$Id: cmps109-2014q3-exam2.mm,v 1.71 2014-08-12 14:49:06-07 - - $

page 1 page 2 page 3 page 4 page 5 Total/54

Please print clearly:

Name:

Login: @ucsc.edu

No books; No calculator; No computer; No email; No internet; No notes; No phone. Neatness counts! Do your scratch work elsewhere and enter only your final answer into the spaces provided. Assume the "`using namespace std;`" directive.

1. Rewrite the following statement using the two-semicolon form of `for`-loop and iterators instead of the colon form: `for (auto i& : c) f(i);` **[1✔]**

2. The following table contains fragmentary examples of various kinds of polymorphism. For each example: identify the general kind of polymorphism by writing one of the words ***universal*** or ***ad hoc***; also identify the specific kind of polymorphism by writing one of the words ***conversion***, ***inheritance***, ***overloading***, or ***template***. **[2✔]**

`class foo {virtual void f(); }`
`class bar: public foo {`
`  virtual void f();`
`}`
	`class stack<item_t> {`
	`  void push (item_t);`
	`}`

`int add (int x, int y);` `double add (double x, double y);`	`double f (double x);` `f (6);`

3. Write a `main` function that reads in (white-space delimited) words from `cin` using `>>`. The input file has two words per line, the first is a key and the second is its value. Put each key/value pair into a map. At end of file, print each key and value pair on a line by itself, with the keys sorted into lexicographic order. **[3✔]**

4. Code the template function `merge`, which merges two sorted ranges into a single output range. It has two template arguments: an iterator type capable of iterating over the input ranges, and a container type with a `push_back` operation for the output container. For example, if we have two sorted vectors `v1` and `v2` and wish to merge them into an output vector `vo`, we could use the statement

```
merge (v1.cbegin(), v1.cend(), v2.cbegin(), v2.cend(), vo);
```

Assume the elements can be compared with `operator<` and that they are sorted into ascending order. **[4✔]**

5. Write the function `draw_square`, which will draw a square (as a `GL_POLYGON`) in an OpenGL window, given the (x, y) position of the lower left corner, and the width and color of the square. The edges of the square should be parallel to the edges of the window. **[3✔]**

```
void draw_square (GLfloat xpos, GLfloat ypos, GLfloat width, GLubyte* color) {
```

6. Define three classes: `shape`, which is entirely abstract; `rectangle`, non-abstract, which inherits from `shape`; and `square`, which inherits from `rectangle`. Assume there are many other things that might inherit from `shape`, but they are not part of this questions.
   (i) The ctor for `shape` should not be accessible except to its subclasses.
   (ii) If we declare `shape* p`, then `p->area()` and `p->perimeter()` should compute the values required.
   (iii) Do not duplicate code when defining `area` and `perimeter`, so make them abstract, virtual, inherited, etc., as appropriate.
   (iv) Use field initializers as appropriate. `Rectangle` should keep its length and width information.

   Complete the following classes:

   (a) **[2✔]** `class shape`

   (b) **[2✔]** `class rectangle`

   (c) **[1✔]** `class square`

7. Assuming that the previous question defined the virtual abstract function `print` in class `shape`, define `operator<<` so that an arbitrary object of type `shape*` can be printed. Example: The following two statements should print a shape. **[2✔]**

```
shape *p;
cout << p << endl;
```

8. Given a class `complex` outlined here, with real and imaginary fields, show what would go in the header file to define:
   (a) A friend `operator<<` which can be used to print it. **[1✔]**
   (b) A member binary `operator*` which multiplies two complex numbers. **[1✔]**

```
class complex {
   private:
      double real;
      double imag;
   public:
```

9. For the previous question, show what would go into an implementation file for the two definitions above:
   (a) `operator<<` prints a complex number in parentheses with the fields separated by a comma. The imaginary part is suffixed with the letter "`i`". For example: `(8.3,4.9i)`. **[2✔]**
   (b) Some mathematics for `operator*`: **[2✔]**
   $(a + bi)(c + di) = (ac - bd) + (ad + bc)i$, where $i = \sqrt{-1}$.

10. Write the body of a function `inner_product`, which has two arguments of type `vector<double>` and returns the inner product, which is a `double`. **[2✔]**
    The formula for the inner product of two vectors u and v is $p = \sum_{i=0}^{n-1} u_i v_i$.

