NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING

FINAL ASSESSMENT FOR
Semester 1 AY2020/2021

CS1010: PROGRAMMING METHODOLOGY

November 24, 2020 Time Allowed: 120 Minutes

INSTRUCTIONS TO CANDIDATES

1. Note that this paper was conducted online on LumiNUS and this version is for reference purposes only. As such, while all the questions are reproduced here. They may not be presented in their original form.
2. This assessment contains 14 questions. There are 18 printed pages, including this one.
3. Complete each question by filling in the given input fields.
4. The total marks for this assessment is 60. Answer ALL questions.
5. This is an OPEN BOOK assessment. You are only allowed to refer to any printed or written materials. References to online materials/digital documents are not allowed.
6. You can assume that in all the code given, no overflow nor underflow will occur during execution. In addition, unless otherwise specified, you can assume that all given inputs are valid and will fit within the specified variable type.
7. For essay-based questions, state any additional assumptions that you make.

STUDENT NUMBER: _____

EXAMINER'S USE ONLY		
Question	Mark	Score
1		
2		
3		
4		
5		
6		
7		

EXAMINER'S USE ONLY		
Question	Mark	Score
8		
9		
10		
11		
12		
13		
14		
TOTAL		

1. (3 marks) Consider the code fragment below:

```
long foo(long n) {
  long res = 1;
  for (long i = 1; i <= n; i += 1) {
    if (n % i == 0) {
      res *= n;
    }
  }
  return res;
}
```

Given that n is always a positive integer, which of the following functions (goo, hoo, joo, koo) are equivalent to foo above?

Note that for this question, two functions are equivalent if they always return the same value for the same input n.

[MRQ – Pick zero or more options that are correct]

Option 1

```
long goo(long n) {
  long res = n;
  long i = n;
  do {
    if (n % i == 0) {
      res *= n;
    }
    i -= 1;
  } while (i >= 1);
  return res;
}
```

Option 2

```
long hoo(long n) {
  long res = 1;
  long i = n;
  do {
    if (n % i == 0) {
      res *= n;
    }
    i -= 1;
  } while (i > 1);
  return res;
}
```

Option 3

```
long jj(long n, long i) {
  if (i == 1) {
    return n;
  }
  if (n % i == 0) {
    return jj(n * n, i - 1);
  }
  return jj(n, i - 1);
}

long joo(long n)
  return jj(n , n);
}
```

Option 4

```
long kk(long n, long i) {
  if (i == 1) {
    return n;
  }
  if (n % i == 0) {
    return n * kk(n, i - 1);
  }
  return kk(n, i - 1);
}

long koo(long n) {
  return kk(n , n);
}
```

Answer: _____

2. (3 marks) Consider the running time of three functions below using big O notation:

```
long bar(long n) {
  long k = 0;
  for (long i = 1; i < n*n; i += n) {
     k += i;
  }
  return k;
}

long cat(long n) {
   long k = 0;
   for (long i = n; i > 0; i /= 2) {
     k += i;
   }
   return k;
}

long dam(long n) {
   long k = 0;
   for (long i = 1; i < 1000; i += 1) {
     k += n;
   }
   return k;
}
```

Arrange the three functions, in increasing order of growth, in terms of their big O time complexity. A function with a higher order of growth should be listed later.

[MCQ – Pick one option]

Option 1
bar, cat, dam

Option 2
cat, bar, dam

Option 3
dam, cat, bar

Option 4
all three functions have the same order of growth

Option 5
none of the above

Answer: _____

3. (3 marks) Consider the following program.

```
char * foo(char *x) {
  x = "cs1010";
  // Line A
  return x;
}

int main() {
  char *c = malloc(4);
  c = foo(c);
  // Line B
}
```

While of the following statements are correct?

[MRQ – Pick zero or more options that are correct]

Option 1
At Line A, the pointer x is pointing to a memory location on the stack.

Option 2
At Line A, the pointer x is pointing to a memory location on the heap.

Option 3
At Line A, the pointer c is pointing to a memory location on the heap.

Option 4
At Line B, the pointer c is pointing to a memory location on the stack.

Option 5
At Line B, the pointer c is pointing to a memory location on the heap.

