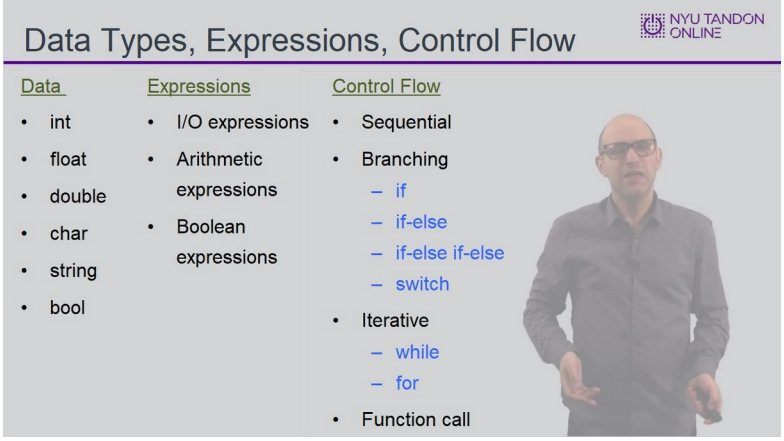
2.5 Flow of a Program That Calls Functions



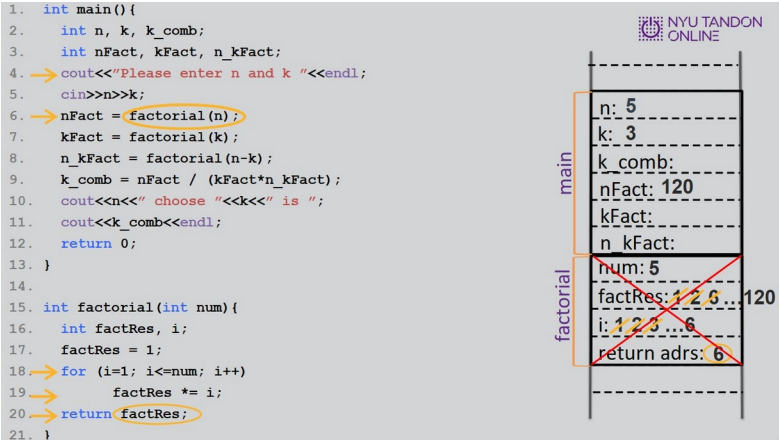
2.6 Calling a Function Code



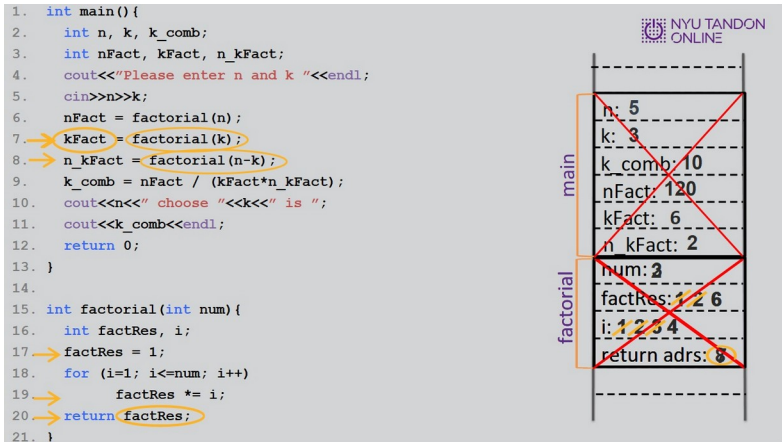
2.7 Data Types, Expressions, Control Flow



