C++FA 5.1 PRACTICE MID-TERM EXAM

This practicemid-term exam covers sections C++FA 1.1 through C++FA 1.4 of ***C++ with Financial Applications*** by Ben Van Vliet, available at **www.benvanvliet.net**.

1.) A pointer is: - ______

a.) An address variable.
b.) A variable containing the location of an existing mailbox.
c.) Another name on the address of an existing memory location.
d.) The address of a value type or reference type.
e.) None of the above.

2.) In the following code, *x* *could be* a: - ______

```
Number x;
```

1. An enum.
2. A structure.
3. A typedef.
4. An object.

a.) 2., or 4.
b.) 1., 3., or 4.
c.) 1., 2., 3., or 4.
d.) 2., 3., or 4.
e.) None of the above.

3.) Which of the following would necessarily imply function overriding? - ______

a.) Two functions where the definition of one replaces the other.
b.) Two functions with the different signatures.
c.) Two virtual functions that are defined in the base class.
d.) Two class member functions with the same name.
e.) None of the above.

For questions 4 - 8, consider the following lines of code in the boxes *in sequence*:

```
int a[ 7 ] = { 5, 6, 3, 8, 7, 1, 2 };
int *b;
b = a;
```

4.) What is the value of (a + 2)? - ______

a.) The address of 6.
b.) The address of 3.
c.) 6.
d.) 3.
e.) None of the above.

5.) What is the value of b[3]? - ______

a.) The address of 8.
b.) The address of 7.
c.) 7.
d.) 8.
e.) None of the above.

***If next the following line of code runs*:**

```
b += 4;
```

6.) Now, what is the value of &((b – 2)[4])? - ______

a.) 2 .
b.) The address of 2.
c.) 1
d.) The address of 1
e.) None of the above.

If now the following line of code runs:

```
( *( ++b ) )++;
```

7.) **Now, what is the value of b? -** ______

a.) The address of 7.
b.) The address of 8.
c.) The address of 1.
d.) The address of 2.
e.) None of the above.

8.) **Now, what is the value of (b - 2)[3]? -** ______

a.) 1.
b.) The address of 1.
c.) 2.
d.) The address of 2.
e.) None of the above.

9.) **The following line of code can *best* be described as: -** ______

```
void bar( int, void (*)[]( int ));
```

a.) The definition of a function that accepts an int and function pointer.
b.) A declaration with type parameters int and array of function pointers.
c.) The prototype of a function that points to another function.
d.) A function of type void that accepts an array of void pointers to ints.
e.) None of the above.

For Questions 10-15, assume the following class definitions:

```
class Instrument
{
protected:
     int value;
     virtual void calc_value() { value = 1; }
public:
     Instrument( int v ) : value( v )
     {
          cout << "Constructor running on " << value << endl;
     }
     Instrument( Instrument &i )
     {
          value = i.get_value() + 1;
     }
     ~Instrument()
     {
          cout << "Destructor running on " << value << endl;
     }
     Instrument operator+( Instrument &i )
     {
          return Instrument( value + i.get_value() );
     }
     int get_value()
     {
          calc_value();
          return value;
     }
     static void print_type()
     {
          cout << "Instrument class!" << endl;
     }
};

class Bond : public Instrument
{
protected:
     void calc_value() { value = 3; }
public:
     Bond( int v ) : value( v ) {}
};
```

10.) The function Instrument(Instrument &i) is: - ______

a.) A constructor overload.
b.) A constructor override.
c.) The copy constructor.
d.) The destructor.
e.) None of the above.

11.) In the Instrument class, the calc_value function is: - ______

a.) Overridable.
b.) Pure virtual.
c.) Must override.
d.) Overloaded.
e.) None of the above.

12.) The operator+ function requires a parameter because: - ______

a.) + is a binary operator.
b.) + is a function that requires a left-hand value.
c.) + is a unary operator.
d.) + implicitly creates a copy of the parameter.
e.) None of the above.

13.) The print_type method: - ______

a.) Could not access the value data member.
b.) Can only be called from the class.
c.) Is not inherited because it is static.
d.) Would print Bond for the bond class.
e.) None of the above.

14.) The Bond class: - ______

a.) Overloads the calc_value method.
b.) Must override the calc_value method.
c.) Will not inherit the base class operator+ method.
d.) Fails to call the base class constructor.
e.) None of the above.

