TIC1001—Introduction to Computing and Programming
National University of Singapore
School of Continuing and Lifelong Education

Midterm Test

4 October 2018 **Time allowed:** 1 hour 30 mins

Student No:	**S**	**O**	**L**	**U**	**T**	**I**	**O**	**N**	**S**

Instructions (please read carefully):

1. Write down your **student number** on the question paper. DO NOT WRITE YOUR NAME ON THE ANSWER SET!
2. This is **an open-book test**. You are allowed to bring any non-electronic notes for reference.
3. This paper comprises **FIVE (5) questions** and **TEN (10) pages**. The time allowed for solving this test is **1 hour 30 mins**.
4. The maximum score of this test is **50 marks**. The marks for each question is given in square brackets beside the question number.
5. Note that the marks and order of the questions do not necessarily correspond to the level of difficulty of the question.
6. All questions must be answered correctly for the maximum score to be attained.
7. The pages marked "scratch paper" in the question set may be used as scratch paper.
8. You are allowed to un-staple the sheets while you solve the questions. Please make sure you staple them back in the right order at the end of the test.
9. You are allowed to use pencils, ball-pens or fountain pens, as you like as long as it is legible (no red color, please).

EXAMINER'S USE ONLY

Question	Marks
Q1	/ 15
Q2	/ 19
Q3	/ 6
Q4	/ 4
Q5	/ 6
Total	/ 50

ALL THE BEST!

This page is intentionally left blank.

It may be used as scratch paper.

Question 1: C/C++ Expressions [15 marks]

There are several parts to this question which are to be answered independently and separately. Each part consists of a fragment of C/C++ code. Write the exact output produced by the code in **the answer box**. If an error occurs, or it enters an infinite loop, state and explain why.

You may show workings **outside the answer box** in the space beside the code. Partial marks may be awarded for workings if the final answer is wrong.

Assume that all appropriate preprocessor directives e.g., `#include <iostream>`, etc. have already been defined.

A. [5 marks]

```
int f(int x, int y) {
    x *= 10;
    y += x;
    return y;
}

int main() {
    int x = 4, y = 7;
    cout << f(f(y, x), y);
}
```

```
747
```

B. [5 marks]

```
int a = 10;
if (a > 5)
    cout << "Here ";
if (a % 5)
    cout << "we ";
else if (a)
    cout << "go ";
else
    cout << "again";
```

```
Here go
```

C. [5 marks]

```
for (int i = 10; i > 0; i -= 2) {
    cout << i << endl;
    if (i % 4 == 0)
        break;
    if (i % 3 == 0) {
        i -= 3;
        continue;
    }
    i += 1;
}
```

```
10
9
4
```

Question 2: Dates [19 marks]

A Georgian calendar date can be presented as a 8-digit integer in the format `YYYYMMDD`, where `YYYY` is a 4-digit year, `MM` is a 2-digit month and `DD` a 2-digit day.

For example, the date 4th October 2018 is represented as `20181004`.

A. The function `int compare_dates(int d1, int d2)` takes two dates, $d1$ and $d2$ as inputs and compares them. It returns

- a negative number (< 0) if $d1$ is before $d2$,
- a positive number (> 0) if $d1$ is after $d2$,
- zero ($= 0$) if $d1$ and $d2$ are the same dates.

Provide an implementation for the function `compare`. [5 marks]

```
int compare(int d1, int d2) {
    return d1 - d2;

}
```

B. The function `int get_num_days(int month, int year)` takes a month and year as inputs, and returns the number of days in a month. For example, `get_num_days(10, 2018)` will return the value `31`.

Why does the function require both month and year as inputs? [2 marks]

Because of leap years. February has 28 or 29 days depending on the given year.

C. The function `void inc_date` takes a date as input, and increments the date by one day. For example, the following code segment

```
int today = 20181004;
inc_date(today);
cout << today << endl;
```

will display `20181005`.

Provide an implementation for the function `inc_date`. You may assume that the function `get_num_days` described in part B is given and correct, and that the date input to the function is always a valid date. [6 marks]

```
// fill in the parameters
void inc_date(int &date) {

    date += 1;

    int year = date / 10000;
    int month = (date / 100) % 100;
    int day = date % 100;

    // max days in current month
    int m_days = get_num_days(month);
    if (day > m_days) {
        day = 1;
        month += 1;
        if (month > 12) {
            month = 1;
            year += 1;
        }

        // Update the date
        date = year * 10000 + month * 100 + day;
    }

}
```

D. The function `int num_days_between(int d1, int d2)` takes two dates as input, and returns the number of days between the dates.