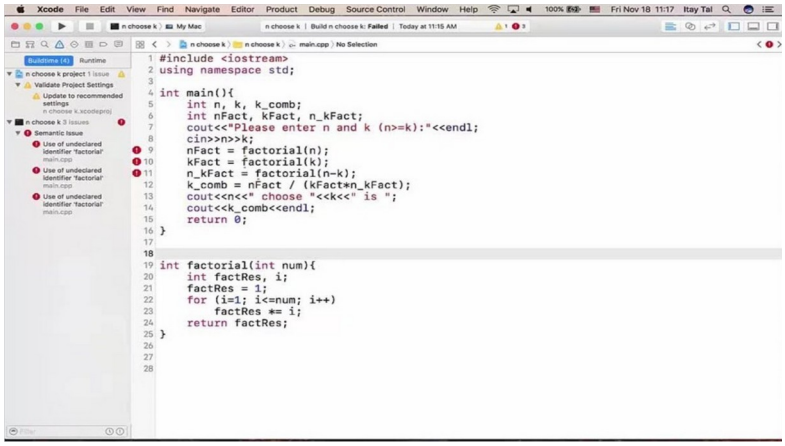
2.8 Runtime Stack Execution Model



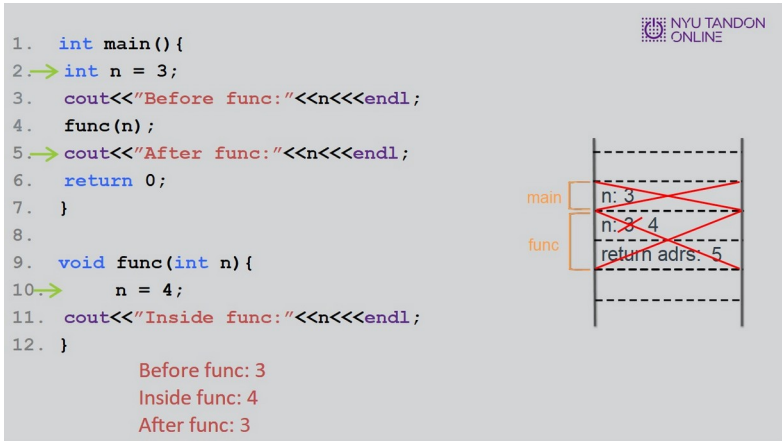
2.9 Runtime Stack Execution Model



2.10 Program Implementation

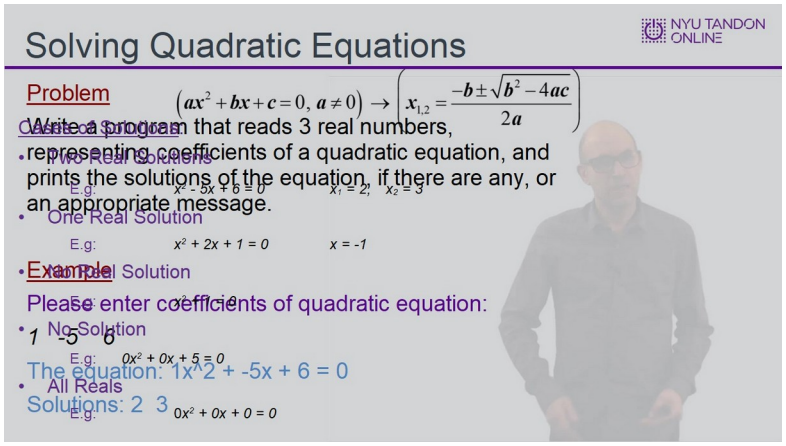


2.11 Scope of Variables Example

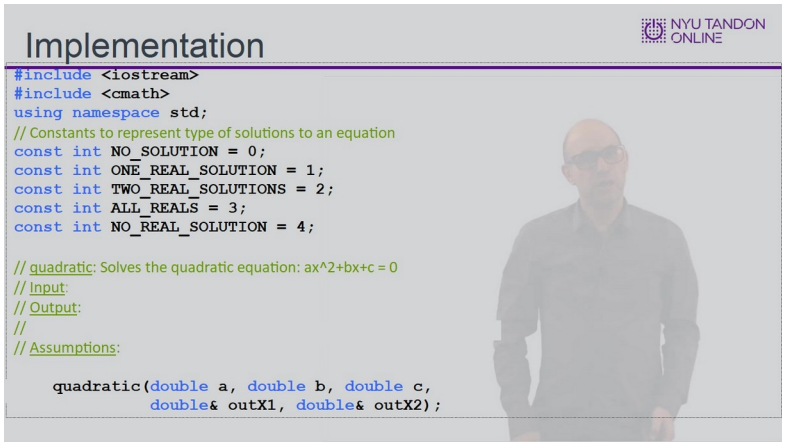


1. Breaking a Program into Functions

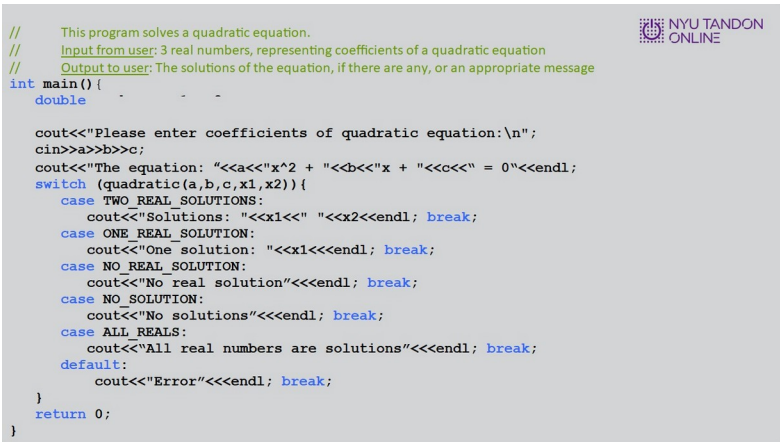
1.1 Solving Quadratic Equation



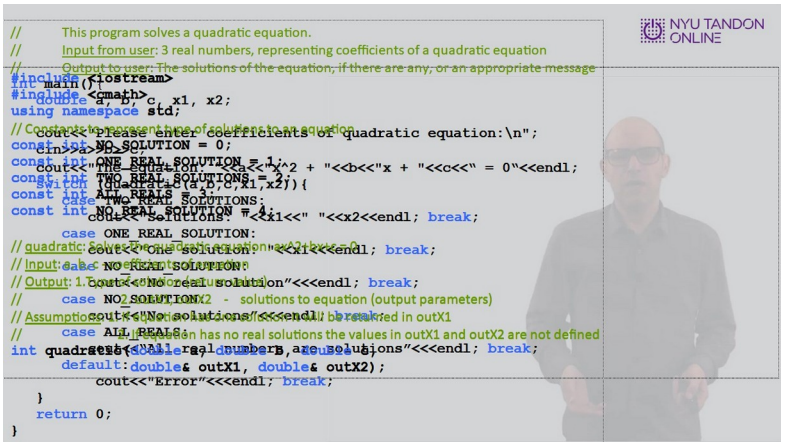
1.2 Implementation 1



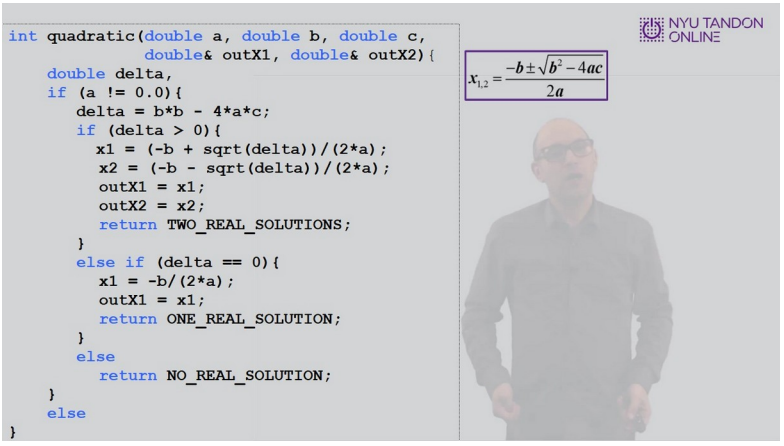
1.3 Implementation 2



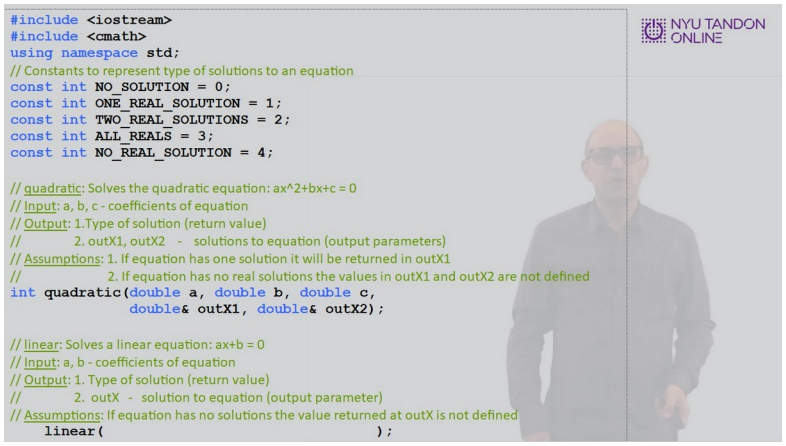
1.4 Implementation 3



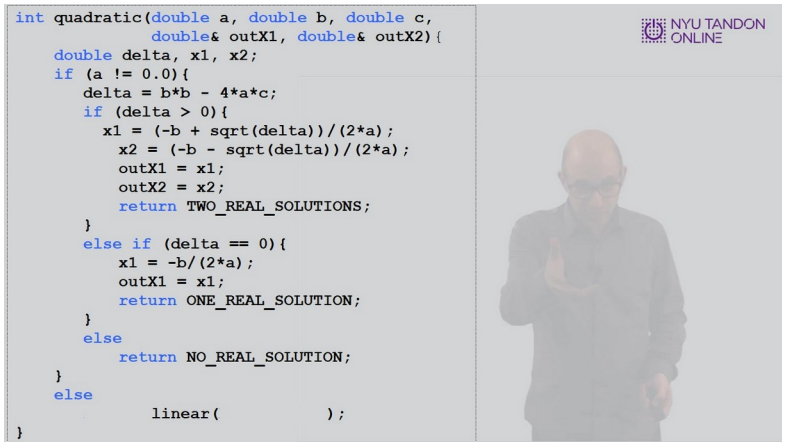
1.5 Implementation 4



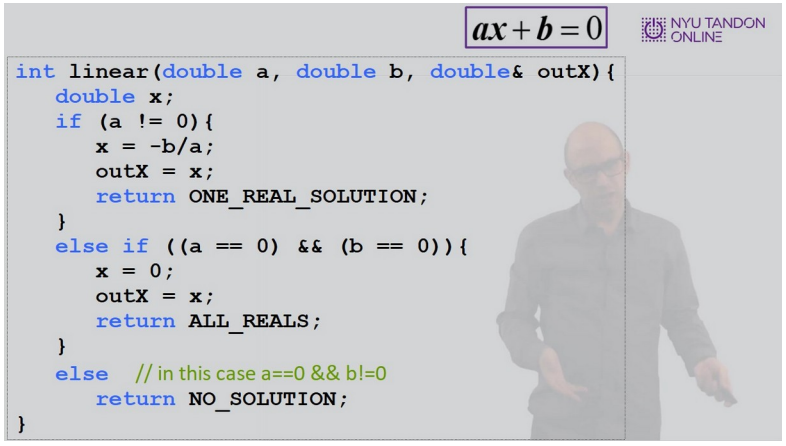
