

Beijing National Day School  
Department of Mathematics

AP Computer Science Principles

Test 2: Lists and Dictionaries

English Name: \_\_\_\_\_

Pinyin Name: \_\_\_\_\_

Mr. Alwin Tareen, Fall 2018

**Exam Record**

Part1 \_\_\_\_\_ / 20 pts

Part2 \_\_\_\_\_ / 15 pts

Part3 \_\_\_\_\_ / 12 pts

Total: \_\_\_\_\_ / 47 pts

Grade: \_\_\_\_\_

**Part I: Multiple Choice** (20 points)

- Determine the answer to each of the following questions, using the available space for any necessary scratchwork.
- Decide which is the best of the choices given, and select the correct answer by placing an “X” in the corresponding box.

(1<sup>pt</sup>) 1. For the following list, how would you print out "Sally"?

```
friends = ["Joseph", "Glenn", "Sally"]
```

☐ `print friends[3]`

☐ `print friends['Sally']`

☒ `print friends[2]`

☐ `print friends[2:1]`

1 pt

(1<sup>pt</sup>) 2. Which of the following Python statements would print out the length of a list stored in the variable `fruit`?

☐ `print length(fruit)`

☐ `print fruit.length()`

☒ `print len(fruit)`

☐ `print strlen(fruit)`

1 pt

(1<sup>pt</sup>) 3. What type of data is produced when you call the `range()` function? For example, consider the statement: `nums = range(5)`

☐ A list of characters

☒ A list of integers

☐ A list of words

☐ A string

1 pt

(1<sup>pt</sup>) 4. What does the following Python code print out?

```
first = [1, 2, 3]
```

```
second = [4, 5, 6]
```

```
nums = first + second
```

```
print len(nums)
```

☐ [1, 2, 3]

☐ [1, 2, 3, 4, 5, 6]

☐ [4, 5, 6]

☒ 6

1 pt

(1<sup>pt</sup>) 5. Which of the following slicing operations will produce the list [12, 3]?

```
nums = [9, 41, 12, 3, 74, 15]
```

☐ `nums[1:3]`

☒ `nums[2:4]`

☐ `nums[2:2]`

☐ `nums[12:3]`

1 pt

5 pts

(1<sup>pt</sup>) 6. Which list method adds a new item to the end of an existing list?

- ☐ add()  
☒ append()  
☐ index()  
☐ push()

1 pt

(1<sup>pt</sup>) 7. What will the following Python code print out?

```
friends = ["Joseph", "Glenn", "Sally"]  
friends.sort()  
print friends[0]
```

- ☒ Glenn  
☐ Joseph  
☐ friends  
☐ Sally

1 pt

(1<sup>pt</sup>) 8. Which of the following Python functions deletes an element from a list?

- ☐ push()  
☒ pop()  
☐ invalidate()  
☐ split()

1 pt

(1<sup>pt</sup>) 9. Which of the following Python functions breaks a string into a list of words?

- ☒ split()  
☐ join()  
☐ remove()  
☐ extend()

1 pt

(1<sup>pt</sup>) 10. What task does the following Python code perform?

```
for num in range(1, 10, 2):  
    print num
```

- ☒ It prints all the ODD numbers in the range [1, 9]  
☐ It prints all numbers in the range [1, 9]  
☐ This code fails with a traceback.  
☐ It prints all the EVEN numbers in the range [1, 10]

1 pt

(1<sup>pt</sup>) 11. What is the purpose of the second parameter of the `get()` method for Python dictionaries?

- ☐ It signifies a key which must be placed in the dictionary.  
☐ It specifies a unique key that the programmer wishes to retrieve.  
☐ It indicates the particular value that the programmer wants to retrieve.  
☒ To provide a default value if the key (from the first parameter of the `get()` method) does not exist in the dictionary.

1 pt

6 pts

- (1<sup>pt</sup>) **12.** How are Python dictionaries different from Python lists?
- ☐ Python lists can store multiple values, whereas Python dictionaries store a single value.
  - ☐ Python lists can store strings, while Python dictionaries can only store words.
  - ☒ Python lists are indexed using integers, whereas Python dictionaries are indexed with any immutable data type.
  - ☐ Python dictionaries are mutable, while Python lists are immutable.

