

NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING

**EXAMINATION FOR
Semester 1 AY2013/2014**

CS1010 – PROGRAMMING METHODOLOGY

November 2013

Time allowed: 2 hours

INSTRUCTIONS TO CANDIDATES

1. This examination paper consists of **SIX (6)** questions and comprises **TEN (10)** printed pages.
2. This is an **OPEN BOOK** examination.
3. Calculators and electronic dictionaries are not allowed.
4. Answer all questions, and write your answers in the ANSWER SHEETS provided.
5. Fill in your Matriculation Number with a pen, clearly on every page of your ANSWER SHEETS.
6. You may use **2B pencil** to write your codes. Pen is preferred for other questions.
7. Note the penalty will be given for codes that are unclear or unnecessarily long.
8. Note that question 6 is a long question so do set aside enough time for it.
9. You must submit only the ANSWER SHEETS and no other document.

Q1. Multiple Choice Questions (MCQs)**[12 marks]**

Each MCQ has one correct answer and is worth 2 marks. There is no penalty for wrong answer.

Q1.1 What is the output of the following C code fragment?

```
int a = 9;
if (isdigit(a))
    printf("true\n");
else
    printf("false\n");
```

- A. `true`
- B. `false`
- C. No output will be printed.
- D. It will give an error.

Q1.2 What is the correct way to assign values to members of structure variable `tray`, given the following code fragment?

```
typedef struct {
    int length, width;
} tray_t;
tray_t tray;
```

- A. `tray->length = 12; tray->width = 12;`
- B. `tray = {12, 12};`
- C. `tray.length = tray.width = 12;`
- D. `tray = 12;`
- E. None of the above.

Q1.3 Which of the following lists will take the most number of passes to complete sorting in ascending order, using the selection sort algorithm given in the lecture notes?

- A. `int list[] = {3, 4, 1, 2};`
- B. `int list[] = {4, 1, 2, 3};`
- C. `int list[] = {1, 2, 3, 4};`
- D. `int list[] = {4, 3, 2, 1};`
- E. The above 4 lists will take an equal number of passes for sorting.

Q1.4 Assuming that **n** is a positive integer, consider the following four functions:

```
int f1(int n) {
    int a, sum = 0;
    for (a=1; a<=n; a++)
        sum += a;
    return sum;
}
```

```
int f2(int n) {
    int sum = 0;
    while (n > 0) {
        sum += n;
        n--;
    }
    return sum;
}
```

```
int f3(int n) {
    if (n==1)
        return n;
    else
        return n + f3(n-1);
}
```

```
int f4(int n) {
    return n*(1+n)/2;
}
```

Which of the following statements is true?

- A. Given a positive **n**, **f1** and **f2** will return different values.
- B. Given a positive **n**, **f1** and **f3** will return different values.
- C. Given a positive **n**, **f2** and **f4** will return different values.
- D. Given a positive **n**, **f3** and **f4** will return different values.
- E. Given a positive **n**, all the four functions will return the same value.

Q1.5 Given an array `arr` of `n` distinct integers ($n > 0$), what does the following function call `what(arr, 0, n-1)` return?

```
int what(int arr[], int start, int end) {
    if (start == end)
        return end;
    else if (arr[start] > arr[end])
        return what(arr, start, end-1);
    else
        return what(arr, start+1, end);
}
```

- A. The minimum value in array `arr`.
- B. The maximum value in array `arr`.
- C. Index of the minimum value in array `arr`.
- D. Index of the maximum value in array `arr`.
- E. None of the above.

Q1.6 What is the output of the following C program?

```
#include <stdio.h>

int main(void){
    int a, b, c, d;

    b = 0;
    c = 9;
    d = (a = 6/9 || b++ || ++c) ? b : c;
    printf("%d %d %d %d\n", a, b, c, d);
    return 0;
}
```

- A. 1 1 10 1
- B. 0 1 10 1
- C. 1 1 9 1
- D. 0 1 9 9
- E. None of the above.

