程式題（每題 10%）

1. Determine whether the following code has syntax errors or not. If it is correct, predict its output. If it is incorrect, point out the mistake(s).

```
// modulus
#include <iostream>
using std::cout;
using std::endl;

int mod(int a, int b)
{
            do     a -= b;
            while (a >= b);
            return a;
}

int main()
{
        cout << mod(621, 6) << endl;
        return 0;
}
```

2. Determine whether the following code has syntax errors or not. If it is correct, predict its output. If it is incorrect, point out the mistake(s).

```
// Caesar Cipher
#include <iostream>
using std::cout;
using std::endl;

void enc(char* a, int key)
{
        while (*a)
        {
                if (*a >= 'A' && *a <='Z')
                        *a = (*a - 'A' + key) % 26 + 'A';
                a++;
        }
}

int main()
{
        char msg[] = "CITY COUNCIL";
        enc(msg, 13);
        cout << msg << "\n";
        return 0;
}
```

3. What will the following code display?

```
#include <iostream>
int main() // bitwise operators
{
        unsigned s = 831;
        int i = (s >> 4) & ~(~0 << 3);
        std::cout << i << "\n";
       return 0;
}
```

4. Determine whether the following code has syntax errors or not. If it is correct, predict its output. If it is incorrect, point out the mistake(s).

```
// Define a class
#include <iostream>
using std::cout;
using std::endl;

class CBox                    // Class definition at global scope
{
public:
    double m_Length;          // Length of a box in inches
    double m_Width;           // Width  of a box in inches
    double m_Height;          // Height of a box in inches
};

int main()
{
  CBox box1;                  // Declare box1 of type CBox

  box1.m_Height = 3.0;        // Define the values
  box1.m_Length = 7.0;        // of the members of
  box1.m_Width  = 2.0;        // the object box1

  cout << "Volume of box1 = "
       <<  box1.m_Height * box1.m_Length * box1.m_Width
       << endl;
  return 0;
}
```

5. Determine whether the following code has syntax errors or not. If it is correct, predict its output. If it is incorrect, point out the mistake(s).

```
#include <iostream>     // Declaring struct and class.
using std::cout;
using std::endl;

struct RECT
{ int left;          int top;
  int right;        int bottom;
};

class CRECT
{
public:
  int left;
  int top;
  int right;
  int bottom;

CRECT(int x1, int y1, int x2, int y2) // Constructor
   : left(x1), top(y1), right(x2), bottom(y2) {}

int Area()
   { return (right-left)*(bottom-top); }

};

int main()
{
  RECT aRect = { 20, 30, 40, 50 };     // Initial value
  CRECT bRect(10, 10, 50, 50);       // Initial value
  cout << (aRect.right - aRect.left) *
            (aRect.bottom - aRect.top) << endl;
  cout << bRect.Area() << endl;

  return 0;
}
```

6. Determine whether the following code has syntax errors or not. If it is correct, predict its output. If it is incorrect, point out the mistake(s).

```
#include <iostream>
using std::cout;
using std::endl;

 // This function returns a pointer to an integer

int *max(int a, int b)
{   return (a>b) ? &a : &b;

int main()
{   cout << *max(83,3)  << endl;
    cout << *max(6,22) << endl;
    return 0;
}
```

7. Determine whether the following code has syntax errors or not. If it is correct, predict its output. If it is incorrect, point out the mistake(s).

```
#include <iostream>
using std::cout;

float average(float a, float b);
float average(float a, float b, float c);

int main()
{   // 3.0 will be output as 3
    cout << average(1.0, 2.2, 3.1) << "\n";
    cout << average(7.0, 5.0) << "\n";
    return 0;
}

float average(float a, float b)
{ return (a+b)/2.0; }

float average(float a, float b, float c)
{ return (a+b+c)/3.0; }
```

8. Determine whether the following code has syntax errors or not. If it is correct, predict its output. If it is incorrect, point out the mistake(s).

```
// Pointer to char
#include <iostream>
#include <string.h>
using std::cout;
using std::endl;

int main()
{
    char s[] = "City Council";
    char *p1, *p2;
    char c;

    cout << s << endl;

    for (p2=s; *p2 != '\0'; p2++);
    p2--;
    p1=s;
    while (p1 < p2)
    {
        c = *p1; *p1 = *p2; *p2 = c;
        p1++; p2--;
    }

    cout << s << endl;

    return 0;
}
```

9. Determine whether the following code has syntax errors or not. If it is correct, predict its output. If it is incorrect, point out the mistake(s).

```
#include <iostream>
using std::cout;

int main()
{
    int i, n = 621;
    for (i=1; i<7; i++)
    {
        int n = 0;
        for (int j=1; j<=i; j++)
            n += j;            // scope of variable n
        cout << n << "\n";
    }
        cout << n << "\n";
    return 0;
}
```

10. Determine whether the following code has syntax errors or not. If it is correct, predict its output. If it is incorrect, point out the mistake(s).

```
// Accessing the members of a class
#include <iostream>
using std::cout;
using std::endl;

class CComplex
{
        double real;
        double imaginary;

        CComplex(double a = 0.0, double b = 0.0):
        real(a), imaginary(b)
        { }

                CComplex operator+(CComplex w)
                { return CComplex(real + w.real,
                        imaginary - w.imaginary); }
        void Show()
        {
                if (real != 0.0) cout << real;
                if (imaginary < 0.0)
                        cout << imaginary << "i";
                else if
                        (imaginary > 0.0)
                        cout << "+" << imaginary << "i";
                cout << "\n";
        }
};

int main()
{
        CComplex c1(2,2);
        CComplex c2(3,1);
        CComplex c3 = c2 + c1;
        c3.Show();          // "cout << 3.0" will display "3".
}
```