11. Write a complete template function whose argument is a `vector` and which prints out all the elements of the vector, with one space between successive elements. There is no space before or after the sequence. Assume `operator<<` has been defined for the elements of the vector. **[2✔]**

Multiple choice. To the ***left*** of each question, write the letter that indicates your answer. Write ***Z*** if you don't want to risk a wrong answer, which count negative points. **[12✔]**

number of correct answers		× 1 =	= a
number of wrong answers		×½ =	= b
number of missing answers		× 0 =	0
column total $c = \max(a-b, 0)$	12		= c

1. If `p` is an iterator, which expression will return the address of the object that it refers to?
   (A) `&&p`
   (B) `&*p`
   (C) `*&p`
   (D) `**p`

2. Which of the following statements will change meaning if the parentheses are removed?
   (A) `a = (b);`
   (B) `double d();`
   (C) `int (i);`
   (D) `return (x);`

3. Which one of these operators may be defined with an arbitrary number of arguments?
   (A) `operator()`
   (B) `operator<<`
   (C) `operator==`
   (D) `operator[]`

4. For `vector<int> v`, the call to `v.crbegin()` will return an iterator pointing at:
   (A) `v[-1]`
   (B) `v[0]`
   (C) `v[v.size()-1]`
   (D) `v[v.size()]`

5. For `vector<int> v`, the call to `v.begin()` will return an iterator pointing at:
   (A) `v[-1]`
   (B) `v[0]`
   (C) `v[v.size()-1]`
   (D) `v[v.size()]`

6. Which is a `Makefile` comment?
   (A) `# comment`
   (B) `-- comment`
   (C) `/* comment */`
   (D) `// comment`

7. What category of membership in a class allows a member to be accessed by the class itself and also by any class derived from it, but by no other class?
   (A) `friend`
   (B) `private`
   (C) `protected`
   (D) `public`

8. Which is a declaration of an abstract function?
   (A) `const void virtual f();`
   (B) `virtual void f() = 0;`
   (C) `virtual void f() override;`
   (D) `void f() const virtual;`

9. An IP address such as 128.114.108.152 is an example of what address family passed to the `socket`(2) system call?
   (A) `AF_INET`
   (B) `AF_INET6`
   (C) `AF_IPX`
   (D) `AF_UNIX`

10. What initializer should be appended to the statement
    `const GLubyte GREEN[] =`
    (A) `{0x00, 0x00, 0xFF};`
    (B) `{0x00, 0xFF, 0x00};`
    (C) `{0x00, 0xFF, 0xFF};`
    (D) `{0xFF, 0x00, 0x00};`

11. In the linked list implementation of project 3, if there are n data items in the list, then how many occurrences will there be of `struct link`?
    (A) $O(1)$
    (B) $O(n-1)$
    (C) $O(n)$
    (D) $O(n+1)$

12. In the library implementation of class `list`, how long does it take to access the n^{th} element of a list?
    (A) $O(1)$
    (B) $O(\log_2 n)$
    (C) $O(n)$
    (D) $O(n \log_2 n)$

Multiple choice. To the ***left*** of each question, write the letter that indicates your answer. Write ***Z*** if you don't want to risk a wrong answer, which count negative points. **[12✔]**

number of correct answers		× 1 =	= a
number of wrong answers		×½ =	= b
number of missing answers		× 0 =	0
column total $c = \max(a - b, 0)$	12		= c

1. The keyword `override` is used in the definition of what kind of function?
   (A) `const`
   (B) `static`
   (C) `template`
   (D) `virtual`

2. The highest AF_INET port number is:
   (A) 255
   (B) 65535
   (C) 16777215
   (D) 4294967295

3. When two threads update the same variable without synchronization, the result is a:
   (A) deadlock
   (B) livelock
   (C) memory leak
   (D) race condition

4. Which is a move constructor?
   (A) `T(T&&);`
   (B) `T(T**);`
   (C) `T(const T&&);`
   (D) `T(const T**);`

5. If `Foo` is an abstract class, and we have the declarations: `Foo* p; Foo f;`
   Then the declaration of:
   (A) `p` is good, `f` is an error.
   (B) `p` is an error, `f` is good.
   (C) both are good.
   (D) both are errors.

6. If an object is called as a function, what operator must it have defined?
   (A) `operator()`
   (B) `operator*`
   (C) `operator->`
   (D) `operator[]`

7. Which container allows direct access via `operator[]` and also allows the push and pop functions at both the front and back in $O(1)$ time?
   (A) `deque`
   (B) `list`
   (C) `stack`
   (D) `vector`

8. After starting a thread with a command like `thread t1(hello);` how does the calling function then suspend until `t1` finishes?
   (A) `t1.catch();`
   (B) `t1.join();`
   (C) `t1.pause();`
   (D) `t1.wait();`

9. Which member of a class must do different things, depending on whether its left operand is the same as its right operand or not?
   (A) copy constructor
   (B) copy `operator=`
   (C) default constructor
   (D) `operator==`

10. The keywords `class` and `struct` mean the same thing, except that by default, members of a `class` are __(x)__ and members of a `struct` are __(y)__.
    (A) (x) = private, (y) = private
    (B) (x) = private, (y) = public
    (C) (x) = public, (y) = private
    (D) (x) = public, (y) = public

11. To convert a `sockaddr_in*` to a `sockaddr*`, which are two different types unrelated by inheritance, what kind of cast is used?
    (A) `const_cast`
    (B) `dynamic_cast`
    (C) `reinterpret_cast`
    (D) `static_cast`

12. Given a pair of variables, an operation to swap (exchange) their values will be done efficiently by a:
    (A) copy constructor
    (B) copy `operator=`
    (C) move constructor
    (D) move `operator=`