Q2. [Total: 13 marks]**Q2.1** Suppose a text file “**numbers.txt**” contains the following values:

[2 marks]

2 3 0 6 -5 8

What is the output of the following C code fragment?

```
FILE *infile;
int num, sum=0, count=0;

infile = fopen("numbers.txt", "r");
fscanf(infile, "%d", &num);
while (num > 0) {
    sum += num;
    count++;
    fscanf(infile, "%d", &num);
}
printf("%.2f\n", (double)(sum/count));
```

Q2.2 What is the output of the following program?

[2 marks]

```
#include <stdio.h>

typedef struct {
    char code[10];
    int numStu;
} module_t;

void f(module_t);

int main(void) {
    module_t list[] = { {"CS1010", 300}, {"CS1231", 100} };

    f(list[1]);
    printf("%s %d\n", list[1].code, list[1].numStu);
    return 0;
}

void f(module_t m) {
    --m.numStu;
}
```

Q2.3 What is the output of the following program?

[2 marks]

```
#include <stdio.h>
#define N 5

int main(void) {
    int i, a[N] = {5, 4, 3, 2}, b[N];

    for (i=0; i<N; i++)
        b[i] = a[(i+3)%5];

    for (i=0; i<N; i++)
        printf("%d ", b[i]);
    printf("\n");

    return 0;
}
```

Q2.4 Study the program below and answer the questions below.

[3 marks]

```
#include <stdio.h>
#include <string.h>

int main(void) {
    char fruits[][20] = { "mango", "apple" };
    char str[6] = "steen";
    int len1 = strlen(fruits[1]);
    int len2 = strlen(str);
    int i;

    for (i = 0; i < len2; i++)
        fruits[1][len1 + i] = str[i];
    fruits[1][len1 + i] = '\0';

    puts(fruits[1]);

    return 0;
}
```

- (a) What is the output of the above program?
- (b) Replace the boxed code by a single call to a certain string function.

Q2.5 What is the output of the following program?

[4 marks]

```
#include <stdio.h>

void swap(int *, int *);

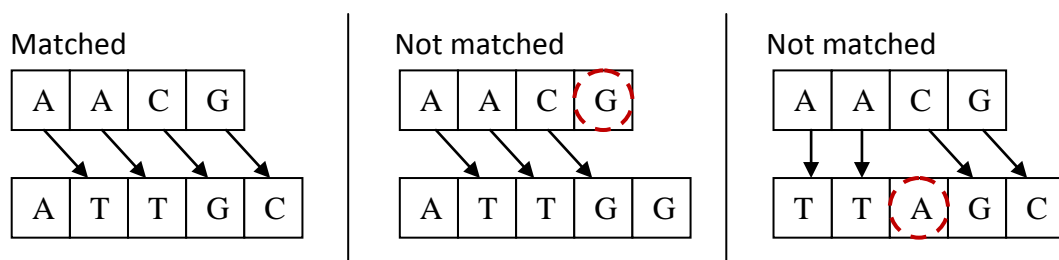
int main(void) {
    int a = 4, b = 7, *ptr;

    ptr = &b;
    swap(&a, ptr);
    printf("%d %d\n", a, b);

    return 0;
}

void swap(int *p, int *q) {
    *p += *q;
    q = p;
    printf("%d %d\n", *p, *q);
}
```

Q3. A DNA sequence is a string of four possible characters 'A', 'T', 'C' and 'G'. Among these four characters, 'A' and 'T' match with each other and so do 'C' and 'G'. Given two DNA sequences *seq1* and *seq2*, each has at most 20 characters, *seq1* matches with *seq2* if all the characters in *seq1* can be matched with some characters in *seq2* consecutively. For example, the sequence "AACG" matches with the sequence "ATTGC" but not with "ATTAGG" or "TTAGC", as explained below.



Write a function

```
int match(char seq1[], char seq2[])
```

that takes in two strings *seq1* and *seq2* and returns 1 if *seq1* matches with *seq2*, or 0 otherwise. You may make use of any character and string functions as necessary. You are not allowed to modify *seq1* or *seq2* in this function. [6 marks]

- Q4.** Do you remember how you performed addition of two numbers when you were in primary school? For example, if given two integers 4673 and 5239:

$$\begin{array}{r} 4673 \\ + 5239 \\ \hline 9912 \end{array}$$

You would start from the right hand side (least significant digit) and add 3 and 9, to get 12. So you put 2 and carry 1 to the next column. For the next column, you would add 7 and 3 and add the carry of 1 to get 11, and so on.