1.6 Implementation 5



1.7 Implementation 6

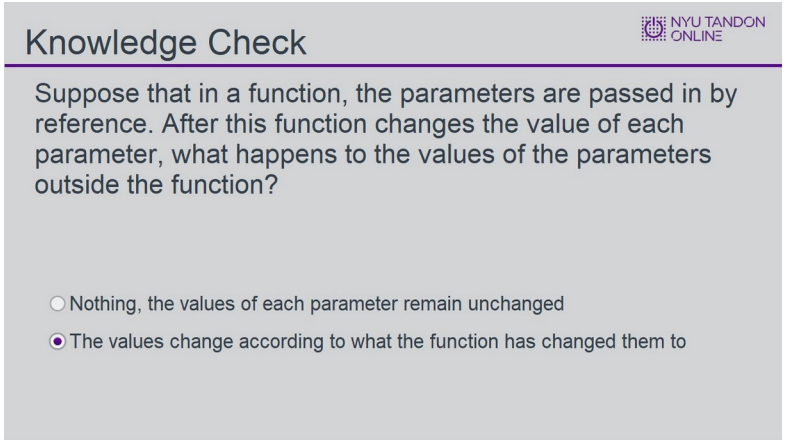


1.8 Implementation 7



1.9 Knowledge Check

(True/False, 10 points, unlimited attempts permitted)

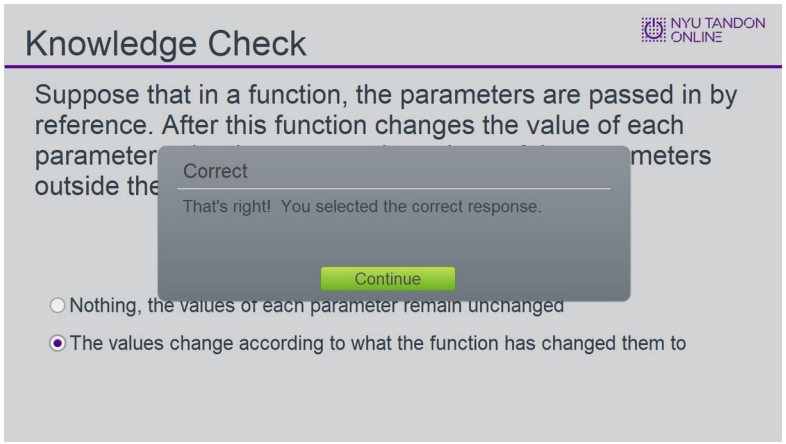


Correct	Choice
	Nothing, the values of each parameter remain unchanged
X	The values change according to what the function has changed them to

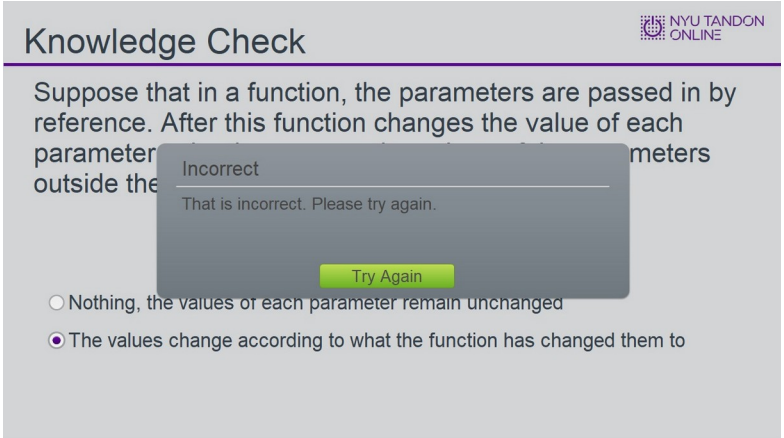
Feedback when correct:

That's right! You selected the correct response.

Correct (Slide Layer)

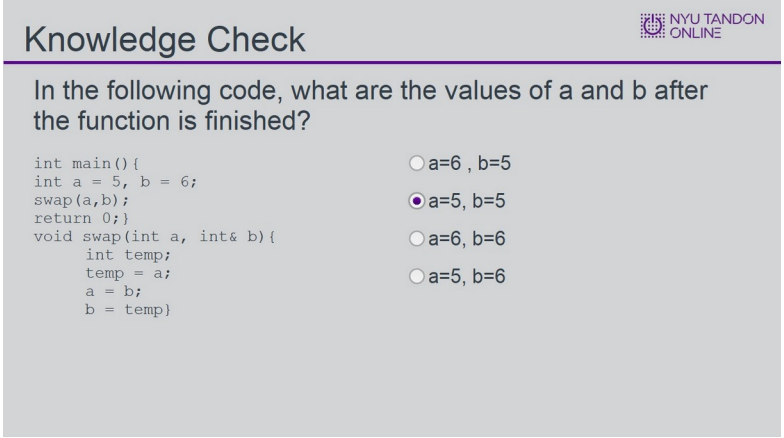


Try Again (Slide Layer)



1.10 Knowledge Check

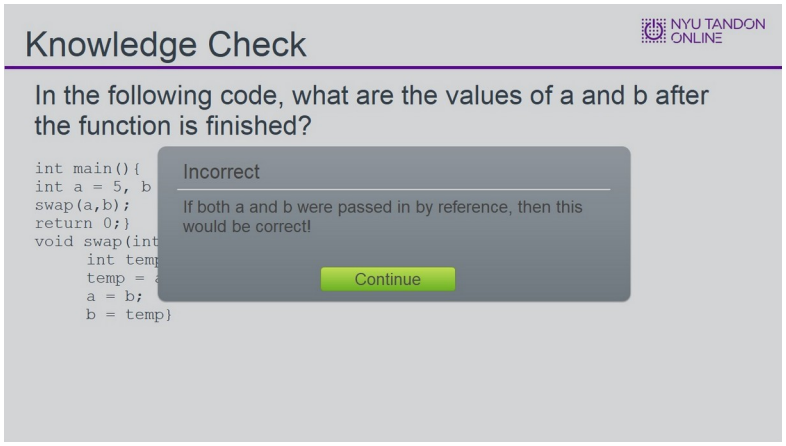
(Multiple Choice, 10 points, 4 attempts permitted)



