

علوم الحاسب الفرقة الأولى

Chapter 2

برمجة الحاسبات

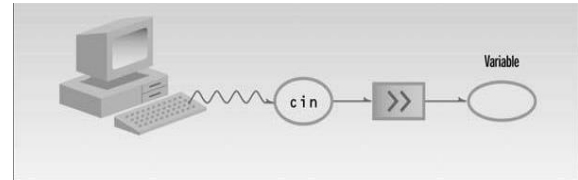
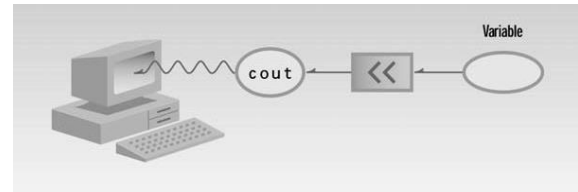
Computer Programming



Computer Science Department

Q.1. Write a C++ program that accepts two integer numbers and prints the sum.

```
#include <iostream>
#include <conio.h>
using namespace std;
void main(){
    int x, y, sum;
    cout << "Enter two numbers" << endl;
    cin >> x >> y;
    sum = x + y;
    cout << "The sum is: " << sum;
    _getch();
}
```



```
Enter two numbers
2
3
The sum is: 5
```

Q.2. Complete each of the following sentences.

- 1) A C++ program can divide into different namespaces.
- 2) The last program contains only one function called main().
- 3) Function name must be followed by parentheses (). [True]
- 4) The body of a function is surrounded by braces. BEGIN { and END }
- 5) The program may consist of many functions, classes, but on startup, control always goes to main().
- 6) The smallest program in C++ is main() { }
- 7) #include is called preprocessor directive.
- 8) IOSTREAM: is an example of a header file concerned about input, output operations.
- 9) Std stands for Standard.
- 10) The endl Manipulator has the same effect as sending the '\n'.
- 11) cout or cin is called object or identifier.
- 12) The operator << is called the insertion.
- 13) The operator >> is called the extraction.

Q.3. Write a program to print Hello world, don't use namespace.

```
#include <iostream>
#include <conio.h>
void main(){
    std::cout << " Hellow World";
    _getch();
}
```

Q.4. What is escape give an example

↳ Backslash \ is called "escape".

Escape	Character
\ a	Bell (beep).
\ b	Backspace.
\ n	Newline.
\ t	Tab.
\ \	Backslash.
\ '	Single quotation.
\ "	Double quotation.

↳ Example:

```
cout << " \" Run, Spot, \n run, \" she said. \t Hello World ";
```

```
" Run, Spot,
run, " she said.      Hello World
```

Q.5. What are comments stating its types?

- A piece of data that the compiler ignores, so not add to the file size.
- There are two types of comments:
 - 1) **Line comment:** starts with a // and end at the end of the line.
 - 2) **Block comment:** Begin with the /* and ends with */

```
// This is a line comment
```

```
/*
```

```
    This is a block comment
```

```
    This is a block comment
```

```
    //This is a block comment → not line comment inside block comment.
```

```
*/
```

Q.6. What are variables state rules for writing variables.

↪ A variable has a name called **identifiers**.

↪ Rules for writing **identifiers**.

- 1) The first character must be a letter or underscore.
- 2) Can't use a C++ keywords (e.g., int, double, if, while, class).
- 3) Can't use special characters (\$ # % ^)or space.
- 4) Compiler distinguishes between upper- and lowercase.
- 5) Can use upper- and lower-case letters, and the digits from 1 to 9, Can also use the underscore (_)
- 6) Can use `sizeof()` function to display the size of all the data types.

```
#include <iostream>
#include <conio.h>
using namespace std;
void main(){
    cout << sizeof(char) << endl; // 1 byte
    cout << sizeof(bool) << endl; // 1 byte
    cout << sizeof(short int) << endl; // 2 byte
    cout << sizeof(int) << endl; // 4 byte
    cout << sizeof(long) << endl; // 4 byte
    cout << sizeof(float) << endl; // 4 byte
    cout << sizeof(double) << endl; // 8 byte
    cout << sizeof(unsigned int) << endl; // 4 byte
    cout << sizeof(unsigned long) << endl; // 4 byte
    _getch();
}
```

