

Capítulo 17

This activity contains 22 questions.

1.

Section 17.2 The Data Hierarchy

17.2 Q1: Select the false statement.

- ☐ c. It is cumbersome to work with data in bit form.
- ☐ It is easiest to produce electronic devices that can assume one of two stable states.
- ☐ A binary digit (bit) can store two values simultaneously.
- ☐ All data items can eventually be reduced to a sequence of 0s and 1s.

2.

17.2 Q2: Which statement is not true about 0, z, %, * and Q?

- ☐ Each is a character.
- ☐ Only two of the elements listed are in typical character sets.
- ☐ Each can be represented as a byte.
- ☐ Each can be represented in terms of 0s and 1s.

3.

17.2 Q3: The data hierarchy, arranged from smallest to largest, is:

- ☐ Bit, byte, field, record, file.
- ☐ Bit, byte, record, field, file.
- ☐ Byte, bit, field, file, record.
- ☐ Byte, bit, field, record, file.

4.

17.2 Q4: In general, which of the following contains the most amount of data?

- ☐ A byte.
- ☐ A field.
- ☐ A file.
- ☐ A database.

5.

Section 17.3 Files and Streams

17.3 Q1: Which of the following does not have a stream associated with it?

- ☐ *cin.*
- ☐ *cerr.*
- ☐ *cout.*
- ☐ *<iostream>.*

6.

17.3 Q2: In order to perform file processing in C++, which header files must be included?

- ☐ *<iostream> and <fstream>.*
- ☐ *<cstdio> , <iostream> and <fstream>.*
- ☐ *<cstdio> and <fstream>.*
- ☐ *<cstdio> and <iostream>.*

7.

17.3 Q3: Which of the following is not true about files?

- ☐ *Files are opened by creating objects of stream classes.*
- ☐ *istream, ostream and iostream are derived from ifstream, ofstream and fstream, respectively.*
- ☐ *C++ views each file as a sequential stream of bytes.*
- ☐ *Member functions of stream objects can be applied to file streams.*

8.

Section 17.4 Creating a Sequential File

17.4 Q1: Select the false statement.

- ☐ *d. C++ files do not understand notions such as "records" and "fields."*
- ☐ *C++ files include information about their structure.*
- ☐ *C++ imposes no structure on a file.*
- ☐ *The programmer must impose a structure on a file.*

9.

17.4 Q2: Which file open mode would be used to write data only to the end of an existing file?

- ☐ *ios::app.*

- ☐ *ios::out.*
- ☐ *ios::trunc.*
- ☐ *ios::in.*

10.

17.4 Q3: When used with ofstream objects, operator! is not:

- ☐ Used to return a nonzero value if an error occurs.
- ☐ Used to determine if the open operation succeeded.
- ☐ Overloaded.
- ☐ Used to close a file explicitly.

11.

Section 17.5 Reading Data from a Sequential File

17.5 Q1: Which of the following will not change the file-position pointer to the same position as the others? Assume a 10-byte file size and a current position at byte # 1.

- ☐ *fileObject.seekg(8, ios::end);*.
- ☐ *fileObject.seekg(1, ios::cur);*.
- ☐ *fileObject.seekg(2);*.
- ☐ *fileObject.seekg(2, ios::beg);*.

12.

17.5 Q2: What is not true about this code segment?

```
location = fileObject.tellg();
```

- ☐ The value of location after the segment executes must be less than or equal to the number of bytes in the file attached to fileObject.
- ☐ location is a pointer.
- ☐ fileObject is an istream object.
- ☐ tellg is a member function of fileObject.

13.

Section 17.6 Updating Sequential Files

17.6 Q1: Which of the following is not a disadvantage of trying to modify a sequential access file?

- ☐ Overwriting a record with another record of the same size is very difficult.

- ☐ *Modifying data can potentially destroy other data.*
- ☐ *Things that are stored in the same number of "raw data" bytes internally may not take up the same amount of space in a file.*
- ☐ *It may be necessary to modify every record in the file to make a slight change.*

14.

Section 17.7 Random-Access Files

17.7 Q1: Random access files are more effective than sequential files for:

- ☐ *All of the above.*
- ☐ *Instant access to data.*
- ☐ *Inserting data into the file without destroying other data.*
- ☐ *Updating data easily.*

15.

17.7 Q2: A random access file is organized most like a(n):

- ☐ *Object.*
- ☐ *Class.*
- ☐ *Pointer.*
- ☐ *Array.*

16.

Section 17.8 Creating a Random-Access File

17.8 Q1: Select the false statement. The write function:

- ☐ *Writes to files in hexadecimal format.*
- ☐ *Creates unformatted data.*
- ☐ *Takes an argument of type size_t.*
- ☐ *Expects data type const char * as its first argument.*

17.

17.8 Q2: Select the false statement. The reinterpret_cast operator:

- ☐ *Performs its operation at compile time.*
- ☐ *Is compiler-dependent and can cause programs to behave differently on different platforms.*
- ☐ *Changes the value of the object to which its operand points.*

- ☐ *Is easy to use to perform dangerous manipulations that could lead to serious execution-time errors.*

18.

17.8 Q3: To write fixed-length records, use file open mode:

- ☐ *ios::app.*
☐ *ios::trunc.*
☐ *ios::ate.*
☐ *ios::binary.*

19.

Section 17.9 Writing Data Randomly to a Random-Access File

17.9 Q1: Which of the following functions would not be used to write data randomly to a random access file?

- ☐ *All of the above would be used.*
☐ *seekp.*
☐ *tellg.*
☐ *write.*

20.

Section 17.10 Reading from a Random-Access File Sequentially

17.10 Q1: For an ifstream object A, a class type B and a local variable of type B called C, the proper way to read in one B object from A into C is:

- ☐ *A.read(&C, sizeof(B));.*
☐ *A.read(char * (&C), sizeof(B));.*
☐ *A.read(reinterpret_cast< char * >(&C), B);.*
☐ *A.read(reinterpret_cast< char * >(&C), sizeof(B));.*

21.

Section 17.11 Case Study: Transaction-Processing Program

17.11 Q1: Select the proper object type.

_____ `file("file.dat", ios::in | ios::out);`

- ☐ *iostream.*
☐ *ofstream.*
☐ *fstream.*
☐ *ifstream.*

22.

Section 17.12 Input/Output of Objects

17.12 Q1: Select the false statement. When objects are saved to a file:

- ☐ *Only data is stored about the object, not type information.*
- ☐ *Different types of objects cannot be written to the same file.*
- ☐ *They can be read from the file at a later time.*
- ☐ *It is often necessary to output the object's type as well.*

[Clear Answers / Start Over](#)[Submit Answers for Grading](#)

Answer choices in this exercise appear in a different order each time the page is loaded.



*Copyright © 1995 - 2010 [Pearson Education](#). All rights reserved.
[Legal Notice](#) | [Privacy Policy](#) | [Permissions](#)*