Answer: _____

4. (3 marks) Consider the function below:

```
void qux() {
  char c;
  while (c > 0) {
    c = 1;
  }
}
```

Which of the following statements about the function above is true?

[MRQ – Pick zero or more options that are correct]

Option 1
The code causes a compilation error because -= can only be used with integer types.

Option 2
The code causes a compilation error because c has not been initialised.

Option 3
The code causes a compilation error because we did not cast 0 and 1 to char.

Option 4
The code may lead to an infinite loop.

Option 5
The code will always lead to an infinite loop.

Answer: _____

5. (3 marks) Consider the following:

```
#define distance_square(a, b) (a*a + b*b)
```

What is the value of x after executing

```
long a = 1;
long b = 2;
long x = distance_square(a + 2, b + 3);
```

[Fill in the blank]

Answer: _____

6. (3 marks) What is printed by the following program?

```
#include "cs1010.h"

long foo(long a, long *b, long **c) {
  **c += 1;
  *b += 2;
  a += 4;
  return a + *b + **c;
}

int main() {
   long p;
   long *q;
   long **r;
   p = 0;
   q = &p;
   r = &q;
   cs1010_println_long(foo(p, q, r));
}
```

What is the output of the program?

[Fill in the blank]

Answer: _____

7. (3 marks)

```
#include "cs1010.h"
long f(long n) {
  if (n < 2) {
    return 1;
  }
  for (long i = 0; i < n; i += 1) {
    cs1010_println_string("hello");
  }
  return f(n/3) + f(n/3) + f(n/3);
}
```

What is the time complexity of the above code?

[MCQ – Pick one option]

Option 1
O(log n)

Option 2
O(n)

Option 3
O(n log n)

Option 4
$O(n^2)$

Option 5
$O(n^3)$

Option 6
None of the above.

Answer: _____

8. (3 marks)

```
#include "cs1010.h"

void f(long n, long a) {
  if (n <= 0) {
    return;
  }
  f(n - 1, a + 1);
  cs1010_println_long(a);
  f(n - 1, a - 1);
}

int main() {
  f(3, 4);
}
```

What is the output of the program above?

[Fill in the blanks]

Answer:

9. (3 marks) Consider the program below:

```
int main() {
    long *p = bar();
    cs1010_println_long(*p);
}
```

Which implementation of function bar below would lead to illegal memory access when the program is executed? Assume that malloc does not return NULL.

[MRQ: pick zero or more options that are correct]

Option 1

```
long *bar() {
   long x = 10;
   return &x;
}
```

Option 2

```
long *bar() {
   long *px;
   *px = 10;
   return px;
}
```

Option 3

```
long *bar() {
   long *px;
   px = (long *)malloc(1);
   *px = 10;
   return px;
}
```

Option 4

```
long *bar() {
   return (long *)10;
}
```

Answer: _____

10. (3 marks) Consider the following (incomplete) function, which merges two sorted arrays a1 and a2 of size len1 and len2 respectively into the output array out of size len1 + len2. The arrays a1 and a2 may contain duplicate values and are sorted in non-decreasing order.

```
void merge(const long a1[], const long a2[], long out[], long
len1, long len2) {
  long i = 0;
  long j = 0;
  long k = 0;
  while (i < len1 && j < len2) {
    if (a1[i] < a2[j]) {
      out[k] = a1[i];
      i += 1;
    } else {
      out[k] = a2[j];
      j += 1;
    }
    k += 1;
  }
  // Line C
}
```

Which of the following assertions must be true at Line C?

[MRQ: pick zero or more options that are correct]

Option 1

```
i != len1 ||  j < len2
```

Option 2

```
i != len1 || k == len1 + j - 1
```

Option 3

```
i != len1 || a1[len1 - 1] < a2[j]
```

Option 4

```
i != len1 || out[k] == a1[len1 - 1]
```

Answer: _____

11. (3 marks) Consider the following code snippet, which defines a structure called box and four functions, titi, tutu, tata, and toto.

```
struct box {
    long x;
  };

  void titi(struct box b) {
    b.x = 1;
  }

  void tutu(struct box *b) {
    b->x = 1;
  }

  struct box tata() {
    struct box b;
    b.x = 1;
    return b;
  }

  void toto(long *x) {
    *x = 1;
  }
```

Which of the following correctly initialise the field x in b to 1?