↪ To defined constant use `const`

```
const float PI = 3.14F;
```

Q.7. True or False

- 1) `int _ = 5;` (True)
- 2) `int _x = 5;` (True)
- 3) `int first age;` (False)
- 4) `int 1x;` (False)
- 5) `int x1;` (True)
- 6) `int int;` (False)
- 7) `int Int;` (True)
- 8) `int void;` (False)
- 9) `int Void;` (True)
- 10) `int include;` (True)
- 11) `int main = 5;` (True)
- 12) `int main() = 5;` (False)
- 13) `int bool;` (False)
- 14) `int cin;` (True)
- 15) `int iostream;` (True)
- 16) `int #iostream;` (False)
- 17) `int const;` (False)
- 18) `int Const;` (True)
- 19) `int Mix-Up;` (False)
- 20) `int Mix_Up;` (True)

Q.8. Does the next program run without error, why?

```
int x = 5;
float y = 6.7f;
double z = x * y;
```

- ↳ This program compiles without error because the compiler converted the lower-type variable to the higher-type variable.

Data Type	Order
long double	Highest
double	
float	
long	
int	
short	
char	Lowest

Q.9. Evaluate the next Expression

Expression	Evaluates to
<code>static_cast<int> (7.9)</code>	7
<code>static_cast<int> (3.3)</code>	3
<code>static_cast<double> (25)</code>	25.0
<code>static_cast<double> (5+3)</code>	= <code>static_cast<double> (8)</code> = 8.0
<code>static_cast<double> (15) / 2</code>	= 15.0 / 2 (because <code>static_cast<double> (15)</code> = 15.0) = 15.0 / 2.0 = 7.5
<code>static_cast<double> (15 / 2)</code>	= <code>static_cast<double> (7)</code> (because $15 / 2 = 7$) = 7.0
<code>static_cast<int> (7.8 + static_cast<double> (15) / 2)</code>	= <code>static_cast<int> (7.8 + 7.5)</code> = <code>static_cast<int> (15.3)</code> = 15
<code>static_cast<int> (7.8 + static_cast<double> (15 / 2))</code>	= <code>static_cast<int> (7.8 + 7.0)</code> = <code>static_cast<int> (14.8)</code> = 14

Q.10. Complete each of the following sentences.

- 1) +, -, *, /, % called arithmetic operators.
- 2) = called assignment operator.
- 3) +=, -=, *=, /=, %= called arithmetic assignment operators.
 ↪ E.g., total = total + item; → total += item;
- 4) ++ Increment operator.
- 5) -- Decrement operator.
 ↪ Increment and decrement operators can be prefix or postfix.
 ↪ ++i is called prefix where i++ is called postfix.

```
#include <iostream>
#include <conio.h>
using namespace std;
void main(){
    int count = 10;
    cout << count++ << endl;
    cout << ++count << endl;
    cout << count++ << endl;
    cout << count++ << endl;
    cout << ++count << endl;
    cout << count++ << endl;
    cout << ++count << endl;
    _getch();
}
```

```
10
12
12
13
15
15
17
```

Q.11. Write output for the following code.

```
int a, c=3;
a = --c + --c + --c;
cout << a;
```

The output.....

```
int a = 2, b = 2, c=2;
c = ++a + --b;
b = --a - ++b;
a = --c - --c;
c = ++a - --a;
cout << a << endl;
cout << b << endl;
cout << c << endl;
```

The output.....

Q.12. Write a program that asks the user for a temperature in degree Fahrenheit, convert it to Celsius, and display result.(use integer variables). (Celsius=(Farhinheit-32)*5/9)

Q.13. Write a program that asks the user to type in floating-point number representing the radius of a circle, then calculate and display the circle's area.

Exercise Chapter 2

2.1 Mark the following statements as true or false.

- a) An identifier can be any sequence of digits and letters. [False]
- b) In C++, there is no difference between a reserved word and a predefined identifier. [False]
- c) A C++ identifier can't start with a digit. [True]
- d) The operands of the modulus operator must be integers. [True]
- e) If $a = 4$; and $b = 3$; , then after the statement $a = b$; the value of b is still 3. [True]
- f) In the statement $\text{cin} \gg y$; , y can only be an `int` or a `double` variable. [False]
- g) In an output statement, the newline character may be a part of the string. [True]
- h) The following is a legal C++ program:

```
int main()
{
return 0;
} [True]
```
- i) In a mixed expression, all the operands are converted to floating-point numbers. [False]
- j) Suppose $x = 5$. After the statement $y = x++$; executes, y is 5 and x is 6. [True]
- k) Suppose $a = 5$. After the statement $++a$; executes, the value of a is still 5 because the value of the expression is not saved in another variable. [False]

