Name:_____ Class:_____ Date:_____

Chapter 11: Sets and Dictionaries

Reference and copyright:

Author	Title	Publisher	Date	ISBN
Kenneth Lambert	Fundamentals of Python: Data Structures	Cengage Learning; 2nd Edition	October 11, 2018	ISBN-10 : 0357122755 ISBN-13 : 978-0357122754

1. Much like a list, a set contains items that are in a particular order.
   a. True
   b. False

2. With a set, the difference and subset operations are not symmetric.
   a. True
   b. False

3. The __*sub*__ method of the *set* class returns a set containing items in s1 that are not in s2.
   a. True
   b. False

4. If *S1* and *s2* are sets, the expression *s1.issubset(s2)* returns *True* if *s2* is a subset of *s1*.
   a. True
   b. False

5. A set is similar to a bag, but it contains unique data items and additional methods.
   a. True
   b. False

6. Python supports multiple inheritance, so a class can have more than one parent class.
   a. True
   b. False

7. The *AbstractSet* class is a subclass of *AbstractCollection* because *AbstractSet* introduces new instance variables for data.
   a. True
   b. False

Chapter 11: Sets and Dictionaries

8. The *AbstractSet* class is a subclass of *AbstractBag*.
   a. True
   b. False

9. In Python's *dict* type, values are inserted or replaced at a given location using the index operator *{}*.
   a. True
   b. False

10. The dictionary constructor has two optional collection arguments: a collection of keys and a collection of corresponding values.
   a. True
   b. False

11. The entries in a dictionary consist of a key, a value, and an index.
   a. True
   b. False

12. In the *Entry* class for a dictionary, comparisons are done using the *value* item in each entry.
   a. True
   b. False

13. Array-based implementations of sets and dictionaries do not perform well.
   a. True
   b. False

14. A hashing function acts on a given key by returning its absolute position in an array.
   a. True
   b. False

15. If a hashing function runs in constant time, insertions, accesses, and removals of the associated keys are O(1).
   a. True
   b. False

16. Two keys that hash to the same index is called a collision.

Chapter 11: Sets and Dictionaries

a. True
b. False

17. To reduce the probability of collisions with hashes, you can decrease the array length.
a. True
b. False

18. As the density, or number of keys relative to the length of an array decreases, so does the probability of hashing collisions.
a. True
b. False

19. A small load factor and an array length that is a prime number increases the chances for a hashing collision.
a. True
b. False

20. Two strings that are anagrams will return a unique integer value when the sum of the ASCII values is calculated.
a. True
b. False

21. The standard Python hash function always returns a unique positive integer.
a. True
b. False

22. In the hashing implementation of a set, *self.index* refers to the index of the chain in which the node was just located, or is -1 otherwise.
a. True
b. False

23. In the hashing implementation of a set, the *Bag* class is used to represent an item and a pointer to the next item in a chain.
a. True
b. False

24. In the hashing implementation of a dictionary, the data field of each node in a chain contains

Chapter 11: Sets and Dictionaries

an *Entry* object.

a. True
b. False

25. The data in sets and dictionaries are ordered by position by default.

a. True
b. False

26. Which of the following is true about sets?

a. the items in a set are arranged in order
b. the difference and subset operations on a set are symmetric
c. there are no duplicate items in a set
d. there is no standard *set* class in Python

27. Which of the following is a subset of Set A if Set A is {19 4 26 8}?

a. None of the choices are subsets of Set A
b. {19 4 26 8 0}
c. {4 8 19 26 44}
d. {}

28. For a given set *s*, which method returns *True* if *item* is in *s*, or *False* otherwise.

a. s.__contains__(item)
b. s.__iter__(item)
c. s = set()
d. S1.__sub__(s2)

29. What is the value of set *S* after the following operations?

```
S = set([3, 9, 6])
S.add(6)
S.add(4)
S.remove(6)
```

a. {3 9 6 4}
b. {3 9 4}
c. {3 9 6}
d. {3 4 6 9}

Name:_____ Class:_____ Date:_____

Chapter 11: Sets and Dictionaries

30. What strategy for implementing sets attempts to approximate random access into an array for insertions, removals, and searches?