Write a program to simulate the above addition process. The program reads in two positive integers (*you may assume both do not contain leading zeroes and both are equal in length*), and calls a recursive function **add()** to do the addition. This function takes in three integers as parameters (the first two representing the two numbers being added and the third representing the carry from the previous column), and returns the sum.

A partially completed program is given on the Answer Sheet. You have to complete the function call in the main function and complete the recursive function **add()**.

A sample run is given below with user input highlighted in bold.

```
Enter two positive integers: 4673 5239
Sum = 9912
```

[8 marks]

- Q5.** A square black-and-white image can be represented as an $N \times N$ array of 1s (black pixels) and 0s (white pixels). An image can be manipulated in many ways, two of which are flip and rotate. When an image is flipped, it becomes upside-down. When an image is rotated, it is rotated 90 degrees clockwise. Figure (a) shows a sample image where $N = 5$, while (b) and (c) show the resulting image of flipping and rotating the sample image respectively.

	0	1	2	3	4
0	0	0	0	0	0
1	0	1	1	1	0
2	0	0	1	0	0
3	0	0	1	0	0
4	0	0	0	0	0

	0	1	2	3	4
0	0	0	0	0	0
1	0	0	1	0	0
2	0	0	1	0	0
3	0	1	1	1	0
4	0	0	0	0	0

	0	1	2	3	4
0	0	0	0	0	0
1	0	0	0	1	0
2	0	1	1	1	0
3	0	0	0	1	0
4	0	0	0	0	0

(a) Sample image

(b) Sample image flipped

(c) Sample image rotated

Write two functions:

```
void flip(int img[][MAX_SIZE], int size)
```

and

```
void rotate(int img[][MAX_SIZE], int size)
```

to perform these two operations on the $size \times size$ image *img*. Here *MAX_SIZE* is a constant that indicates the maximum possible size of the input image. The resulting image should be stored in *img*. [13 marks]

Q6. [Total: 28 marks]

You are to write a program to model the operations of a Football Association and its associated football clubs. For this question, you will be required to write parts of the program but not the entire program.

The Football Association regularly organizes tournaments for its football clubs. Each club has a **club-id**, a **home venue**, a **resident coach**, **team of players** and a **list of blackout dates**. The club-id is an integer, the home venue, coach and players are identified by their names (maximum length of 50 characters). Each club can have at most one coach and 20 players. The blackout dates indicate the dates that the club is unable to participate in the tournament. Each club can indicate at most 10 blackout dates. The blackout date is represented as a structure called **date_t**, as shown below:

```
typedef struct {
    int day, month, year;
} date_t;
```

For all functions you are asked to write, you are to figure out their parameters.

- a) Define a structure called **club_t** to keep track of the information related to a club. [3 marks]
- b) A tournament schedule consists of a series of matches. The tournament is conducted in a round-robin manner, such that a club **A** will have to play against the other clubs, once on home ground, i.e. home venue of **A**, and another at the opponent's home venue. Each match consists of the date of the match, the two clubs competing with each other, and the venue of the match. Define a structure called **match_t** to keep track of the details of each match. [3 marks]
- c) In order for the Association to find out if a particular date conflicts with the blackout dates of a certain club, you need to provide a function for this purpose. Complete the function **conflict()** that returns 1 if the date is in the list of blackout dates of the given club, or 0 otherwise. [5 marks]
- d) Assume the Association has 8 clubs under its care and their details are stored in an array called **FAclub** while tournament schedule is stored as the variable **tournament** which is declared as an array of elements of the type **match_t** as shown:

```
match_t tournament[56];
club_t FAclub[8];
```

Complete the function **numHomeGames()** that returns the number of matches that are played in the given club's home venue. [5 marks]

- e) A tournament is fair if each club plays the same number of matches in their own home venue. Write a function **isFair()** that returns 1 if the tournament is fair, or 0 otherwise. [6 marks]
- f) Write a function **sortSchedule()** to sort the schedule in increasing order of the date of the matches. You may assume that no two matches can occur on the same day, and you may define any other helper function. [6 marks]

=== END OF PAPER ===