2.2 Which of the following are valid C++ identifiers?

- a.Myfirsrtprogram b. MIX-UP c. C++Program2 d.quiz7
e.ProgrammingLecture2 f. 1footEqual12inche g.Mike'sFirstAttempt
h.Update Grade i.4th j.New Student

2.3 Which of the following is a reserved word in C++?

- a. main b. #include c. double
d.const e. cin f.Bool

2.4 What is the difference between a reserved word and a user-defined identifier?

- » A reserved word is part of the C/C++ language. A standard identifier is not.

2.5 Are the identifiers `firstName` and `FirstName` the same?

- » The identifiers are not the same. C++ is case sensitive.



2.6 Evaluate the following expressions.

a. $25/3$

b. $20-12/4*2$

c. $32\%7$

d. $3-5\%7$

e. $18.0/4$

f. $28-5/2.0$

g. $17+5\%2-3$

2.7 If `int x = 5;`, `int y = 6;`, `double z = 4;`, and `double w = 3.5;`, evaluate each of the following statements, if possible. If it is not possible, state the reason.

a. $(x + y) \% y$

b. $x \% y - w$

c. $(y + z) / w$

d. $x * z \% y + w$

e. $(x \% y) * z$

f. $(x * y \% z) - w$

g. $x \% (y + z)$

h. $(x \% y + z) / w$

2.8 Given: `int num1, num2, newNum;` `double x, y;` Which of the following assignments are valid? If an assignment is not valid, state the reason.

a. `num1 = 35;`

b. `newNum = num1 - num2;`

c. `num1 = 5; num2 = 2 + num1; num1 = num2 / 3;`

d. `num1 * num2 = newNum;`

e. `x = 12 * num1 - 15.3;`

f. `num1 * 2 = newNum + num2;`

g. `x / y = x * y;`

h. `num2 = num1 % 2.0;`

i. `newNum = static_cast<int>(x) % 5;`

j. `x = x + y - 5;`

k. `newNum = num1 + static_cast<int>(4.6 / 2);`

2.9 Suppose that `x, y, z, w,` and `t` are `int` variables. What is stored in `x, y, z, w,` and `t` after the following statements execute?

`x = 5;`

`y = x + 2;`

`z = x % (y - 2) + 4;`

`w = (x * y) / z - 5;`

`t = z + (x + y + 2) % w;`

Answer: `x = 5, y = 7, z = 4, w = 3, t = 6`

2.10 Which of the following variable declarations are correct? If a variable declaration is not correct, give the reason(s) and provide the correct variable declaration.

`n=12;`

`char letter=;`

`int one=5, two;`

`double x,y,z;`



2.11 Which of the following are valid c++ assignment statements? Assume that i, x and present are double variables.

a. `i=i+5;`

b. `x+2=x`

c. `x = 2.5*x;`

d. `percent = 10%;`

2.12 Write C++ statement(s) that accomplish the following:

a. Declare int variables x and y. Initialize x to 25 and y to 18.

b. Declare and initialize an int variable temp to 10 and a char variable ch to 'A'.

c. Update the value of an int variable x by adding 5 to it.

d. Declare and initialize a double variable payRate to 12.50.

e. Copy the value of an int variable firstNum into an int variable tempNum.

f. Swap the contents of the int variables x and y. (Declare additional variables, if necessary.)

g. Suppose x and y are double variables. Output the contents of x, y, and the expression $x + 12 / y - 18$.

h. Declare a char variable grade and set the value of grade to 'A'.

i. Declare int variables to store four integers.

j. Copy the value of a double variable z to the nearest integer into an int variable x.

2.13 What is the output of each of the following statements? Suppose a, b, are int and c is a double variable, and a=13, b=5, and c=17.5

a. `cout<< a+b-c<<endl;`

b. `cout << 15/2+c <<endl;`

c. `cout << a/static_cast<double>(b)+2*c <<endl;`

d. `cout <<14 % 3+ 6.3 +b/a <<endl;`

e. `cout << static_cast <int> (c) % 5 +a-b <<endl;`

f. `cout << 13.5 /2 +4.0 * 3.5 +18 <<endl;`

2.14 Why is the main() function special?

Because it is the first function executed when the program starts.



