

CS2311 Computer Programming (2017-2018 Semester B)

Test 1 / Mar 02, 2017

Duration : 60 minutes

Total : 60 marks

NOTE:

- ❖ There are 4 questions in this paper. Each question 15 marks.
- ❖ Complete all questions in C++. No need to show “**Press any key to continue . . .**” in your code. Program execution speed is not a major concern. Unless stated in the questions, you may assume that all user inputs are valid.
- ❖ Pay attention to code design, naming and code formatting. They are considered in marking.
- ❖ Libraries other than **iostream** and **iomanip** are not allowed.
- ❖ The following are NOT allowed: *break*, *continue*, *conditional operator* `?:`
- ❖ Do NOT amend any given code.
- ❖ In program outputs, ignore English grammar of singular/plural terms like *dollar* vs *dollars*.
- ❖ In all given sample outputs in the questions, Underlined contents are input by the user.
- ❖ Hint: Whenever you find the logic is difficult to write, try to leave these places blank first:
`if ( ? ), while ( ? ), for ( ?;?;? )`

Question 1. Cokes – *cans and packs* (15 marks)

Complete the given program that asks for the amount of cokes wanted and suggests the cheapest buy.

Given costs: \$3 per can  \$22 per pack (10 cans)



The program should run as shown in the examples below:

No. of cokes to buy: <u>24</u> My advice: ===== Buy 2 packs + 4 cans. Pay 56 dollars (\$2.3 per can). Press any key to continue ... Example 1	No. of cokes to buy: <u>30</u> My advice: ===== Buy 3 packs + 0 cans. Pay 66 dollars (\$2.2 per can). Press any key to continue ... Example 2	No. of cokes to buy: <u>28</u> My advice: ===== Buy 2 packs + 8 cans. Pay 68 dollars (\$2.4 per can). Hint: Buy 2 more cans. 3 packs (30 cans) is just 66 dollars. Press any key to continue ... Example 3	← 14 marks ← 1 mark Only. Skip if you find it hard. ☺
--	--	--	---

(14 marks) Basic requirements (Examples 1 and 2) : Handle the advised amounts of packs and cans, as well as the total cost and the average cost (with 1 digit after the decimal point).

(1 mark) Advanced requirement (See example 3) : Show extra "Hint" for smart purchase if the user can get more with the same or even lower cost!!

Question 2. Typing Test (15 marks)

Complete the given program to test the user for typing a-z (ended with '.') as shown on the right.

Handle the following 3 cases only:

- (i) The input is totally correct.
- (ii) The first letter for 'a' is already wrong.
- (iii) It is correct at the beginning, then it becomes wrong.

Assume that the full-stop must be correct. No need to check for its correctness.

Do NOT add any array in your code.

Example for case (i)

```
Input a-z and then full-stop:
abcdefghijklmnopqrstuvwxyz.
Good job!!
Press any key to continue . . .
```

Example for case (ii)

```
Input a-z and then full-stop:
1234.
Wrong at the beginning!!
Press any key to continue . . .
```

Examples for case (iii)

```
Input a-z and then full-stop:
abcd1234.
First 4 letters are correct: abcd
Press any key to continue . . .
```

```
Input a-z and then full-stop:
axcdefghijklmnopqrstuvwxyz.
First 1 letters are correct: a
Press any key to continue . . .
```

Question 3. Grading boundaries & ranges (15 marks)

Complete the given program that asks for the number of passing grades ('A', 'B',...) and their lower boundaries; and then lists all grading ranges (including the failing 'F' grade) as shown in the given examples.

- Note:
- Assume that the number of passing grades $n \geq 2$ and the full-range of marks is 0-100.
 - To display a range like 65 to 79.9, simply write `cout << ".9"` to show the fractional part.

Example 1 – For 3 passing grades: A-C

```
Input number of passing grades: 3
Input the lower boundaries: 80 65 50

[All grades]
A: 80 to 100
B: 65 to 79.9
C: 50 to 64.9
F: 0 to 49.9

Press any key to continue . . .
```

Example 2 – For 5 passing grades: A-E

```
Input number of passing grades: 5
Input the lower boundaries: 85 75 65 55 40

[All grades]
A: 85 to 100
B: 75 to 84.9
C: 65 to 74.9
D: 55 to 64.9
E: 40 to 54.9
F: 0 to 39.9

Press any key to continue . . .
```

Question 4. Next of a 2-digit hexadecimal number (15 marks)

Complete the given program that inputs a 2-digit hexadecimal number and outputs its next number, as shown on the right.