a. indexing
b. linking
c. hashing
d. keying

31. The simplest implementations of sets are subclasses of which other class?

a. bags
b. queues
c. stacks
d. lists

32. Which method is specific to a set compared to a bag?

a. remove
b. __str__
c. add
d. __sub__

33. In the code for the *__sub__ method for the AbstractSet class, what is the missing code?*

```
def __sub__(self, other):
    difference = type(self)()
    for item in self:
        if not item in other:
            <missing code>
    return difference
```

a. difference.remove(item)
b. intersection.add(item)
c. difference.add(item)
d. return(item)

34. Which method in the interface for a dictionary collection returns an iterator on the key/value pairs in the dictionary?

a. keys()
b. entries()
c. pairs()
d. values()

Chapter 11: Sets and Dictionaries

35. In the code for the *__init__* method in the *Entry* class or a dictionary, what is the missing code?

```
def __init__(self, key, value):
    <missing code>
    self.value = value
```

a. key = self.key
b. self.key = key
c. value = self.key
d. self.key = value

36. In the implementation of the AbstractDict class, which four methods have the same implementation for all dictionaries?

a. keys, values, __add__, __eq__
b. entries, values, __init__, __str__
c. keys, values, entries, get
d. get, __add__, keys, __init__

37. In the code for the *__iter__* method for the *ArrayDict* class, what is the missing code?

```
def __iter__(self):
    cursor = 0
    while cursor < len(self):
        yield self.items[cursor].key
        <missing code>
```

a. return(self.items[cursor])
b. cursor -= 1
c. return(self.items[key])
d. cursor += 1

38. What is the performance value of the array-based implementations of sets and dictionaries?

a. $O(n^2)$
b. $O(n)$
c. On
d. O(1)

39. For a key value of 93 and a hashing function of *key % 46*, what is the index into the array?

Chapter 11: Sets and Dictionaries

a. 1
b. 2
c. 46
d. 47

40. For key values of 84 and 108 and a hashing function of *key % 12*, what is the result?
a. indexes of 7 and 9
b. indexes of 0 and 1
c. a collision
d. indexes of 12 and 14

41. What happens when two keys map to the same index after a hashing function has been applied?
a. the array is lengthened
b. a collision occurs
c. the second item is mapped to the next cell
d. the hash is reapplied with a random value

42. In the code for the *keysToIndexes* function, what is the missing code?

```
def keysToIndexes(keys, n):
    return <missing code>
```

a. list(map(lambda key: key % n, keys))
b. list(map(lambda key: keys, key % n))
c. map(list(lambda key: key % n, keys))
d. map(list(lambda key % n: keys, key))

43. Referring to the *keysToIndexes* function, what is the result of the following statement?
`keysToIndexes([39, 18, 4, 51, 6, 28], 9)`
a. [4, 1, 5, 7, 6, 2]
b. [3, 0, 4, 6, 6, 1]
c. [2, 0, 3, 5, 5, 0]
d. [8, 3, 6, 0, 1, 4]

44. Referring to the *keysToIndexes* function, what is the result of the following statement?
`keysToIndexes([39, 18, 4, 51, 6, 28], 17)`
a. [6, 2, 5, 1, 7, 12]

Chapter 11: Sets and Dictionaries

b. [7, 3, 6, 2, 8, 13]
c. [4, 2, 3, 1, 5, 10]
d. [5, 1, 4, 0, 6, 11]

45. Which statement is true when considering a hashing strategy and the density of the keys/array length relationship?
a. as the density decreases, the probability of collisions decreases
b. as the density increases, the probability of collisions decreases
c. as the density decreases, the probability of collisions increases
d. as the density increases, the probability of collisions stays the same

46. Which of the following is the best array length to reduce the probability of collisions given the set [8, 6, 18, 9, 14, 23]?
a. 9
b. 8
c. 11
d. 12

47. When considering an insertion into a set using a hash function and linear probing, which of the following is defined as the position where the item should go if the has function works perfectly?
a. absolute index
b. probe index
c. zero index
d. home index

48. In which collision-avoidance hashing strategy are the items stored in an array of linked lists in which each item's key locates the bucket where the item is to be inserted?
a. clustering
b. chaining
c. quadratic probing
d. linear probing

49. In the algorithm for the *__contains* method of the hashing implementation of sets, what is the first step in the algorithm?
a. set foundNode to table[index]
b. set priorNode to foundNode
c. set index to the home index of the item

Chapter 11: Sets and Dictionaries

d. set foundNode to foundNode.next

50. What strategy does the hashing implementation of a dictionary use?
    a. a binary search tree
    b. quadratic probing
    c. linear probing/bucket
    d. bucket/chaining