2.15 What header file must you #include with your source file to use cout and cin?

iostream

2.16 Write a statement that gets a numerical value from the keyboard and places it in the variable temp.

cin >> temp;

2.17 Write a statement that uses an arithmetic assignment operator to increase the value of the variable temp by 23. Write the same statement without the arithmetic assignment operator.

temp += 23;
temp = temp + 23;

2.18 Assuming var1 starts with the value 20, what will the following code fragment print out?

cout << var1--;
cout << ++var1;

The output is 2020

2.19 On a certain day the British pound was equivalent to \$1.487 U.S., the French franc was \$0.172, the German deutschemark was \$0.584, and the Japanese yen was \$0.00955. Write a program that allows the user to enter an amount in dollars, and then displays this value converted to these four other monetary units.

```
#include<iostream>
using namespace std;
void main()
```

```
{
    float pound = 1.478;
    float franc = 0.172;
    float deutschemark = 0.584;
    float yen = 0.00955;
    float dollar;
    cout<<"Enter an ammount in US Doller = ";
    cin>> dollar;
    cout<<"Pound = "<<dollar/pound<<endl;
    cout<<"France = "<<dollar*franc<<endl;
    cout<<"deutschemark = "<<dollar*deutschemark<<endl;
    cout<<"Yen = "<<dollar*yen<<endl;
}
```



2.20 You can convert temperature from degrees Celsius to degree Fahrenheit by multiplying by 9/5 and adding 32. Write a program that allows the user to enter a floating point number representing degree Celsius, and then displays the corresponding degrees Fahrenheit.

```
#include <iostream>
using namespace std;
void main()
{
    float c,f;
    cout<< "Enter degrees in Celsius "<< endl;
    cin >> c;
    f=c*9/5 +32;
    cout << "Degree in Fahrenheit "<<f;
}
```

2.21 Rewrite the following program without errors:

```
#include
<iostream>;

using
namespace std;

int main
{
int intVar = 1500000000;
intVar=(intVar * 10) /10;
cout << "intVar= " << intVar <<endl

intVar = 1500000000;
intVar = ((double intVar *10) /10;
cout
<<
"intVar= " << intVar endl;
}
```

```
#include <iostream>

using namespace std;

int main()
{
int intVar = 1500000000;
intVar=(intVar * 10) /10;
cout << "intVar= " << intVar <<endl;

intVar=1500000000;
intVar = ((double) intVar *10) /10;
cout << "intVar= " << intVar <<endl;
return 0;
}
```



2.22 The following program has syntax mistakes. Correct them. On each successive line, assume that any preceding error has been corrected

```
#include <iostream>\
    const int SECRET_NUM =11,213;
    const PAY_RATE =18.35

main()
{
    int one, two;
    double first, second;
    one=18;
    two=11;
    first=25;
    second=first*three;
    second=2*SECRET_NUM;
    SECRET_NUM=SECRET_NUM +3;
    cout<< first<<" "<< second<<SECRET_NUM<<endl;
    paycheck=hoursWoked* PAY_RATE
    cout<<"Wages " << paycheck<<endl;
    return 0;
}
```

2.23 The following program has syntax errors. Correct them. On each successive line, assume that any preceding error has been corrected.

```
const char = STAR = '*'
const int PRIME = 71;
int main
{
    int count, sum;
    double x;
    count = 1;
    sum = count + PRIME;
    x = 25.67
    newNum = count * ONE + 2;
    x + sum)++;
    sum + count = sum;
    x = x + sum * COUNT;
    sum += 3—;
    cout << " count = " << count << ", sum = " << sum
    << ", PRIME = " << Prime << endl;
}
```

```
#include <iostream>
using namespace std;
const char STAR = '*';
const int PRIME = 71;
int main()
{
    int count, sum;
    double x;
    int newNum;
    count = 1;
    sum = count + PRIME;
    x = 25.67;
    newNum = count * 1 + 2;
    sum++;
    sum = sum + count;
    x = x + sum * count;
    sum += 3;
    cout << " count = " << count << ", sum = " << sum
    << ", PRIME = " << PRIME << endl;
    return 0;
}
```



2.24 Write the following compound statements as equivalent simple statements.

a. $x += 5 - z;$ b. $y *= 2 * x + 5 - z;$ c. $w += 2 * z + 4;$
 d. $x -= z + y - t;$ e. $sum += num;$

2.25 Suppose a, b, and c are int variables and a = 5 and b = 6. What value is assigned to each variable after each statement executes? If a variable is undefined at a particular statement, report UND (undefined).

