



Sheet 2 – Control Flow

(A) REVIEW EXERCISES

1. Find the errors in the following if statements

A	if quarters > 0 then cout << quarters << " quarters";
B	if (1 + x > pow(x, sqrt(2))) y = y + x;
C	if (x = 1) y++; else if (x = 2) y = y + 2;
D	if (x && y == 0) cwin << Point(0, 0);
E	if (1 <= x <= 10) cout << "Enter y: "; cin >> y;
F	if (s != "nick" s != "penn" s != "dime" s != "quar") cout << "Input error!";
G	if (input == "N" or "NO") return 0;
H	cin >> x; if (cin.fail()) y = y + x;
I	language = "English"; if (country == "USA") if (state == "PR") language = "Spanish"; else if (country = "China") language = "Chinese";

2. Of the following pairs of strings, which comes first in lexicographic order?

A	"Tom", "Dick"
B	"Tom", "Tomato"
C	"car manufacturer", "carburetor"
D	"Harry", "hairy"
E	"C++", " Car"
F	"Car", "Carl"

3. Reading numbers is surprisingly difficult, because a C++ input stream looks at the input one character at a time. First, white space is skipped. Then the stream consumes those input characters that can be a part of a number. Once the stream has recognized a number, it stops reading if it finds a character that cannot be a part of a number. However, if the first non-white space character is not a digit or a sign, or if the first character is a sign and the second one is not a digit, then the stream fails.

Consider a program reading an integer:

```
cout << "Enter the number of quarters: ";
int quarters;
cin >> quarters;
```

For each of the following user inputs, circle how many characters have been read and whether the stream is in the failed state or not.

A	15.9
B	15 9
C	+159
D	-15A9
E	Fifteen
F	-Fifteen
G	+ 15
H	1.5E3
I	+1+5

- Explain the difference between an if/else/else statement and nested if statements. Give an example for each.
- Give an example for an if/else/else statement where the order of the tests does not matter. Give an example where the order of the tests matters.
- Complete the following truth table by finding the truth values of the Boolean expressions for all combinations of the Boolean inputs p, q, and r.

p	q	r	(p && q) !r	!(p && (q !r))
0	0	0		
0	0	1		
0	1	0		

- Formulate the following conditions in C++:

x is positive
x is zero or negative
x is at least 8
x is less than 8
x and y are both zero

8. Explain the difference between

```
s = 0;
if (x > 0) s++;
if (y > 0) s++;
```

and

```
s = 0;
if (x > 0) s++;
else if (y > 0) s++;
```

9. What is an infinite loop? On your computer, how can you terminate a program that executes an infinite loop?

10. How often do the following loops execute? Assume that *i* is not changed in the loop body.

A	for (int i = 1; i <= 10; i++) ...
B	for (int i = 0; i < 10; i++) ...
C	for (int i = 10; i > 0; i--) ...
D	for (int i = -10; i <= 10; i++) ...
E	for (int i = 10; i >= 0; i++) ...
F	for (int i = -10; i <= 10; i = i + 2) ...
G	for (int i = -10; i <= 10; i = i + 3) ...

11. Rewrite the following for loop into a while loop.

```
int i;
int s = 0;
for (i = 1; i <= 10; i++) s = s + i;
```

12. Rewrite the following do/while loop into a while loop.

```
int n;
cin >> n;
double x = 0;
double s;
do
{
s = 1.0 / (1 + n * n);
n++;
x = x + s;
}
while (s > 0.01);
```

13. What are the values of s and n after the following loops?

A	<pre>int s = 1; int n = 1; while (s < 10) s = s + n; n++;</pre>
B	<pre>int s = 1; int n; for (n = 1; n < 5; n++) s = s + n;</pre>
C	<pre>int s = 1; int n = 1; do { s = s + n; n++; } while (s < 10 * n);</pre>

14. What do the following loops print? Work out the answer without using the computer.

A	<pre>int s = 1; int n; for (n = 1; n <= 5; n++) { s = s + n; cout << s; }</pre>
B	<pre>int s = 1; int n; for (n = 1; n <= 5; cout << s) { n = n + 2; s = s + n; }</pre>
C	<pre>int s = 1; int n; for (n = 1; n <= 5; n++) { s = s + n; n++; } cout << s << " " << n;</pre>

