CS303E Week 9 Worksheet: Lists

Name: _____ EID: _____

Read the questions carefully, and answer each question in the space provided. Use scratch paper to do your work and then copy your answers neatly and legibly onto the test paper. Only answers recorded on the test paper will be graded.

1. (10 points: 1 point each) The following are true/false questions. **Write either T or F in the boxes at the bottom of page 1.** If there's any counterexample, it's false.

   (a) If `L1` and `L2` are lists of different lengths, `L1 == L2` will raise an error.

   (b) It is possible to directly convert a string to a list of its characters in Python.

   (c) The `append()` method adds an element to the beginning of a list in Python.

   (d) The expression `[x for x in range(10) if not x % 2]` creates a list containing only positive even integers.

   (e) Lists cannot be negatively indexed.

   (f) The `remove()` method deletes an element at a specified index from a list.

   (g) Lists can only contain items of the same data type.

   (h) The `+` operator in Python can be used to concatenate two lists.

   (i) Lists have a fixed maximum size, and once this size is reached, no more elements can be added to the list.

   (j) The `pop()` method in Python is used to remove and return the last element of a list.

a	b	c	d	e	f	g	h	i	j

Page total: _____/11

Questions 2-9 are multiple choice. Each counts 2 points. **Write the letter of the BEST answer in the box on page 3. Please write your answer in UPPERCASE. Each problem has a single answer.**

2. What is the primary difference between the list methods `.append()` and `.extend()`?

   A. `.append()` is used to add a single element to the end of the list, while `.extend()` is used to add multiple elements to the end of the list.

   B. `.extend()` is used to add a single element to the end of the list, while `.append()` is used to add multiple elements to the end of the list.

   C. `.append()` is used for merging two lists, while `.extend()` is used to add a single element to a list.

   D. Both methods are identical; they can be used interchangeably to achieve the same result.

3. Which of the following list comprehensions is not valid in Python?

   A. `[x for x in range(0)]`

   B. `[x**2 for x in range(5) if x % 2 == 0]`

   C. `[len(item) for item in ["my", "anaconda", "don't"]]`

   D. `[x if x > 0 else -1 for x in range(5)]`

   E. `[x for x in "mercury is visible"]`

   F. all of the above are valid list comprehensions

4. Assuming `list1` is a list, which of the following declarations will not create a new list object with the same items as `list1`?

   A. `list2 = list1[0:len(list1)]`

   B. `list3 = list1`

   C. `list4 = [goodies for goodies in list1]`

   D. `list5 = list(list1)`

   E. all of the above create lists distinct from `list1` in memory

5. Assuming `myLst = ["Basil", "Kookie", "KitKat", "Grey"]`, what does the expression `myLst[:-2]` evaluate to?

   A. `["Basil, "Kookie"]`

   B. `["Basil, "Kookie", "KitKat"]`

   C. `["Kookie, "KitKat", "Grey"]`

   D. it's an error

6. In Python, when comparing two lists using the `>` or `<` operators, how is the comparison evaluated?

   A. Compares the lengths of the lists.

   B. Compares the elements of the lists lexicographically, after converting them to strings.

   C. Compares the elements at corresponding indices in the lists.

   D. List comparison using `>` or `<` is not allowed in Python.

7. What is the difference between the `.find()` and `.index()` methods in Python?

   A. `.find()` works only on strings and returns -1 if the specified element is not found, while `.index()` works on strings and lists but raises an error if the specified element is not found.

   B. `.find()` only works on lists, and `.index()` only works on strings, but both return -1 if the specified element is not found.

   C. `.find()` and `.index()` are interchangeable methods, and there is no significant difference between them.

   D. `.find()` works on strings and lists, while `.index()` works only on strings, but both raise an error if the specified element is not found.

8. Which of the following statements accurately describes the uses of function `list`()?

   A. `list()` is used to create an empty list.

   B. `list()` can convert a string into a list, where each character becomes an element

   C. `list()` can convert other iterable objects like a range (from `range()`) into a list.

   D. all of the above are true

9. List items cannot be removed from a list after it's been created?

   A. False.

   B. True because lists are immutable.

   C. True because there are no list methods that do that.

   D. Lists don't contain elements.

2	3	4	5	6	7	8	9

Page total: _____/16

The following questions (2 points each) ask you to trace the behavior of some Python code and identify the output of that code. For each question, write the output for the code segment in the box provided. Don't worry about line breaks. If there's an error, say so, but you don't need to specify the type of error.

10.
```
school = ["nemo", "marlin", "dory"]
school2 = school
school2.append("gill")
print(school)
```

11.
```
doubleTrouble = ["garfield", "odie", "tom", "jerry"]
doubleTrouble = doubleTrouble.sort()
print(doubleTrouble)
```

12.
```
toontown = []
clubhouse = ["mickey", "minnie", "pluto", "donald"]
for i in range(len(clubhouse)):
    toontown += clubhouse[i]
print(toontown)
```

13.
```
pink = ["courage", "cheshireCat", "pinkPanther"]
myFavoriteColor = ["courage", "cheshireCat", "pinkPanther"]
print(myFavoriteColor is pink and pink == myFavoriteColor)
```

14.
```
answer = ["deepThought", "life", "universe", "everything", 42]
answer.sort()
print(answer)
```

Page total: _____/10

15.
```
myNums = [1, 2, 3, 4, 5]
print([num ** 2 for num in myNums])
```

16.
```
super = ["peach"]
shrooms = ["toad", "mario", "yoshi"]
for minion in range(len(super)):
    super.append(shrooms[minion])
print(super)
```

17.
```
lst1 = ["Squirtle", "Snorlax"]
lst2 = ["Bulbasaur", "Charmander"]
print(min(max(lst1, lst2)))
```

18.
```
def iChooseYou(pc):
    pc.append("Pikachu")
    pc.sort()

pokemon = ["Onix", "Psyduck"]
iChooseYou(pokemon)
print(pokemon)
```

Page total: _____/8