$a = (b++) + 3;$
 $c = 2 * a + (++b);$
 $b = 2 * (++c) - (a++);$

2.26 Suppose a, b, and sum are int variables and c is a double variable. What value is assigned to each variable after each statement executes? Suppose a = 3, b = 5, and c = 14.1.

$sum = a + b + c;$
 $c /= a;$
 $b += c - a;$
 $a *= 2 * b + c;$

2.27 Write a C++ program that prompts the user to input the elapsed time for an event in seconds. The program then outputs the elapsed time in hours, minutes, and seconds. (For example, if the elapsed time is 9630 seconds, then the output is 2:40:30.), and seconds.

```
#include <iostream>
using namespace std;
int main()
{
    int inputSeconds, hours, minutes, seconds;
    cout << "Please enter the number of seconds: ";
    cin >> inputSeconds;
    hours = inputSeconds / (60 * 60);
    inputSeconds = inputSeconds % (60 * 60);
    minutes = inputSeconds / 60;
    seconds = inputSeconds % 60;
    cout << hours << ":" << minutes << ":" << seconds << endl;
    return 0;
}
```



2.28 Write a c++ program that prompts the user to input the elapsed time for an event in hours, minutes, and seconds. The program then outputs the elapsed time in seconds.

```
#include <iostream>
using namespace std;
int main()
{
    int hours, minutes, seconds, totalSeconds;
    cout << "Please input the hours minutes and seconds: ";
    cin >> hours >> minutes >> seconds;
    totalSeconds = seconds + (minutes * 60) + (hours * 60);
    cout << "That equals a total of " << totalSeconds << " seconds." << endl;
    return 0;
}
```

2.29 To make a profit, a local store marks up the prices of its items by a certain percentage. Write a C++ program that reads the original price of the item sold, the percentage of the marked-up price, and the sales tax rate. The program then outputs the original price of the item, the percentage of the mark-up, the store's selling price of the item, the sales tax rate, the sales tax, and the final price of the item. (The final price of the item is the selling price plus the sales tax.)

```
#include <iostream>
using namespace std;
int main()
{
    double originalPrice, salesTaxRate, totalPrice, markupPercentage, salesTaxPrice, markupPrice;
    cout << "Please enter the original price: $";
    cin >> originalPrice;
    cout << "Please enter the mark-up percentage: ";
    cin >> markupPercentage;
    cout << "Please enter the sales tax percentage: ";
    cin >> salesTaxRate;
    markupPrice = originalPrice * (markupPercentage / 100);
    salesTaxPrice = originalPrice * (salesTaxRate / 100);
    totalPrice = originalPrice + salesTaxPrice + markupPrice;
}
```



```

cout << "Original Price: $" << originalPrice << endl;
cout << "Sales Tax: $" << salesTaxPrice << endl;
cout << "Mark-Up: $" << markupPrice << endl;
cout << "Total Price: $" << totalPrice << endl;
return 0;
}

```

2.30 Write a program that prompts the user to input a length expressed in centimeters. The program should then convert the length to inches (to the nearest inch) and output the length expressed in yards, feet, and inches, in that order. For example, suppose the input for centimeters is 312. To nearest inch 312 centimeters is equal to 123 inches. 123 inches would thus be output as 3 yard(s), 1feet (foot),and 3 inch(es).

```

#include <iostream>
#include <conio.h>
using namespace std;
void main(){
    double centimeters;
    int inches, feet, yards, totalinches;
    cout << "Enter centimeters" << endl;
    cin >> centimeters;
    totalinches =(int)(centimeters / 2.54);

    yards = totalinches / 36;
    totalinches = totalinches % 36;

    feet = totalinches / 12;
    totalinches = totalinches % 12;

    inches = totalinches;
    cout << "Yards= " << yards << " Feets= " << feet
        << " Inches= " << inches << endl;
    _getch();
}

```