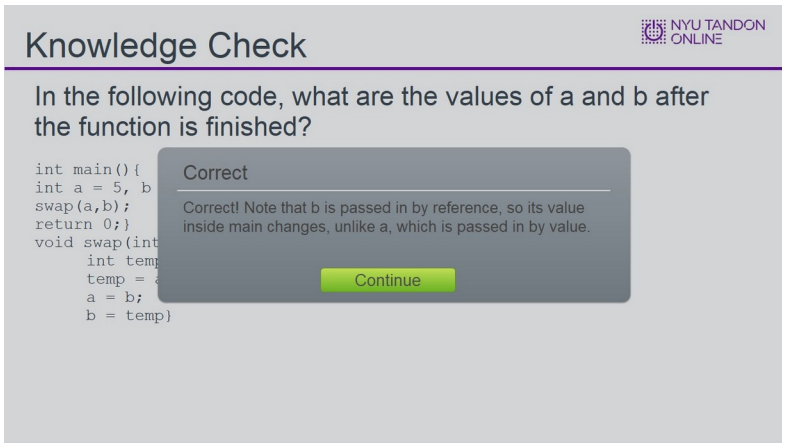
Correct	Choice	Feedback
	a=6 , b=5	If both a and b were passed in by reference, then this would be correct!
X	a=5, b=5	Correct! Note that b is passed in by reference, so its value inside main changes, unlike a, which is passed in by value.

a=6, b=6	Recall that a is passed in by value, so the value of it inside the main function doesn't change!
a=5, b=6	Recall that b is passed in by reference, so the value of it inside the main function changes!

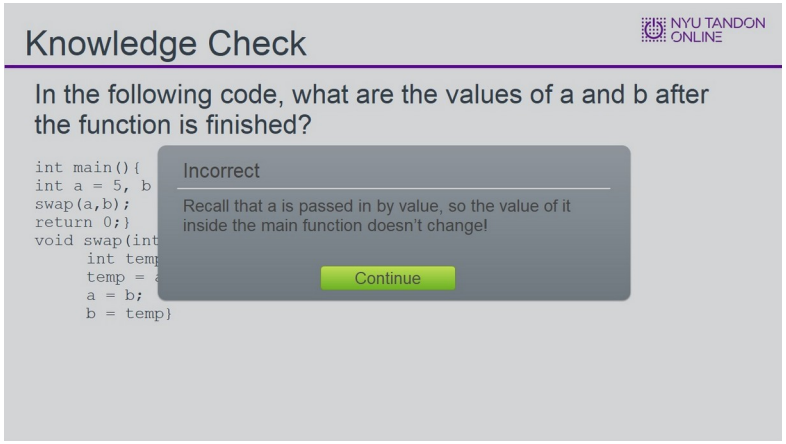
a=6 , b=5 (Slide Layer)



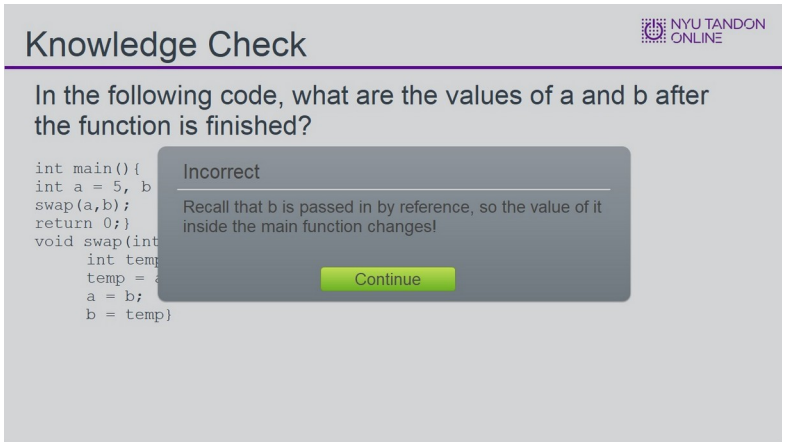
a=5, b=5 (Slide Layer)



a=6, b=6 (Slide Layer)

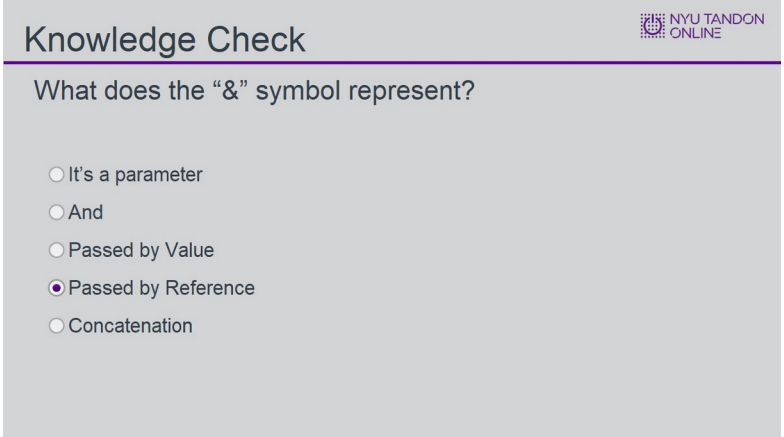


a=5, b=6 (Slide Layer)



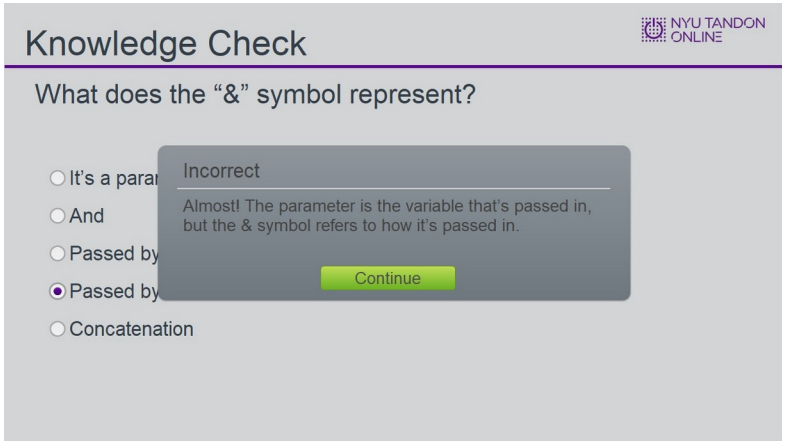
1.11 Knowledge Check

(Multiple Choice, 10 points, 6 attempts permitted)