15.) The Instrument class destructor: - ______

a.) Runs only when an instance of Instrument is destroyed.
b.) Should be declared as virtual.
c.) Will not run when calling delete on a Bond.
d.) Must be overridden in a derived class.
e.) None of the above.

16.) **The following line of code is *best* described by: -** ______

```
MyClass::MyClass( int v ) : value( v ) {}
```

a.) A .cpp file class method.
b.) A constructor prototype.
c.) A static method call.
d.) A .h file copy constructor.
e.) None of the above.

17.) **Given the following function: -** ______

```
void foo( int *x )
{
     *x *= *x;
     *x = ( *x + ( 2 * *x ) ) + ( *x + 2 );
}
```

What is the output from the following code?

```
int y = 5;
foo( &y );
cout << y;
```

a.) 42
b.) 22
c.) 32
d.) 17
e.) None of the above.

18.) **What is the output from the following code? -** ______

```
int a[] = { 8, 9, 10, 11, 12 };
( *( a + 3 ) )++;
cout << *a;
```

a.) 12
b.) 10
c.) 11.
d.) The address of 11.
e.) None of the above.

19.) **What is the output from the following code? -** ______

```
for( int x = 1; x < 10; x += 2 )
{
     if ( x > 7 || x <= 2 )
     {
          cout << x << " ";
          x--;
     }
}
```

a.) 1 3 7 9
b.) 1 3 9
c.) 1 2 7 9
d.) 1 2 9
e.) None of the above.

20.) **What is the output from the following code assuming `&a = 100` and `&b = 200` and `&c = 300`? -**

```
int a = 3;
int *b = &a;
int **c = &b;
cout << c << " ";
cout << *c << " ";
cout << &( *c ) << " ";
cout << &( **c ) << endl;
```

a.) 100 200 300 100
b.) 200 100 200 100
c.) 200 100 300 3
d.) 100 200 300 3
e.) None of the above.

21.) What is the output from the following code? - ________

```
int a[] = { 7, 8, 3, 4 };
int *b = a;

cout << a[ 1 ] << " ";
cout << ( a + 2 )[ 1 ] << " ";
cout << b[ 2 ] << " ";
cout << *( ++b ) << " ";
cout << ( b - 1 )[ 0 ] << " ";
```

a.) 8 4 3 8 8
b.) 8 4 4 8 8
c.) 8 4 4 8 4
d.) 8 4 3 8 7
e.) None of the above.

22.) What is the purpose of the following line of code? - ________

```
typedef double Real;
```

a.) Declare a variable called Real of type double.
b.) Define double as Real.
c.) Create a type called Real that is a double.
d.) Create Real as an instance of double.
e.) None of the above.

23.) Given the code in question 22, what is the purpose of the following code? - ________

```
Real values[ 5 ];
```

a.) Create a values object.
b.) Create a two dimensional array of doubles.
c.) Create an array of Reals.
d.) Create a double typedef.
e.) None of the above.

24.) If a and b are booleans, which answer is logically equivalent to the following statement? - ________

```
!( a || b ) == ( a && b )
```

a.) !a || b
b.) a && b
c.) a || b
d.) !(a == b)
e.) None of the above.

25.) Given the following function: - ________

```
int f( int &x )
{
	x -= 3;
	cout << x << " ";
	x += x;
	return x;
}
```

What is the output from the following code?

```
int a = 2;
int b = f( a );
cout << f( b ) << " ";
cout << a << endl;
```

a.) -1 -8 -10 -2
b.) 1 -6 -8 0
c.) -1 -8 -10 -10
d.) -1 -6 -2 -2
e.) None of the above.

26.) What is the output from the following code? - ________

```
cout << "a\t\tb\n\t\tc\n";
```

```
a.)  a             b
     c
b.)  a             b
                   c
c.)  a
     b             c
d.)  a             b             c
e.)  None of the above.
```

For Questions 27-31, assume the following class definition:

```
class MyClass
{
private:
      int v;

public:

     MyClass( int a = 2 ) : v( a )
      {
           cout << v << “ is alive.” << endl;
      }
     ~MyClass(void)
     {
          cout << v << “ is dying.” << endl;
     }

     MyClass operator-( MyClass &c )
     {
           MyClass r( v - c.get_Value() * 3 );
           return r;
     }
     void set_Value( int a )
     {
           v = a;
     }
     int get_Value()
     {
           return v;
      }
};
```

27.) What is the output from the following code? - _______

```
if ( true )
{
     MyClass m_Obj( 4 );
     cout << m_Obj.get_Value() << " ";
}
```

a.) 4 is alive. 4 4 is dying.
b.) 4 4 is dying.
c.) 4 is alive. 4
d.) 4
e.) None of the above.