Note: Only A-F and 0-9 are valid digits. If it has invalid letter(s), the program only displays "Cannot Show.". If the user inputs FF, the program also displays "Cannot Show.".

Your task:

Complete the `getNextHex2D` function according to how it is called by `main`.

Note: The given `isValidHex` function is useful. You can call it in `getNextHex2D`.

Example 1 (In hexadecimal, 1A+1 is 1B)

```
Input a 2-digit hexadecimal number: 1A
The next number is: 1B
Press any key to continue . . .
```

Example 2 (In hexadecimal, BF+1 is C0)

```
Input a 2-digit hexadecimal number: BF
The next number is: C0
Press any key to continue . . .
```

Complete your answers for Q1-Q4:

Student ID: _____ (Seat Number: _____)

Q1.

```
#include <iostream>
#include <iomanip>

using namespace std;

int main()
{

    cout << "No. of cokes to buy: ";

    cout << "My advice:" << endl;
    cout << "======" << endl;

    cout << fixed << setprecision(1) << endl;

    return 0;
}
```

No. of cokes to buy: 28

My advice:

=====

Buy 2 packs + 8 cans.
Pay 68 dollars (\$2.4 per can).

Hint: Buy 2 more cans.
3 packs (30 cans) is just 66 dollars.

Press any key to continue ...

Q2.

```
#include <iostream>
using namespace std;

int main()
{
    int count=0; //count correct characters from the beginning
```

```
    cout << "Input a-z and then full-stop: \n";
```

Example for case (i)

```
Input a-z and then full-stop:
abcdefghijklmnopqrstuvwxyz.
Good job!!
Press any key to continue . . .
```

Example for case (ii)

```
Input a-z and then full-stop:
1234.
Wrong at the beginning!!
Press any key to continue . . .
```

Examples for case (iii)

```
Input a-z and then full-stop:
abcd1234.
First 4 letters are correct: abcd
Press any key to continue . . .
```

```
Input a-z and then full-stop:
axcdefghijklmnopqrstuvwxyz.
First 1 letters are correct: a
Press any key to continue . . .
```

```
    if (count==0)
        cout << " Wrong at the beginning!!";
    else if ( )
        cout << "Good job!!";
    else
    {
        cout << "First " << count << " letters are correct: ";

    }

    cout << endl;

    return 0;
}
```

Q3.

```

#include <iostream>
using namespace std;

int main()
{
    int lb[5]; // An array of at most 5 lower boundaries of passing grades
    int n;      // No. of passing grades

    cout << "Input number of passing grades: ";
    cin >> n;

    cout << endl << "Input the lower boundaries: ";

    cout << endl << "[All grades]" << endl;

    return 0;
}

```

Example 1 – For 3 passing grades: A-C

```

Input number of passing grades: 3
Input the lower boundaries: 80 65 50

[All grades]
A: 80 to 100
B: 65 to 79.9
C: 50 to 64.9
F: 0 to 49.9

Press any key to continue . . .

```

Example 2 – For 5 passing grades: A-E

```

Input number of passing grades: 5
Input the lower boundaries: 85 75 65 55 40

[All grades]
A: 85 to 100
B: 75 to 84.9
C: 65 to 74.9
D: 55 to 64.9
E: 40 to 54.9
F: 0 to 39.9

Press any key to continue . . .

```

Q4.

```
#include <iostream>
using namespace std;

// Check whether character c is a valid hexadecimal digit
bool isValidHex(char c)
{
    return (c>='A'&&c<='F' || c>='0'&&c<='9');
}

// Based on the digits of a 2-digit hexadecimal number (c1 and c2), attempt to calculate its next hexadecimal number
// in 2 digits. If successful, return true, otherwise return false.
```

```
_____ getNextHex2D(_____)
```

3 marks for
function header

12 marks for
function body

Hint: for valid cases,
check c2 and decide
the result.

Eg. if c2 is 'A'-'E',
then decide the
resultant number as
c1 and c2+1.

```
}

int main()
{
    char c1, c2, n1, n2;
    cout << "Input a 2-digit hexadecimal number: ";
    cin >> c1 >> c2;

    if (getNextHex2D(c1,c2,n1,n2))
        cout << "The next number is: " << n1 << n2;
    else
        cout << "Cannot show.";

    cout << endl;

    return 0;
}
```

Example 1 (In hexadecimal, 1A+1 is 1B)

```
Input a 2-digit hexadecimal number: 1A
The next number is: 1B
Press any key to continue . . .
```

Example 2 (In hexadecimal, BF+1 is C0)

```
Input a 2-digit hexadecimal number: BF
The next number is: C0
Press any key to continue . . .
```

Total mark: _____ (/60)