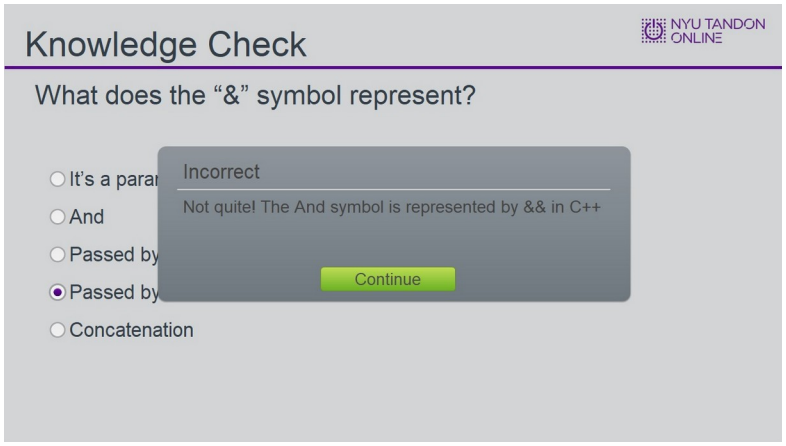


Correct	Choice	Feedback
	It’s a parameter	Almost! The parameter is the variable that’s passed in, but the & symbol refers to how it’s passed in.
	And	Not quite! The And symbol is represented by && in C++
	Passed by Value	Not quite! The & symbol allows values to change outside the function while a parameter passed by value creates a “copy” that only changed within the function.
X	Passed by Reference	Yes! Remember, this means that the value will change even outside of the function.
	Concatenation	Remember, concatenation is represented by + in C++

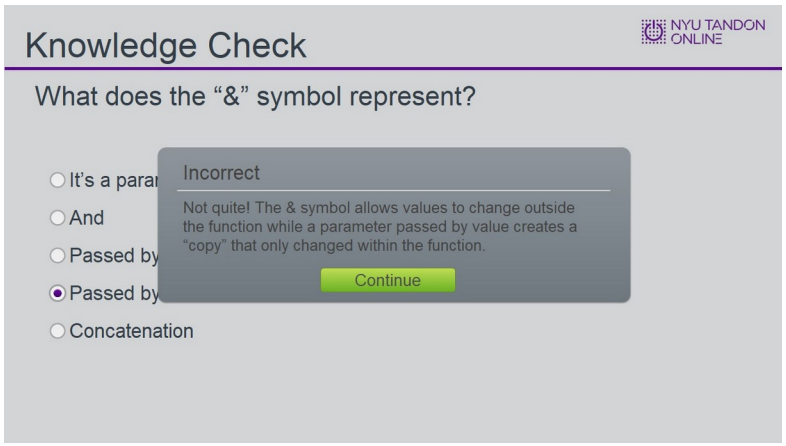
It’s a parameter (Slide Layer)



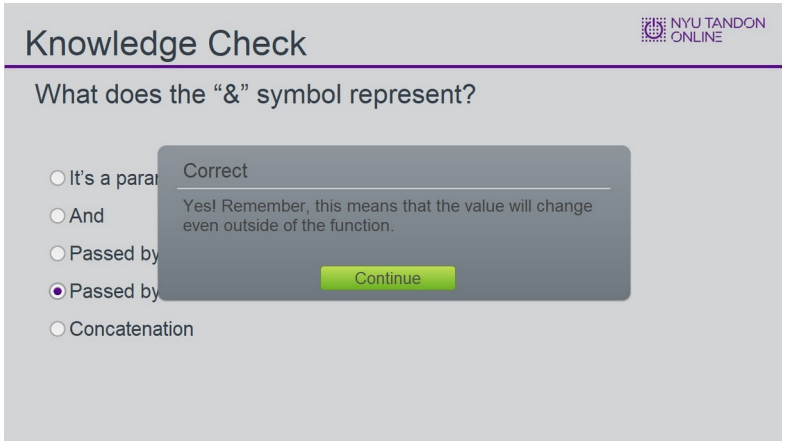
And (Slide Layer)



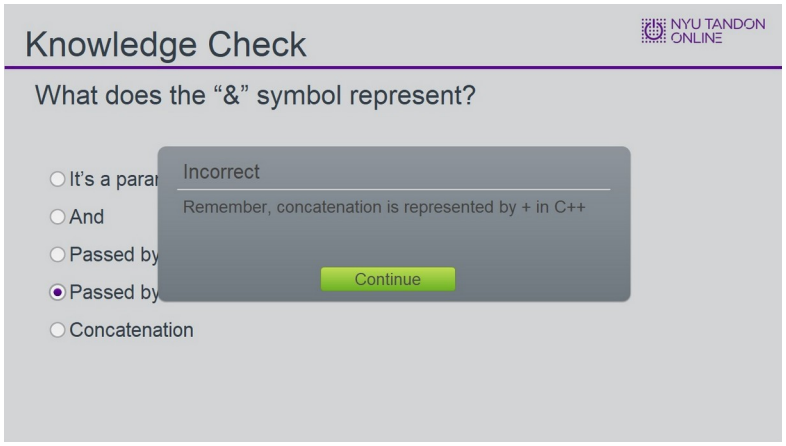
Passed by Value (Slide Layer)



Passed by Reference (Slide Layer)

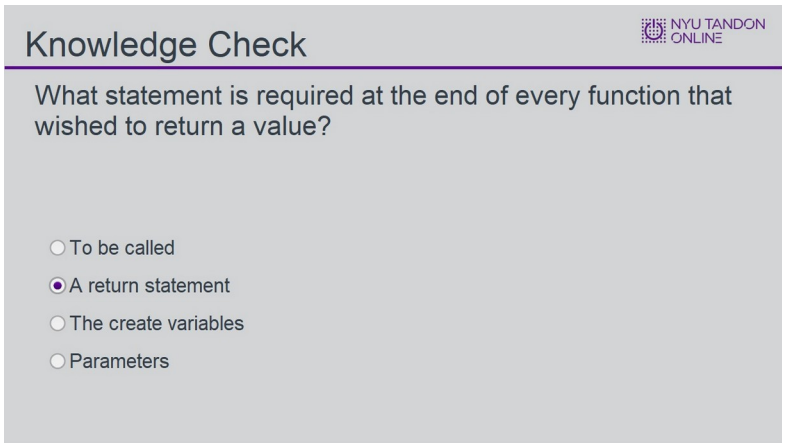


Concatenation (Slide Layer)



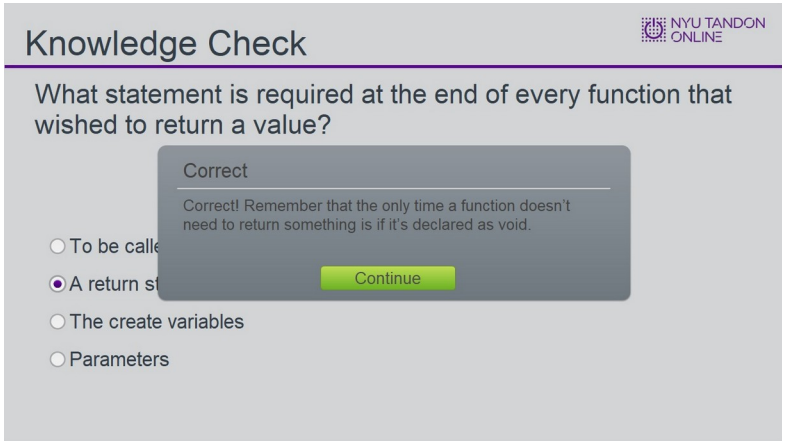
1.12 Knowledge Check

(Multiple Choice, 10 points, 4 attempts permitted)