Provide an implementation for the function `num_days_between`. You may assume that $d1$ is no later than $d2$, and may reuse any function previously defined in this question. [6 marks]

```
int num_days_between(int d1, int d2) {
    int days = 0;
    while (d1 < d2) {
        inc_date(d1);
        days += 1;
    }
    return days;
}
```

Question 3: Computer Organization [6 marks]

Note that for the **multiple response questions** in this section, every suboption carries **equal** weightage. **Please *circle* your answer clearly**.

A. Which of the following is **TRUE** regarding the concept of **Von Neumann Architecture**?

i. The major components of a computing device is connected by BUS. [**T** / F]

ii. In modern laptops, the SSD (solid state drive) plays the role of the MEMORY. [T / **F**]

iii. The registers are storage inside the processor. [**T** / F]

iv. The executable compiled by Visual Studio Code, e.g. "task1.exe" is stored in the HARDDISK. [**T** / F]

[2 marks]

B. If you have a **quadcore** processor (i.e. 4-core), which of the following is **TRUE**?

i. A program written for single-core processor can now run up to **4-time** faster. [T / **F**]

ii. If each of the core runs at **1 GHz** (1 gigahertz), then the quadcore is running at **4 GHz**. [T / **F**]

iii. There can be up to four program running **at the same time** on this processor. [**T** / F]

iv. It is possible to get 4 times productivity (i.e. more work gets done) in the same amount of time on this processor. [**T** / F]

[2 marks]

C. Which of the following regarding instruction execution is **TRUE**?

i. Execution cycle of an instruction may take up to **three** memory accesses. [**T** / F]

ii. Memory instruction refers only to instruction that moves a value from memory to register. [T / **F**]

iii. Instruction and data are stored in the same region in memory. [T / **F**]

iv. Store-memory concept refers to the idea of storing computation results in the memory. [T / **F**]

[2 marks]

Question 4: Data Representation [4 marks]

Note that for the **multiple response questions** in this section, every suboption carries **equal** weightage. **Please *circle* your answer clearly**.

A. Which of the following is **TRUE** regarding the data storage in computer?

i. On a 32-bit platform, each *double* variable in C/C++ needs 64 bit storage in memory. [**T** / F]

ii. If a 4-byte word is used to store only **non-negative** values, then its range is $[0...2^{32}-1]$ [**T** / F]

iii. Directly comparing floating point values / variables is not safe because of the **overflowing** problem. [T / **F**]

iv. Computer can store either binary, octal (base-8) or hexadecimal (base-16) values in the actual hardware. [T / **F**]

[2 marks]

B. Which of the following **number base** related statements is **TRUE**?

i. The decimal number 58_{10} is 10_{58} in base-58. [**T** / F]

ii. The decimal number 58_{10} is 110010_2 in base-2. [T / **F**]

iii. The number 101111001_2 is an **odd** number. [**T** / F]

iv. The hexadecimal number $B7_{16}$ is 267_8 in base-8. [**T** / F]

[2 marks]

Question 5: Type of Errors [6 marks]

The following C++ function attempts to check whether the two input parameters *n* and *m* are **co-prime**. A pair of numbers are **co-prime** if they have no common factor other than 1. (Note: X is a factor of Y, if X divides Y fully).

The function produces a '1' (true) or '0' (false) using the **pass-by reference** parameter `result`.

```
1  void is_coprime(int n, int m, int &result)
2  {
3      int factor = n;
4
5      if (factor < m) {
6          factor = m;
7
8      while (factor > 1) {
9          if (n % factor == 0 || m % factor == 0) {
10             &result = factor;
11             break;
12         }
13         factor--;
14     }
15
16     return 1;
17 }
```

However, there are a number of errors in the implementation. For each error, indicate the line number, the type of error (logic error OR compilation error) and then give the corrected statement. You should consider each mistake separately from others. [6 marks]

Line No	Type of Error	Correction
5	logic error	should be `factor > m`
6	compilation error	add } before line 7.
9	logic error	&& instead of ||
10	compilation error	should be `result = 0;`
12	logic error	should be `return` instead.
16	compilation error	`result = 1`

— END OF PAPER —