28.) What is the output from the following code? - _______

```
if ( true )
{
      MyClass *m_Obj = new MyClass;
}
```

a.) 2 is alive.
b.) [Garbage value] is alive.
c.) is alive. is dying.
d.) Compile error.
e.) None of the above.

29.) *Not* including the constructor and destructor code, what is the output from the following code? – _______

```
MyClass m_Obj1( 4 );
MyClass m_Obj2( 5 );
MyClass m_Obj3 = m_Obj1 - m_Obj2;
cout << m_Obj3.get_Value() << endl;
```

a.) -9
b.) -11
c.) Compile error.
d.) Runtime error.
e.) None of the above.

30.) ***Not*** **including the constructor and destructor code, what is the output from the following code? -**

```
MyClass *m_Obj1 = new MyClass( 4 );
MyClass *m_Obj2 = new MyClass( 6 );
MyClass m_Obj3 = *m_Obj1 - *m_Obj2;
cout << m_Obj3.get_Value() << endl;
```

a.) -2
b.) Compile error.
c.) Runtime error.
d.) -14
e.) None of the above.

31.) What is the output from the following code? - ______

```
int main()
{
     MyClass m_Obj1(4);
     MyClass *m_Obj2 = new MyClass(4);
     return 0;
}
```

a.) Nothing
b.) 4 is alive. 5 is alive.
c.) 4 is dying.
d.) 4 is alive. 4 is dying.
e.) None of the above.

32.) What is ***logic*** **error in the following code? -** _______

```
int foo( int a )
{
      return a + foo( a - 1 );
}
```

a.) Should be pass by reference.
b.) No termination condition.
c.) Return type is incorrect.
d.) Iteration is unnecessary.
e.) None of the above.

33.) **What is the *syntax* error in the following code? -** _______

```
int main()
{
     int size = 5;
     int a[ size ];
     a[ 4 ] = 10;
     cout << *a << endl;
     return 0;
}
```

a.) Can’t dereference constant pointer.
b.) Must declare array size with a literal integer value.
c.) Size must be constant.
d.) Index is out of bounds.
e.) None of the above.

34.) **What is the *syntax* error in the following code? -** _______

```
int x = 0;
do
{
     cout << x << endl;
} while ( x < 0 )
```

a.) This code will never run.
b.) While loop syntax is incorrect.
c.) X does not have scope in the loop.
d.) Missing ;.
e.) None of the above.

35.) **Given a class called MyClass, what is the *syntax* error in the following code? -**

```
MyClass operator@( MyClass &x )
{
    return MyClass( x.get_Value() );
}
```

a.) Can’t return nameless object.
b.) Returning by value causes the copy constructor to run.
c.) Cannot overload this operator.
d.) No call to left-hand value in definition.
e.) None of the above.

36.) What is a friend? - ________

a.) A class that has access to data in a parent class.
b.) A class that has access to data in another class.
c.) A class that has access to data in a child class
d.) A class that has access to data in a static class.
e.) None of the above.

37.) What is the output from the following code? - ________

```
int a = 3, b = 3, c = 1;
a += b - c;
b += a * c;
c += a - b;
cout << a << " " << b << " " << c << endl;
```

a.) 3 4 0
b.) 5 8 -2
c.) 5 10 0
d.) 3 5 -5
e.) None of the above.

38.) What is the output from the following code? - ________

```
int x = 17, y = 7;
while ( x > 0 )
{
  y += 3;
  x -= 5;
}
cout << x << " " << y << endl;
```

a.) -1 19
b.) -1 31
c.) -3 19
d.) -2 19
e.) None of the above.

39.) **Given the following two functions: -** ________

```
int f( int x )
{
	return x + 2;
}
int g( int x )
{
	return x * 2;
}
```

What is the output from the following code?

```
int x = 1;
x += f( g( x ) ) + g( f( x ) );
cout << x;
```

a.) 11
b.) 3
c.) -1
d.) 9
e.) None of the above.

