SRN | | | | | | | | | | | | |

PES University, Bengaluru

(Established under Karnataka Act No. 16 of 2013)

MAY- 2019: END SEMESTER ASSESSMENT (ESA) B.TECH. VI SEMESTER
UE16CS335 – GENERIC PROGRAMMING

Instructions: You may bring any written matter. You cannot have any printed matter. You cannot exchange any material during the exam.

Time: 3 Hr | Answer All Questions. Neglect syntax errors | Max Marks: 100

1. a) (6)

```
#include <iostream>
using namespace std;
template<typename ptr_t>
void myswap(ptr_t p, ptr_t q)
{
        //TODO
}
int main()
{
        int x{10}; int y{20};
        myswap(&x, &y);
        cout << x << " " << y << "\n";
}
```

hint: how to create a local variable for swapping? Helper function?

b) (6) Write a template function to check whether a given sequence is sorted in ascending order.

```
template<typename ptr_t>
bool is_sorted(ptr_t first, ptr_t last)
{
      // TODO
}
```

c) (2+2)

```
#include <iostream>
using namespace std;
template<typename T>
bool eq(T x, T y)
{
        return x == y;
}
int main()
{
        cout << boolalpha;
        char a[] = "pes";
        char *b = a;
        char c[] = "pes";
        cout << eq(a, b) << "\n";
        cout << eq(a, c) << "\n";
}
```

Find the output. Give reasons.

d) (2+2)

```
template<typename T>
T what(T x)
{
        return x * x;
}
```

SRN | | | | | | | | | | | | |

What if the function is instantiated as follows? Why?

```
int a[] = {5, 3, 1, 2, 4};
cout << what(a) << "\n";
cout << what(*a) << "\n";
```

2. a) Find the output in each case. Justify your answer in a single sentence. 6

```
template <class T>
T mymax (const T &a, const T &b)
{
        cout << "one ";
  return (a > b)? a : b;
}
 template <>
int mymax <int> (const int &a, const int &b)
{
  cout << "two ";
  return (a > b)? a : b;
}
template <>
long mymax <long> (const long &a, const long &b)
{
  cout << "three ";
  return (a > b)? a : b;
}
template <>
string mymax(const string &a, const string &b)
{
  cout << "four ";
        return (a > b)? a : b;
}
int main ()
{
  int a = 10, b = 20;
        string c = "apple";
        string d = "banana";
  cout << mymax <int> (a, b) << "\n";
        cout << mymax (10l, 20l) << "\n";
        cout << mymax <long> (a, b) << "\n";
        cout << mymax <char> (c[0], d[0]) << "\n";
        cout << mymax(c, d) << "\n";
        cout << mymax(c.c_str(), d.c_str()) << "\n";
}
// c_str() returns a  c string
```

b) Find the output. Please note that default parameters are allowed for template functions. 6

```
template <typename T = int, int count = 3>
T foo(T x)
{
        for(int ii = 0; ii < count; ii++)
        {
                x = x * x;
        }
        return x;
};
```

SRN | | | | | | | | | | | |

```
int main()
{
        int x(2);
        cout << x  << ": " << foo<>(x) << endl;
        cout << x  << ": " << foo<int, 2>(x) << endl;
        cout << 2.5  << ": " << foo<double, 2>(2.5) << endl;
}
```

c)

```
template<typename T1, typename T2>
struct mypair
{
    public:
    mypair(T1 f, T2 s);
    private:
    T1 first; T2 second;
};
```

I) write the constructor for this class.

ii)

```
template<typename T1, typename T2>
mypair<T1, T2> make_mypair(T1 f, T2 s)
{
    // TODO
}
```

Complete the code to create an object of mypair. 2+ 2

d)

```
template<typename T>
struct What
{
    public:
    What(int n) : n(n), p(new int[n]) { }
    private:
    T *p;
    int n;
};
```

Develop

I) move assignment

ii) move constructor for this class. 2+2

3. a) Find the output.

```
int main ()
{
    vector<int> myvector(10);
    iota(begin(myvector), end(myvector), 1);
        sort(begin(myvector), end(myvector), greater<int>());
        sort(begin(myvector) + myvector.size() / 2, end(myvector),   less<int>());
        for_each(begin(myvector), end(myvector), [](auto e){cout << e << "\t"; });
        cout << "\n";
        cout << *find_if(begin(myvector), end(myvector),
        [myvector](auto e) { return e < myvector.size() / 2; }) << "\n";
        cout << count_if(begin(myvector), end(myvector),
                [myvector](auto e) { return e < myvector.size() / 2; }) << "\n";
}
```

6

b) Comment precisely about these codes.

i) `for_each(begin(c), end(c), [](auto e){ e += 10; }`

ii) `cout << *find(begin(c), end(c), e) << "\n";`

iii) `vector<int> v{1, 2, 3, 4, 5}; list<int> l; copy(begin(v), end(v), begin(l));`

6

c) Write a template meta program for the following.

I) check whether the given integer constant is odd

ii) find the number of digits of a given integer constant

4 + 4

SRN | | | | | | | | | | | |

4	a)	Given a vector of strings, write the relevant code to create a set of strings not necessarily unique ordered based on the last character of each string.	6
	b)	Given a multimap of state and its cities, write the relevant code create a map of state and # of cities.	6

	c)	We would like to search for occurrence of a given name in a sequence of structure Emp having the following structure. Write the relevant code
		Struct Emp
		{
		char name[20];
		int age;
		};
		template<typename ptr_t>
		Emp* my_search(ptr_t first, ptr_t last, const string& s)
		{
		/ / TODO
		}
		hint: use find_if with lambda
			4

	d)	What happens if algorithm sort is called on a list? Why? How can we sort a list?	4
5	a)	iterator_traits has the following member types. difference_type value_type pointer reference iterator_category input_iterator_tag output_iterator_tag forward_iterator_tag bidirectional_iterator_tag random_access_iterator_tag Given an iterator of type ptr_t, write code for the following. i) Create a variable of the value type. Do not use auto. ii) Find the iterator category and display a string indicating the category. Hint: use typeid to compare types of objects.	2 + 4

5	b)	All containers provide == operator. This would work as long as the component type supports == operator.
		Develop this algorithm.
		template<typename C> // c1 and c2 are containers of type C
		bool operator==(const C& c1, const C& c2)
		{
		// TODO
		}
			4

5	c)	I) Why all pop functions in STL are void functions? ii) Why cannot stack be implemented by using vector by inheritance? Iii) Why the complexity of size method list could be linear amortized complexity? iv) How does set.find work? Does it require == operator on the component type? v) Why queue cannot be implemented using vector?	10