Chapter 1 - Bags

True/False (10)

1. An abstract data type is not necessarily a collection.

Answer: true

2. A collection is not necessarily an abstract data type.

Answer: false

3. Entries of a bag can belong to classes related by inheritance.

Answer: true

4. Entries of a bag may belong to different classes because items are unordered.

Answer: false

5. All entries of a bag must have the same data type or a subtype of that data type.

Answer: true

6. By declaring a data type to be a BagInterface, we can replace the bag class with another class that implements the BagInterface.

Answer: true

7. A method can change the state of an object passed to it as an argument.

Answer: true

8. A set is a special kind of bag.

Answer: true

9. You cannot determine the exact number of times a particular item appears in a bag because it is unordered.

Answer: false

10. Code written with respect to an interface makes it difficult to replace one implementation of a bag with another.

Answer: false

Short Answer (5)

1. Explain the rationale for returning a Boolean value from the *add* bag operation.

Answer: A client should know what happened. A Boolean value indicates true for success or false for a failed operation.

2. Explain why writing a test program before implementing a class is a good idea.

Answer: It helps you understand the specifications before committing to your design and implementation.

3. Explain why the *set* ADT does not need a *getFrequencyOf* operation.

Answer: Because sets do not contain duplicate items, only a 0 for absence or 1 for presence of an item would be returned. Using a *contains* method is a better design.

4. What is the design rationale for providing a *toArray* method instead of a *displayBag* method?

Answer: The user/client is free to display any or all of the items in any way desired.

5. Explain why null is a potential return value for the remove operation.

Answer: It is a value that cannot be in the bag and therefore signals a problem if, for example, a client tries to remove an entry from an empty bag.

Multiple Choice (30) WARNING: CORRECT ANSWERS ARE IN THE SAME POSITION AND TAGGED WITH **. YOU SHOULD RANDOMIZE THE LOCATION OF THE CORRECT ANSWERS IN YOUR EXAM.

1. A language-independent specification for a group of values and operations on those values is called a/an:
    a. abstract data type **
    b. data structure
    c. collection
    d. primitive

2. An implementation of an ADT with a programming language is called a/an:
    a. data structure **
    b. abstract data type
    c. collection
    d. primitive

3. An object that groups other objects and provides services to its clients is called a/an:
    a. collection **

b. abstract data type
c. data structure
d. primitive

4. Which of the following is not true with regard to bags?
    a. objects are in a specific order **
    b. can contain duplicate items
    c. is an abstract data type
    d. is a kind of collection in Java

5. Which behavior is not represented in a bag?
    a. reorder the bag **
    b. report the number of items in the bag
    c. report if the bag is empty
    d. add an item to the bag

6. Which method returns a count of the current number of items in a bag?
    a. getCurrentSize() **
    b. getSize()
    c. size()
    d. currentSize()

7. Why would the add method return false?
    a. when the addition of a new item was not successful **
    b. when there was a duplicate of the entry already in the bag
    c. when there was not a duplicate of the entry already in the bag
    d. when addition of a new item was successful

8. Which method removes all entries of a bag?
    a. clear() **
    b. remove()
    c. delete()
    d. empty()

9. Which method removes one unspecified entry from a bag if possible?
    a. remove() **
    b. clear ()
    c. delete()
    d. empty()

10. Which method removes one occurrence of a particular entry from a bag if possible?
    a. remove(anItem) **

b. clear (anItem)
c. delete(anItem)
d. empty(anItem)

11. Which behaviors do not change the contents of a bag?
    a. add()
    b. clear()
    c. remove()
    d. none of the above **

12. Which behaviors change the contents of a bag?
    a. clear()
    b. remove()
    c. add()
    d. all of the above **

13. Which method can be used to retrieve all of the elements of a bag?
    a. toArray() **
    b. printBag()
    c. displayBag()
    d. getAllItems()

14. When using abstraction as a design principle you should focus on:
    a. what you want to do with the data **
    b. how the data is stored
    c. how the data is manipulated
    d. all of the above

15. Which of the following is a special kind of bag that does not allow duplicate entries?
    a. set **
    b. collection
    c. extended bag
    d. super bag

16. Which of the following are properties of a set?
    a. duplicates are permitted **
    b. elements are stored in order
    c. elements may not be removed
    d. none of the above

17. When adding an item to a bag, which of the following statements are true?
    a. You cannot specify the position of the item in the bag. **

b. You can specify the item will be placed as the first in the bag.
c. You can specify the item will be placed as the last in the bag.
d. None of the above.

18. To accommodate entries of any class, the bag methods use _____ .
    a. a generic type **
    b. an inherited type
    c. a sub class
    d. all of the above

19. Which behavior(s) change the contents of a bag?
    a. remove() **
    b. getCurrentSize()
    c. isEmpty()
    d. getFrequencyOf()

20. Which behavior(s) change the contents of a bag?
    a. add() **
    b. getFrequencyOf()
    c. contains()
    d. toArray()

21. Which behaviors leave the contents of a bag unaltered?
    a. contains() **
    b. remove()
    c. clear()
    d. add()

22. Which behavior(s) leave the contents of a bag unaltered?
    a. toArray()
    b. getFrequencyOf()
    c. isEmpty()
    d. all of the above **

23. Which of the following are good reasons to write Java statements that use your bag ADT before the implementation is done?
    a. it helps confirm the design
    b. it helps check the suitability of the specification
    c. it helps check your understanding of the specification
    d. all of the above **

24. Which of the following would be suitable items to store in a bag?

a. marbles **
b. coins **
c. student roster
d. all of the above

25. You wish to keep track of the distribution of words in a document, for example, how many times the word “apple” appears. You store each word in the document in a bag. Why is a bag a potentially good solution?
    a. the order of the words is unimportant
    b. the words in the document are not unique
    c. getFrequencyOf() is a useful behavior for determining the word distribution
    d. all of the above **

26. For which of the following would a bag implementation be suitable?
    a. a shopping cart
    b. a vending machine
    c. a postage stamp collection
    d. all of the above **

27. Which behavior is not specified in the bag ADT?
    a. sort() **
    b. contains()
    c. clear()
    d. remove()

28. Which bag behavior is not useful or relevant for a set?
    a. getFrequencyOf() **
    b. add()
    c. remove()
    d. contains()

29. Which bag behaviors would need to be modified when implementing a set?
    a. add() **
    b. clear()
    c. contains()
    d. toArray()

30. Sets that adhere to the standard interface in the Java Class library
    a. do not contain a pair of objects x and y such that x.equals(y) is true **
    b. are completely compatible with the bag interface
    c. allow duplicate entries
    d. order the objects contained in the set