Q1.1: What is the second character of the first argv table element?

CORRECT ANSWER

[0][1]

Q2.1Upon successful completion of this workshop, you will have demonstrated the abilities to

work with objects of the string class

declare and use enumeration constants

use the correct constant type in initializations

retrieve data from and backup data to a text file

move data between unsigned and signed integers

CORRECT ANSWER

fstream os(filename, ios::out | ios::trunc)

Q3.1: The workshop opens a file by a fstream is(filename, ios::in); fstream constructor call.

The program reads the file twice.

The first pass counted the nummber of lines on the file.

The program allocated a string table of the required size.

The second pass read the lines into the string table.

The first pass reads the file until it encountered an EOF (end-of-file) state.

How do you reset the EOF state so the file can be read in again in the second pass?

CORRECT ANSWER

is.clear()

is.seekg(0)

Q4.1 Is it possible to accomplish the same thing as Workshop 4 if you code as a table of KV-pairs?

template<typename K, typename V,int N>

class KVList2 {

struct {

K k;

V v;

} KVPair[N];

int count;

pubic:

...

};

CORRECT ANSWER

True

Q5.1 This workshop uses composition.

There are two classes: class Message and the MessageManager (class Notifications).

Is workshop 2, the TTC workshop (class Station and the StationManager, class Stations) also a composition?

CORRECT ANSWER

True

Q6.1 Can lambas defined locally (or inside) a function be viewed as increasing O-O encapsulation?

CORRECT ANSWER

True

Q7.1 What does this code print?

try {

auto X = [] { throw nullptr; }

auto Y = [X] { X(); };

auto Z = [Y] { Y(); };

Z();

} catch (const char\* e) {

cerr << "something threw a char\* exception\n";

} catch (const std::string\* e) {

cerr << "something threw a string exception\n";

} catch (...) {

cerr << "something threw a ... exception\n";

}

CORRECT ANSWER

something threw a ... exception\n

Q8.1

A.

T sum = initialvalue;

for(auto it = begin, it != end, it++)

sum += \*it;

return sum;

B.

T sum = initialvalue;

for(auto it = begin, it != end, it++)

sum += \*it \* \*begin2++;

return sum;

C.

size\_t hits = 0;

for(auto it = begin, it != end, it++)

if(\*it == Value)

hits++;

return hits;

D.

for(auto it = begin, it != end, it++)

if(\*it == Value)

return it;

return end;

A -> T accumulate(iterator begin, iterator end, T initialValue)

B -> T inner\_product(iterator begin, iterator end, iterator begin2, T initialValue)

C -> size\_t count count(iterator begin, iterator end, T Value)

D -> iterator find(iterator begin, iterator end, T Value)

Q9.1 Quadratic complexity components usually consist of a number nested for loops. How many?

CORRECT ANSWER

2

Q10.1 How many threads were launched in Workshop 10?

CORRECT ANSWER

1 OR MORE