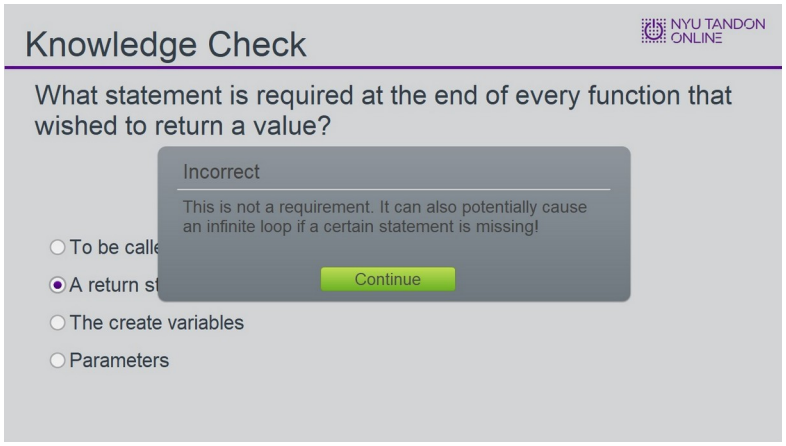


Correct	Choice	Feedback
	To be called	This is not a requirement. It can also potentially cause an infinite loop if a certain statement is missing!
X	A return statement	Correct! Remember that the only time a function doesn't need to return something is if it's declared as void.
	The create variables	Not quite! Your function can use the parameters passed in to do all the work required without ever creating new variables.
	Parameters	Not quite! Parameters are inserted before a function even begins. If you wish to save changes to a parameter, a specific keyword is needed at the end of a function.

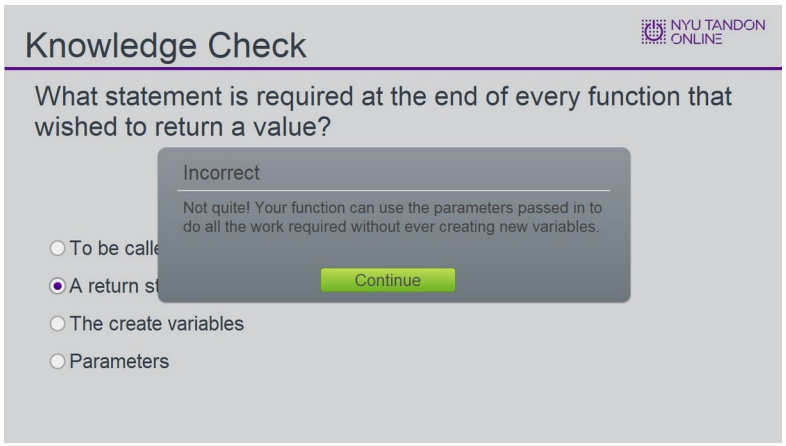
A return statement (Slide Layer)



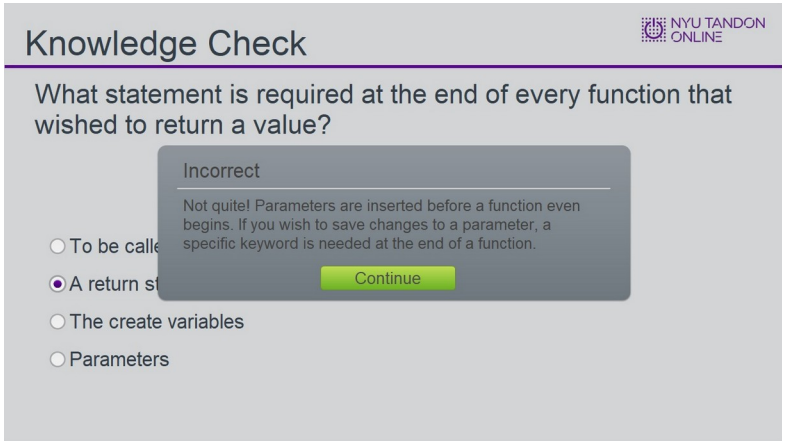
To be called (Slide Layer)



The create variables (Slide Layer)



Parameters (Slide Layer)



1.13 For the following piece of code, what are the values of a and b after you exit the code?

int a = 1;

int b = 15;

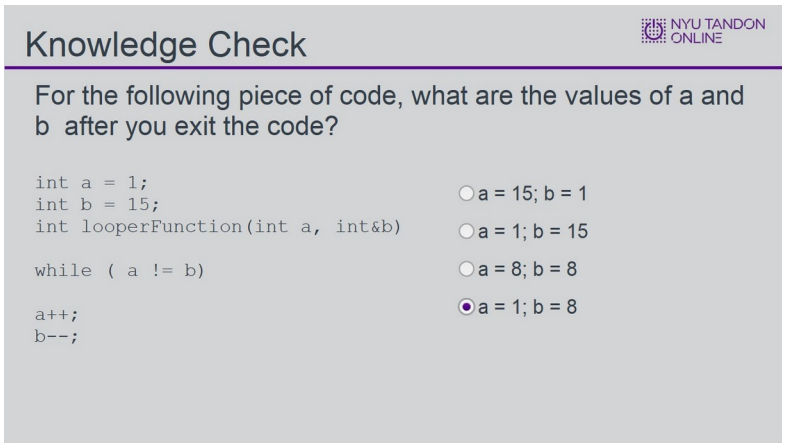
int looperFunction(int a, int&b)

while (a != b)

a++;

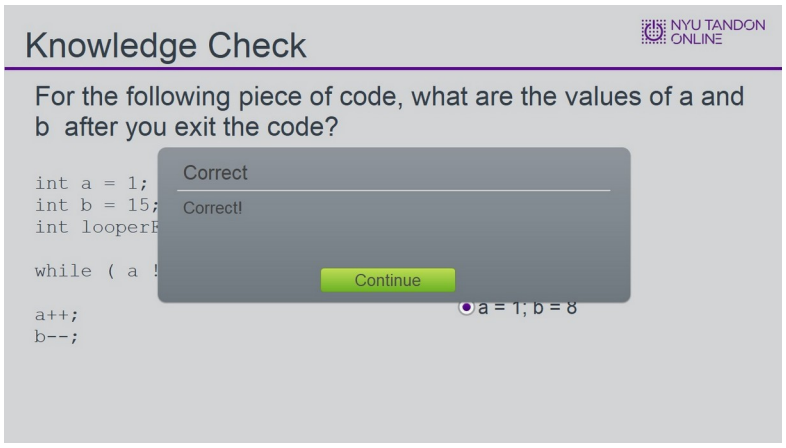
b--;

(Multiple Choice, 10 points, 4 attempts permitted)

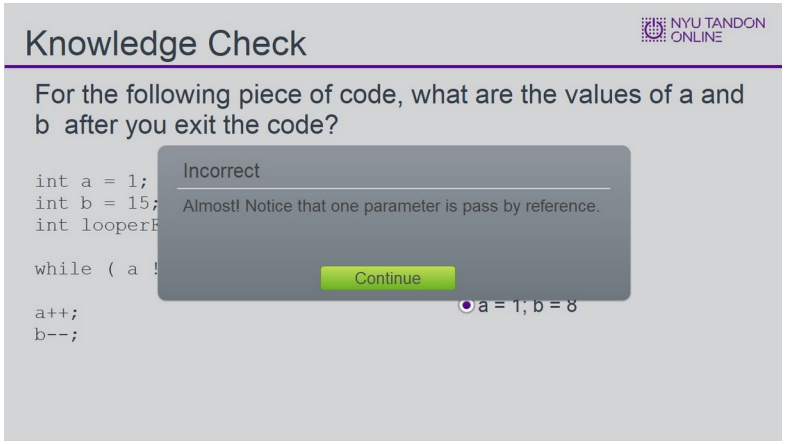


Correct	Choice	Feedback
	a = 15; b = 1	Not quite!
	a = 1; b = 15	Not quite!
	a = 8; b = 8	Almost! Notice that one parameter is pass by reference.
X	a = 1; b = 8	Correct!

