1.

a. This one will NOT generate error but declaring a variable (pointer or not) without initializing will assign garbage value to the variable and may cause logic error later in the program.

b. This will generate compile time error as the two pointers have different types (long vs double). To correct this error, we need to (1) change one of the two to match the type (either both long or both double) or (2) change one of them to be \*void or (3) cast the type in the assignment e.g. integerPtr = (int\*)realPtr; .

c. This will generate compile time error since x is int pointer whereas y is int and we cannot assignment int to int\* variable. We can declare both as int or both as int\* e.g. int \*x, \*y; .

d. Compile time error will be generated as s is declared built-in array, to correct this, we should declare s as pointer (built-in array is essentially pointer)

char \*s{“this is a char array”}; .

e. Compile time error since genericPtr is void\* and cannot be used directly in the assignment by dereferencing. We can either declare the genericPtr as int\* or in the assignment we cast it to int pointer.

f. xPtr is not declared as pointer as is initialized with the address & notation, this will generate compile time error, we should declare xPtr this way: double \*xPtr{&x}; .

2.

The function mystery1 is essentially concatenating two strings, or more precisely, it appends s2 to s1. It first loop through s1 and make the pointer pointing to the end of the string ‘\0’ and then uses a for loop to assign each char of s2 to the end of s1 one by one. And the main program asks for input for two strings then concat them and eventually print the combined string. For example, if s1 is entered as “aaa” and s2 is “bbb” then the main program will print “aaabbb”.

3.

The function mystery2 will count how many chars are there in the string passed to the function and return the number which is the length of the string. For example, if the string passed is “abc”, the function will return 3 and 3 will be printed in the main function.

4.

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

5.

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述