40.) **What is the output from the following code? -** ________

```
int main()
{
        int a = 1;
        int b = 1;
        while ( b < 3 )
        {
                switch ( a )
                {
                        case 0:
                                b--;
                        case 1:
                                a--;
                                break;
                        case 3:
                                b++;
                        case 4:
                                a++;
                                break;
                        default:
                                b++;
                }
                cout << b << " ";
        }
        return 0;
}
```

a.) 1 0 1 3
b.) 1 0 1 2 3
c.) 1 1 2 3
d.) 1 0 1 1 2 3
e.) None of the above.

41.) **Polymorphism means basically: -** ________

a.) Overloaded definitions at the base class level.
b.) Derived classes inherit base class definitions.
c.) A group of methods with many implementations.
d.) Function overloading as well as overriding.
e.) None of the above.

42.) **After including cstdlib, what is the output from the following code? - ________**

```
int main()
{
     int a = 1234;
     char *s = "2309";
     printf( "%s != %i\n", s, a );
     double d = atof( "3245" );
     printf( "%s != %e\n", s, d );
     return 0;
}
```

a.) Compile error.
b.) true true
c.) 2309 != 1234
    2309 != 3245
d.) 2309 != 1234
    Runtime error.
e.) None of the above.

43.) **What is the output from the following code? - ________**

```
int foo( int [], int, int (*)( int, int ) );
int bar( int, int );

int main()
{
     int a[] = { 6, 1, 7, 3, 8 };
     cout << foo( a, 5, bar ) << endl;
     return 0;
}

int foo( int a[], int c, int ( *f )( int, int ) )
{
     int v = 0;
     for( int i = 0; i < c; i++ )
        v += f( a[ i ], c );
     return v;
}

int bar( int a, int b )
{
     return a - b;
}
```

a.) 8
b.) 0
c.) 50
d.) 1
e.) None of the above.

44.) Ignoring prototypes, what is the output from the following code? - ________

```
int main()
{
      int b = 2;
      cout << foo( b ) << endl;
      cout << b << endl;
      return 0;
}


int day( int b )
{
      b++;
      cout << bar( &b ) << endl;
      return b;
}

int foo( int &b )
{
      b++;
      day( b );
      return b;
}

int bar( int *b )
{
       *b += *b;
       return *b;
}
```

a.) 8 3 2
b.) 3 3 3
c.) 3 3 2
d.) 8 3 3
e.) None of the above.

45.) What is the output from the following code? - ________

```
int main()
{
    int a = 37;
    for( int b = 128; b > 0; b /= 2 )
    {
        if ( a & b )
             cout << "1 ";
        else
             cout << "0 ";
    }
    cout << endl;
    return 0;
}
```

a.) 00100101
b.) 01000011
c.) 00011111
d.) 00101111
e.) None of the above.

46.) What is the point of the following code? - ________

```
#define Par 1000.00

enum instr_type { Stock,Option,Future,Bond };
typedef double price;
struct Tick { price Price; instr_type Type; };

int main()
{
    Tick a;
    a.Price = Par;
    a.Type = Bond;

    return 0;
}
```

a.) To obfuscate the code (i.e. to make it more confusing).
b.) To enable object oriented design.
c.) To make it more human-readable.
d.) To speed up execution of the code.
e.) None of the above.

47.) What is the output from the following code? - ________

```
char name[] = "John Doe";
int n = 0;
for( ; name[ n ] != '\0' ; )
{
      cout << name[ n ] << endl;
      n++;
}
```

a.) John Doe, horizontally.
b.) John Doe, vertically.
c.) Compile error.
d.) John, vertically.
e.) None of the above.

48.) What is the point of the following code? - ________

```
typedef float price;
price prices[];
```

a.) To create a two dimensional array of floats.
b.) To create an array of prices.
c.) To declare price as a variable of type float.
d.) To define price as a pointer to an array.
e.) None of the above.

49.) What is the output from the following code? - ________

```
#define IBM "IBM"

int main()
{
      cout << IBM << endl;
      return 0;
}
```

a.) true
b.) Compile error.
c.) IBM
d.) 0
e.) None of the above.

50.) **What is the output from the following code? -** ________

```
enum Pos
{
    LONG,SHORT
};

Pos p;
p = ( 3 > 2 ? SHORT : LONG );
cout << p << endl;
```

a.) 0
b.) 1
c.) LONG
d.) SHORT
e.) None of the above.