1. Write a template function named swap that has two parameters of the same type. The function swaps the value of two data types, test with float and integer.
   1. Change the types to pass by reference
   2. Change the types to pass by address
2. Create the C++ Function Template named multiples so that it has

three parameters sum, x, and n. The first two parameters will have

the type represented by the function template type parameter WhatKind.

n will always be int. The return type is void. All parameters are

passed by value except for sum which is passed by reference. A

Template Function created from multiples will compute...

sum = 1 + x + 2x + 3x + ... + nx

1. Create the C++ Function Template named init so that it has three

parameters whose types are determined by the function template type

parameters T1 and T2. The function header is shown below. init sets

the value of the parameter start to a T2-type value of 1. init

returns a T1-type value which is the sum of num1 and num2.

T1 init (T1 num1, T1 num2, T2& start)

Create a template function for the QuickSort algorithm (<http://en.wikipedia.org/wiki/Quicksort>)

template<typename T>

T\* quicksort(T\* array)