[MRQ: select zero or more options that are correct]

Option 1

```
struct box b;
titi(b);
```

Option 2

```
struct box b;
tutu(&b);
```

Option 3

```
struct box b = tata();
```

Option 4

```
struct box b;
toto(&(b.x));
```

Answer: _____

12. (3 marks) What is the output of this program?

```
#include "cs1010.h"

char* quack(char *input, int i, int j) {
  char* output = input;
  output[j] = output[i];
  output[i] = input[j];
  return output;
}

int main() {
  char input[] = "01234567";
  char* output = quack (
    quack (
      quack (
        quack (input, 3, 4),
      2, 5),
    1, 6),
  0, 7);
  cs1010_println_string(output);
}
```

[Fill in the blank]

Answer: _____

13. Study the function moo() below.

```
void swap(long a[], long i, long j) {
  long temp = a[j];
  a[j] = a[i];
  a[i] = temp;
}

void moo(long a[], long len) {
  long i = 0;
  while (i < len) {
    if (i == 0 || a[i] >= a[i - 1]) {
      i += 1;
    } else {
      swap(a, i, i-1);
      i -= 1;
    }
    // Line Z
  }
}
```

(a) (4 marks) Suppose we call moo with the array a below as input.

```
long a[3] = {9, 8, 7};
moo(a, 3);
```

Show the value of i and the content of the array in the first six iterations of the while loop in moo at Line Z. The answers for the 1st and 6th iteration are:

1st Iteration: i is 1, a is {9, 8, 7}
6th Iteration: i is 0, a is {7, 8, 9}

Specify the values for i and a in iterations 2nd, 3rd, 4th, and 5th.

[Fill in the blanks]

Answer:

(b) (1 marks) What is the content of array a when moo returns?

[Fill in the blank]

Answer: _____

(c) (4 marks) The function moo above is reproduced below with additional commented lines.

```
void moo(long a[], long len) {
  long i = 0;
  while (i < len) {
    if (i == 0 || a[i] >= a[i - 1]) {
      // Example: { a[i] >= a[i - 1] }
      i += 1;
      // Line V
    } else {
      // Line W
      swap(a, i, i-1);
      // Line X
      i -= 1;
      // Line Y
    }
  }
}
```

Write down the assertions that must be true at Line V, W, X, Y. Your assertion should relate either a[i] or a[i-1] to one of its adjacent elements in the array (i.e., relate a[i] to either a[i-1] or a[i+1], or relate a[i-1] to either a[i] or a[i-2]).

For example, the assertion on Line 5 above is a[i] >= a[i-1]

[Fill in the blanks]

Answer:

Line V: _____

Line W: _____

Line X: _____

Line Y: _____

(d) (3 marks) The function moo above is reproduced below with comments Line P and Line Q.

```
void moo(long a[], long len) {
  long i = 0;
  while (i < len) {
    // Line P
    if (i == 0 || a[i] >= a[i - 1]) {
      i += 1;
    } else {
      swap(a, i, i-1);
      i -= 1;
    }
    // Line Q
  }
}
```

The loop invariant for the while loop in moo is as follows:

The elements in a[0] .. a[i-1] (if any) are sorted in non-decreasing order.

Explain why, if this invariant is true at Line P, it will also be true at Line Q.

Answer:

(e) (2 marks) Suppose the input array is already sorted in non-decreasing order, what is the running time of the function moo? Express your answer using Big O notation in its simplest form in terms of n, the number of elements in the input array. Explain how you derived your answer.

Answer:

(f) (4 marks) Suppose the input array is inversely sorted in decreasing order, what is the running time of the function moo? Express your answer using Big O notation in its simplest form in terms of n, the number of elements in the input array, and explain how you derived your answer.

Answer:

14. (6 marks) Uncle wants to write a recursive function to help him count how many ways he can sell n masks. It takes n as input and returns the number of ways to sell n masks. Help him complete the function by filling in the body of the function (you do not have to write the function header again in your answer).

A non-recursive function will receive 0 marks. Since Uncle Tan does not sell many masks each day, you do not have to worry about (I) stack overflow, (ii) integer overflow, and (iii) the efficiency of your program.

Answer:

```
long masks(long n) {



}
```

THIS IS THE END OF THE PAPER