Question 1

What statement is true about the following code?

Code (C++):

2. namespace
3. {
4. int x=20;
5. }

1.  This does not compile because the namespace has no name.
2.  This code defines a local variable *x* in its own scope.
3.  The variable *x* is a global variable only accessible in the current compilation unit and not as external global variable in another compilation unit.
4.  Variable *x* is inaccessible because the namespace has no name.

Question 2

Which statement is true about the following code?

Code (C++):

2. using namespace A::B;

1.  After this statement all classes in the namespace *A::B* can be used without specifying the namespace name.
2.  This statement cannot appear inside functions, only at the beginning of your file before any functions.
3.  After this statement, classes in the specified namespace cannot be referenced anymore using their full namespace name.
4.  After this statement, class *B* in namespace *A* can be used without specifying the namespace name.
5. You answered this question correctly.

Question 3

Which of the options below is the best operator declaration to add a two objects of type*Complex* (*Complex+Complex*)?

1.  Complex& operator + (const Complex& c1, const Complex& c2) const;
2.  Complex operator + (const Complex& c) const;
3.  Complex operator + (const Complex& c);
4.  Complex& operator + (const Complex& c) const;
5. You answered this question correctly.

Question 4

What is the output of the following code?

Code (C++):

2. int size=3; int\* a=new int[size];
3. for (int i=0; i<size; i++) a[i]=10-i;
4. std::cout<<a[1]<<", "<<\*a<<", "<<(a+1)[0]<<", "<<\*a+1<<std::endl;
5. delete[] a;

1.  9, 10, 9, 11
2.  9, [address of variable *a*], 9, [address of variable *a* + *sizeof(int)*]
3.  9, 10, 9, 9
4.  9, 10, 11, 9
5. You answered this question correctly.

Question 5

Which of the options below is the best description of a canonical header file?

1.  A header file with minimal a constructor, destructor, equal compare operator and not equal compare operator.
2.  A header file with minimal a default constructor, copy constructor, destructor and equal compare operator.
3.  A header file with minimal a constructor, destructor, assignment operator and equal compare operator.
4.  A header file with minimal a default constructor, copy constructor, destructor and assignment operator.
5. You answered this question correctly.

Question 6

Which statement is false about operator overloading?

1.  You can overload unary, binary and ternary operators.
2.  The input of an operator can be different than the class type.
3.  Operator functions can be a member function or global function.
4.  Operator functions generally don't change the operator arguments except the assignment operators.

Question 7

Which statement is false about namespaces?

1.  A namespace must be compiled in its own *.lib* file.
2.  Namespaces can be nested.
3.  Multiple namespace blocks with the same name are possible.
4.  Namespaces can prevent name collisions or be used to group functionality in logical blocks.

Question 8

Which of the following options declares and allocates an array of *int* pointers?

1.  int\*\* array=new int\*[size];
2.  int\* array=new int\*[size];
3.  int[] array=new int\*[size];
4.  int\*[] array=new int\*[size];

Question 9

Which statement is false about memory?

1.  *new* and *malloc* allocate memory on the heap.
2.  The size of an array on the stack can only be determined at compile time.
3.  Local variables are allocated on the stack.
4.  When using *new* to allocate memory, you need to pass the number of bytes to allocate.

Question 10

Which of the options below is the best operator declaration to add and assign two objects of type *Complex* (*Complex+=Complex*)?

1.  void operator += (const Complex& c);
2.  Complex& operator += (const Complex& c) const;
3.  Complex& operator += (const Complex& c);
4.  Complex operator += (const Complex& c);
5. You answered this question correctly.

Question 8

Which statement is false about friends?

1.  Friend functions can access the '*this*' pointer.
2.  A class cannot access the private members of its friend classes.
3.  Friend functions can access the private members of the class they are friend of.
4.  Friends violate the information hiding principle of object-oriented programming.

Question 5

What statement is true about the following code?

Code (C++):

2. namespace A::B
3. {
4. class MyClass
5. {
6. };
7. }

1.  This code does not compile.
2.  This code defines a class in namespace *B* that is nested in namespace *A*.
3.  This code defines a class in the single namespace called *A::B*.
4.  This code defines a class in namespace *A* that is nested in namespace *B*.

Question 4

Which of the options below is the best operator declaration to add a *double* and a*Complex* object (*double+Complex*)?

1.  friend Complex operator + (double d, const Complex& c);
2.  friend Complex operator + (double d, const Complex& c) const;
3.  Complex operator + (double d) const;
4.  friend Complex& operator + (double d, const Complex& c);

Question 2

Which of the operator declarations below is the best way to support the index operator for integer indices (*[int]*)?

1.  Type operator [] (int index) const;
2.  const Type& operator [] (int index) const; Type& operator [] (int index);
3.  Type& operator [] (int index) const;
4.  const Type& operator [] (int index) const; Type operator [] (int index);