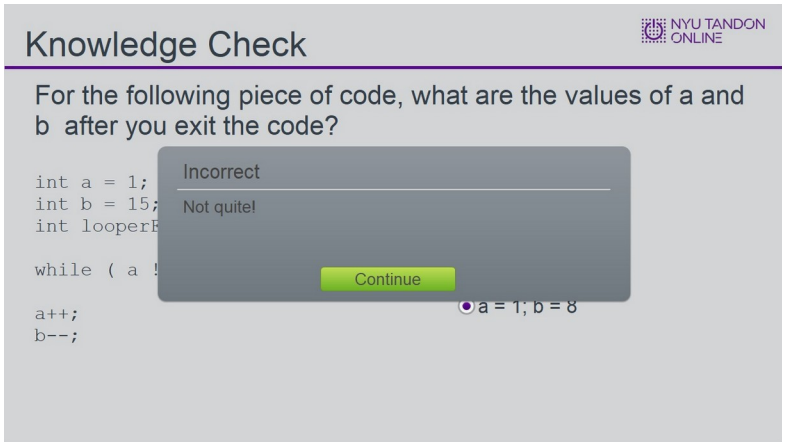
a = 1; b = 8 (Slide Layer)



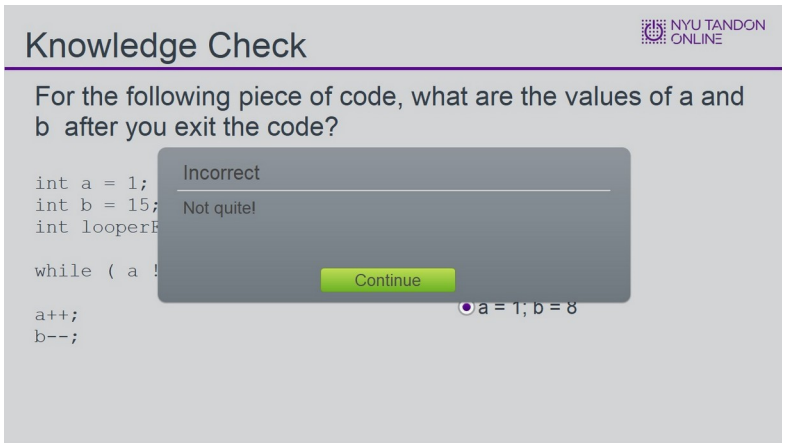
a = 8; b = 8 (Slide Layer)



a = 1; b = 15 (Slide Layer)

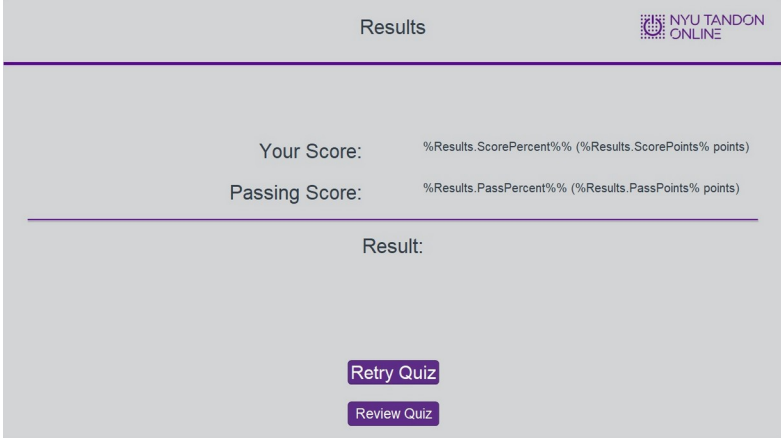


a = 15; b = 1 (Slide Layer)



1.14 Results Slide

(Results Slide, 0 points, 1 attempt permitted)



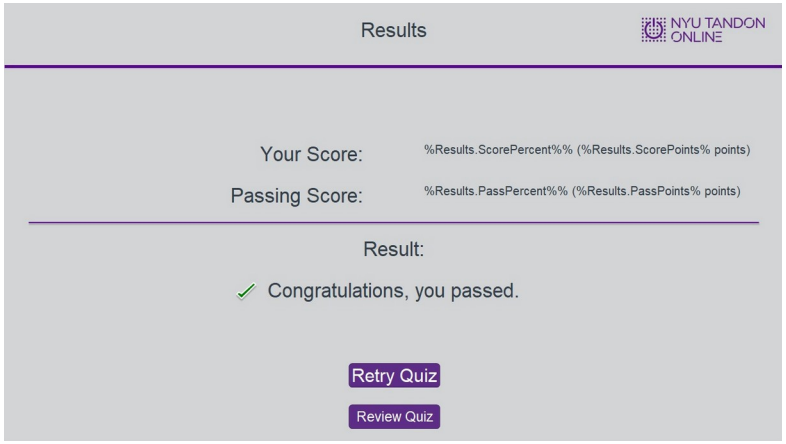
Results for
1.9 Knowledge Check
1.10 Knowledge Check
1.11 Knowledge Check
1.12 Knowledge Check

1.13 For the following piece of code, what are the values of a and b after you exit the code?
 	
int a = 1;
 	
int b = 15;
 	
int looperFunction(int a, int&b)
 	
while (a != b)
 	
a++;
 	
b--;

