**EE2310 C++程式設計 HW 10 (Chapter 14) due: 6/16/2019**

**Part1, True/False是非題(30%, 5% for each)**

1. True/False: When a recursive function directly calls itself, this is known as direct recursion.

T

1. True/False: Indirect recursion means that a function calls itself several times.

F

1. True/False: A recursive function cannot call a function other than itself.

F

1. True/False: Recursive algorithms tend to be less efficient than iterative algorithms.

T

1. True/False: Any algorithm that can be coded with recursion can also be coded using a loop.

T

1. True/False: The speed and amount of memory available to modern computers diminishes the performance impact of the overhead of recursion so much that for many applications, this overhead is not noticeable.

T

**Part2, Choice選擇題(70%, 5% for each)**

1. A \_\_\_\_\_\_\_\_ function is one that calls itself

A) dynamic

B) static

C) recursive

D) data validation

E) None of the above

C

1. Recursion can be used to \_\_\_\_\_\_\_\_.

A) compute factorials

B) find the greatest common divisor of two integers (GCD)

C) program things that cannot be programmed without recursion

D) All of the above

E) Both A and B, but not C

E

1. The \_\_\_\_\_\_\_\_ algorithm uses recursion to sort an array.

A) shell sort

B) quicksort

C) binary sort

D) red/black sort

E) None of the above

B

1. A recursive function that does not correctly handle its base case may \_\_\_\_\_\_\_\_.

A) return 0 and stop

B) return FALSE and stop

C) cause an infinite chain of recursive calls

D) reach the NULL terminator and stop

E) None of the above

C

1. The \_\_\_\_\_\_\_\_ of recursion is the number of times a recursive function calls itself.

A) level

B) breadth

C) type

D) depth

E) None of the above

D

1. The quicksort algorithm works on the basis of \_\_\_\_\_\_\_\_.

A) three sublists

B) two sublists and a pivot

C) two pivots and a sublist

D) three pivots

E) None of the above

B

1. The programmer must ensure that a recursive function does not become \_\_\_\_\_\_\_\_.

A) a static function

B) a prototyped function

C) trapped in an infinite chain of recursive calls

D) a dynamic function

E) None of the above

C

1. A recursive function should be designed to stop making recursive calls when it reaches its \_\_\_\_\_\_\_\_.

A) return statement

B) base case

C) closing curly brace

D) last parameter

E) None of the above

B

1. When a function *A* calls a function *B*, which in turn calls *A*, we have \_\_\_\_\_\_\_\_.

A) direct recursion

B) indirect recursion

C) function call cycling

D) perfect recursion

E) None of the above

B

1. Suppose that a recursive function with integer parameter n has a base case of 0, and for each non-base case, the function makes a recursive call with argument n+1. If the function is initially called with an actual argument of n = 3, the function call will \_\_\_\_\_\_\_\_.

A) cause an infinite chain of recursive calls

B) will return after a chain of 2 recursive calls

C) will return after a chain of 3 recursive calls

D) will return after a chain of 4 recursive calls

E) None of the above

A

1. The QuickSort algorithm was developed in 1960 by \_\_\_\_\_\_\_\_.

A) Bjarne Stroustrup

B) Tony Gaddis

C) C.A.R. Hoare

D) Judy Walters

E) None of the above

C

1. The function

int fact(int k)

{

return k\*fact(k-1);

if (k==0) return 1;

}

A) computes the factorial on an integer k passed to it as parameter

B) returns the value 1 if it is passed a value of 0 for the parameter k

C) does not correctly handle its base case

D) works for all non-negative values of k, but not for negative numbers

E) None of the above

C

1. The quicksort algorithm can be used to \_\_\_\_\_\_\_\_.

A) sort lists stored in arrays

B) perform binary search on arrays

C) quickly sort and search arrays

D) All of the above

E) None of the above

A

1. The base case of a recursive function \_\_\_\_\_\_\_\_.

A) is 0

B) is 1

C) is depth / 2

D) is 1 / (depth \* 3.1415)

E) depends on the problem being solved

**E**