In class Practice 11/8/2019

Group three or four people team.

Part I

1. The operator * is prefixed to an iterator to
   A) Multiply the element in the container
   B) Extract the element in the container to assign to it only
   C) Extract the element in the container to fetch its value only
   D) Extract the element in the container as either an l-value or an r-value
2. Which of the following operations do forward iterators have?
   A) Overloaded `operator+` to add an `int` value to the iterator to move the place the iterator points forward by the argument number of elements.
   B) Overloaded `operator*` to multiply the iterator by an `int` value to move the place the iterator points by a number of elements equal to the argument.
   C) Overloaded `operator++` to move the place the iterator points forward by one element.
   D) Overloaded `operator--` to move the place the iterator points backward by one element.
3. Which of the following member functions is NOT common to the sequential containers (`vector`, `list`, `deque`)?
   A) `begin()`
   B) `rbegin()`
   C) `rend()`
   D) `push_front()`
   E) `front()`
4. Which of the following is an incorrect declarations of iterators for STL containers? You may assume that the proper header has been included and that a using directive makes the names from namespace std available.
   A) `vector<int>::iterator vecIterator;`
   B) `list::iterator<int> listIterator;`
   C) `deque<int>::iterator dequeIterator;`
   D) `list<int>::iterator listIterator;`
5. Which of the following is not a member function of the `queue` adapter template? For members of `queue`, specify any needed arguments.
   A) `size()`
   B) `empty()`
   C) `front()`
   D) `push()`
   E) `top()`
6. Which of the following is not a member function of the `stack` adapter template? For members of `stack`, specify any needed arguments.
   A) `size()`
   B) `empty()`
   C) `front()`
   D) `push()`
   E) `top()`
7. In which container does the position of an inserted element depend on the data, not the order of insertion?
   A) Associative containers
   B) Fraternal container
   C) Sequence containers
   D) Container adapters
8. Given the following definition for a map, which code fragment is valid?
   map<int, string> mymap;
   A) mymap[3, "hello"] = 10;
   B) mymap.push_back(Pair(3, "hello"));
   C) mymap[10] = "hello";
   D) mymap["hello"] = 3;
9. Assume proper includes have been executed, but no `using` directive or declaration. Write a definition of an `iterator` for a `vector` of `int`s that is initialized to point to the first member of the `vector` `vec`.

   vector<int> :: iterator itr = vec.begin();

Part II

Answer the following questions

1. The lists are singly linked lists and offer rapid insertion and deletion anywhere.
2. Associative containers are nonlinear data structures that typically can locate elements stored in the containers quickly.
3. The container member function cbegin returns an iterator that refers to the container's first element.
4. The ++ operation on an iterator moves it to the container's next element.
5. Many algorithms operate on sequences of elements defined by iterators pointing to the first element of the sequence and to the last element.
6. A Map is a function given as a set of ordered pairs. The first is the key that has to have ordering and the second is any type. The position of a pair in the set is determined by the ordering on the keys.

1.false they are doubly linked lists

2.true

3.false const_iterator

4.true

5. the one element past the last element

6.true

Part III

Write a function template palindrome which takes a vector parameter and return true if the vector read the same forward and backward. For example, if vector contains 1, 2, 3, 2, 1 then it is a palindrome. If a vector contains 1, 3, 3, 1then it is also a palindrome. A vector with 3, 5, 7, 9 is not so will return false.