15. What do the following program segments print? Find the answers by hand, not by using the computer.

A	<pre>int i; int n = 1; for (i = 2; i < 5; i++) n = n + i; cout << n; int i; double n = 1 / 2; for (i = 2; i <= 5; i++) n = n + 1.0 / i; cout << i;</pre>
B	<pre>double x = 1; double y = 1; int i = 0; do { y = y / 2; x = x + y; i++; } while (x < 1.8); cout << i;</pre>
C	<pre>double x = 1; double y = 1; int i = 0; while (y >= 1.5) { x = x / 2; y = x + y; i++; } cout << i;</pre>

(B) PROGRAMMING EXERCISES

1. Write a program that prints all solutions to the quadratic equation

$$ax^2 + bx + c$$

Read in a , b , c and use the quadratic formula. If the discriminant is negative, display a message stating that there are no solutions.

2. Write a program that takes user input describing a playing card in the following shorthand notation:

A	Ace
2 ... 10	Card values
J	Jack
Q	Queen
K	King
D	Diamonds
H	Hearts
S	Spades
C	Clubs

Your program should print the full description of the card.

For example,

Enter the card notation: QS
Queen of spades

3. Write a program that reads in three floating-point numbers and prints the largest of the three inputs.

For example:

Please enter three numbers: 4 9 2.5
The largest number is 9.

4. Write a program that translates a letter grade into a number grade. Letter grades are A, B, C, D, and F, possibly followed by + or -. Their numeric values are 4, 3, 2, 1, and 0. There is no F+ or F-. A + increases the numeric value by 0.3, a - decreases it by 0.3. However, an A+ has value 4.0.

For example:

Enter a letter grade: B
The numeric value is 2.7.

5. Write a program that translates a number between 0 and 4 into the closest letter grade. For example, the number 2.8 (which might have been the average of several grades) would be converted to B-. Break ties in favor of the better grade; for example 2.85 should be a B.
6. Write a program that reads in three strings and sorts them lexicographically.

Enter three strings: Charlie Able Baker
Able
Baker
Charlie

7. A year with 366 days is called a leap year. A year is a leap year if it is divisible by four (for example, 1980), except that it is not a leap year if it is divisible by 100 (for example, 1900); however, it is a leap year if it is divisible by 400 (for example, 2000). There were no exceptions before the introduction of the Gregorian calendar on October 15, 1582 (1500 was a leap year). Write a program that asks the user for a year and computes whether that year is a leap year. Your program should contain a single if statement.
8. Write a program that asks the user to enter a month (1 for January, 2 for February, and so on) and then prints the number of days in the month. For February, print "28 or 29 days".

Enter a month: 5
30 days

9. Mean and standard deviation. Write a program that reads a set of floating-point data values from the input. When the end of file is reached, print out the count of the values, the average, and the standard deviation. The average of a data set $\{x_1, \dots, x_n\}$ is \bar{x} , where \bar{x} is the sum of the input values. The standard deviation is

$$s = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n - 1}}$$

However, this formula is not suitable for the task. By the time the program has computed, the individual x_i are long gone. Until you know how to save these values, use the numerically less stable formula

$$s = \sqrt{\frac{\sum x_i^2 - \frac{1}{n}(\sum x_i)^2}{n - 1}}$$

You can compute this quantity by keeping track of the count, the sum, and the sum of squares as you process the input values.

10. The Fibonacci numbers are defined by the sequence

$$f_1 = 1$$

$$f_2 = 1$$

$$f_n = f_{n-1} + f_{n-2}$$

Implement a program that computes the Fibonacci numbers.

11. Prime numbers. Write a program that prompts the user for an integer and then prints out all prime numbers up to that integer. For example, when the user enters 20, the program should print

```
2
3
5
7
11
13
17
19
```

Recall that a number is a prime number if it is not divisible by any number except 1 and itself.