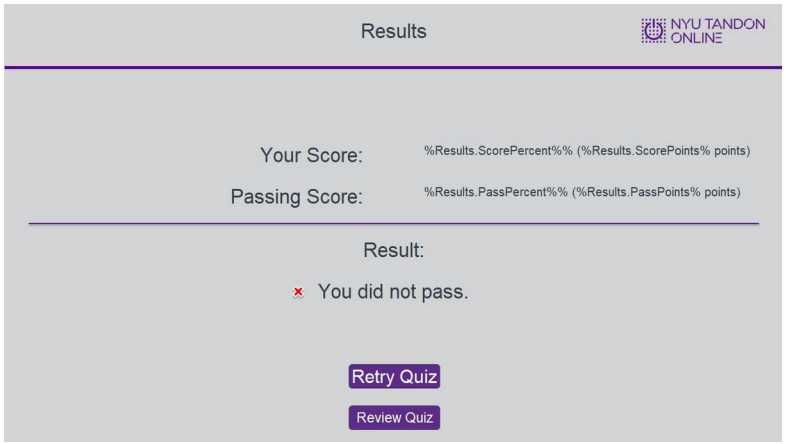
Result slide properties

Passing Score	80%

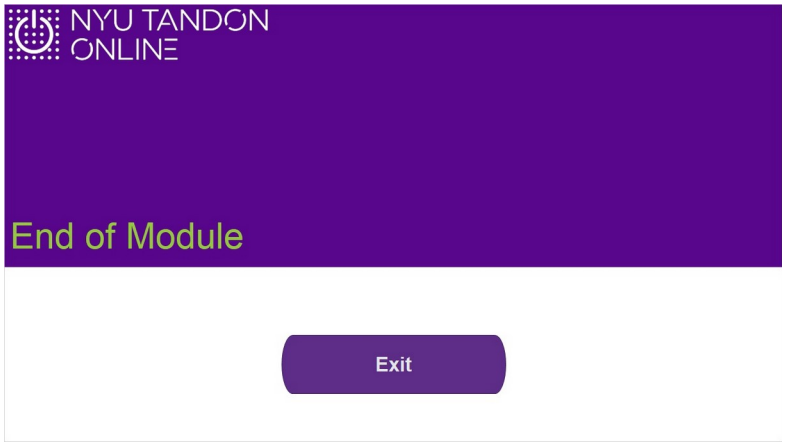
Success (Slide Layer)



Failure (Slide Layer)

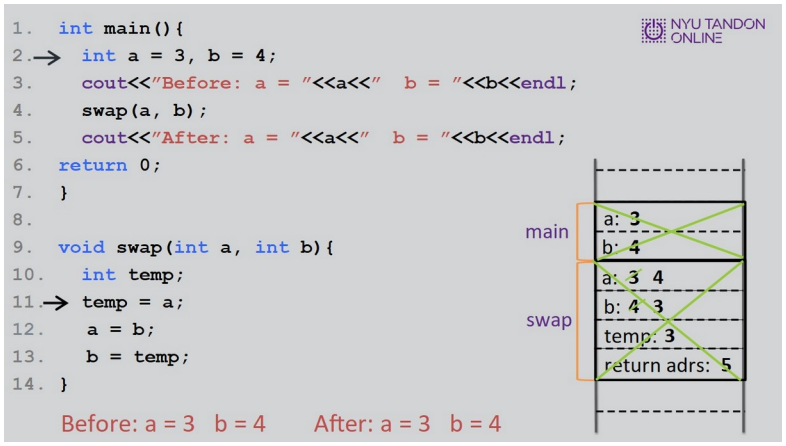


1.15 End of Module

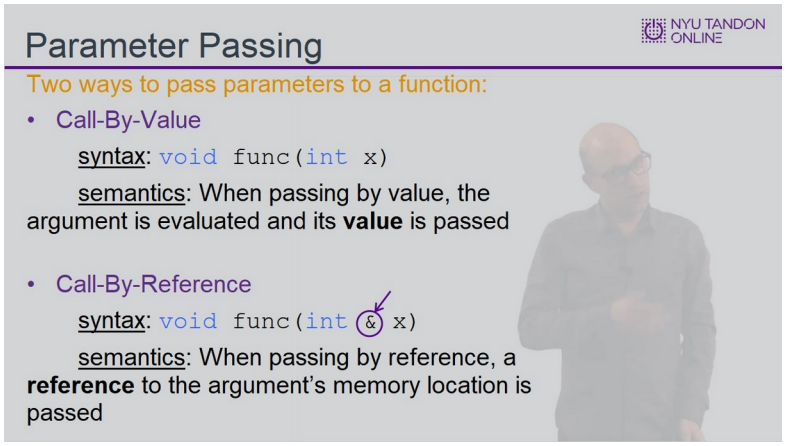


3. Pass by Value & Pass by Reference

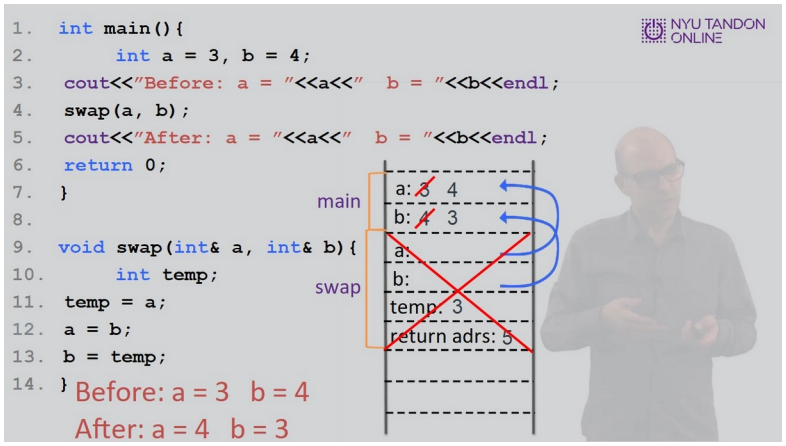
3.1 Pass By Value



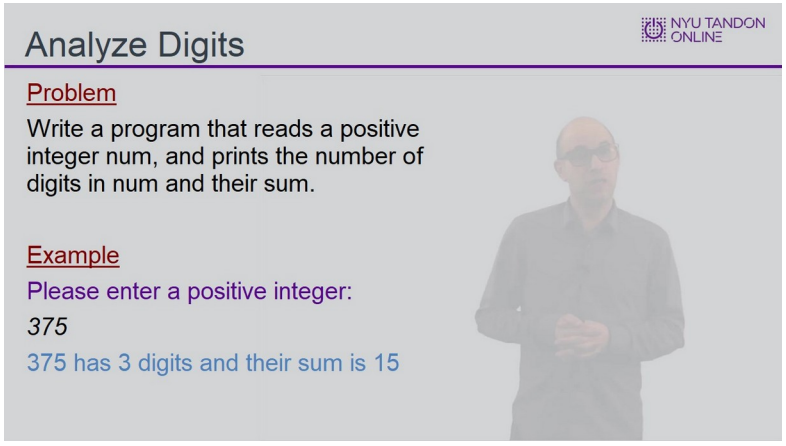
3.2 Parameter Passing



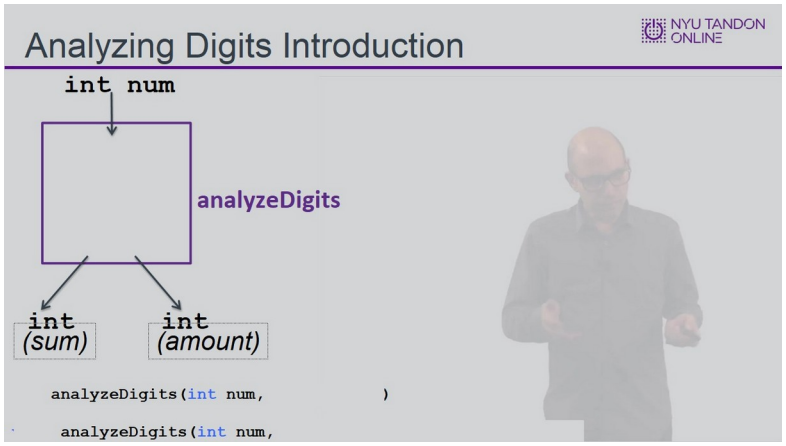
3.3 Example



3.4 Analyze Digits



3.5 Analying Digits Introduction



3.6 Analyze Digits Screencast