1 pt

- (1<sup>pt</sup>) **13.** What would be the output produced by the following Python code?

```
fruit = {"banana":5, "pear":3, "orange":8}
result = fruit["kiwi"]
print(result)
```

- ☐ 0
- ☒ This program would fail with a traceback.
- ☐ kiwi
- ☐ -1

1 pt

- (1<sup>pt</sup>) **14.** What would be the output produced by the following Python code?

```
fruit = {"banana":5, "pear":3, "orange":8}
result = fruit.get("kiwi", 0)
print(result)
```

- ☒ 0
- ☐ This program would fail with a traceback.
- ☐ kiwi
- ☐ -1

1 pt

- (1<sup>pt</sup>) **15.** Consider the following Python code, in which we loop through a dictionary. What are the items that the for loop iterates through?

```
fruit = {"banana":5, "pear":3, "orange":8}
for item in fruit:
    print(item)
```

- ☒ The keys in the dictionary.
- ☐ The integers in `range(0, len(fruit))`
- ☐ The values in the dictionary.
- ☐ All of the mutable data types in the dictionary.

1 pt

- (1<sup>pt</sup>) **16.** Which of the following Python methods would you use to create a separate and distinct copy of a dictionary?

- ☐ `double()`
- ☐ `duplicate()`
- ☒ `copy()`
- ☐ `clone()`

1 pt

5 pts

(1<sup>pt</sup>) 17. Consider the following Python dictionary:

```
fruit = {"banana":5, "pear":3, "orange":8}
```

Which of the following statements would correctly remove the key-value pair "orange":8 from this dictionary?

- ☐ remove.fruit["orange"]
- ☐ eliminate("orange":8)
- ☐ del fruit[8]
- ☒ del fruit["orange"]

1 pt
------

(1<sup>pt</sup>) 18. What would be the output produced by the following Python code?

```
drinks = {"coffee":87, "tea":23, "juice":49}
result = drinks.values()
print(result)
```

- ☐ (78, 32, 94)
- ☐ [("coffee",87), ("tea",23), ("juice",49)]
- ☐ ["coffee", "tea", "juice"]
- ☒ [87, 23, 49]

1 pt
------

(1<sup>pt</sup>) 19. Consider the following Python dictionary:

```
fruit = {"banana":5, "pear":3, "orange":8}
```

Which of the following Python statements would correctly subtract 2 from the value corresponding to the key "orange"?

- ☐ fruit["orange"].reduce(2)
- ☐ orange subtraction 2
- ☐ fruit.orange.minus.2
- ☒ fruit["orange"] -= 2

1 pt
------

(1<sup>pt</sup>) 20. Consider the following Python dictionary:

```
cheese = {"swiss":3, "cheddar":7, "gouda":6}
```

Which of the following Python statements indicates whether "swiss" appears as a key in the dictionary cheese?

- ☐ cheese.excludevalue("cheddar", "gouda")
- ☒ "swiss" in cheese
- ☐ cheese.containsvalue("swiss")
- ☐ cheese --> "swiss"

1 pt
------

4 pts
-------

**Part II: Short Answer** (15 points)

- Solve each of the following short answer questions. Write your solution in the corresponding box labelled, “Answer:”.

(1 <sup>pt</sup> )	1. What is the output of the following code: <code>cheeses = ["Cheddar", "Edam", "Gouda"] print(cheeses[0])</code> Answer: Cheddar	<div>1 pt</div>
(1 <sup>pt</sup> )	2. What is the output of the following code: <code>print([0] * 4)</code> Answer: [0, 0, 0, 0]	<div>1 pt</div>
(1 <sup>pt</sup> )	3. What is the output of the following code: <code>snacks = ["pizza", "burger"] snacks.append("fries") print(snacks)</code> Answer: ["pizza", "burger", "fries"]	<div>1 pt</div>
(1 <sup>pt</sup> )	4. What is the output of the following code: <code>drinks = ["tea", "soda", "cola", "juice"] drinks.sort() print(drinks)</code> Answer: ["cola", "juice", "soda", "tea"]	<div>1 pt</div>
(1 <sup>pt</sup> )	5. What is the output of the following code: <code>dinner = ["salad", "bread", "steak", "potato"] del dinner[1] print(dinner)</code> Answer: ["salad", "steak", "potato"]	<div>1 pt</div>
(1 <sup>pt</sup> )	6. What is the output of the following code: <code>nums = [3, 41, 12, 9, 74, 15] print(max(nums))</code> Answer: 74	<div>1 pt</div>
(1 <sup>pt</sup> )	7. What is the output of the following code: <code>food = {"pizza":3} food["fries"] = 10 print(food)</code> Answer: {"pizza":3, "fries":10}	<div>1 pt</div>
(1 <sup>pt</sup> )	8. What is the output of the following code: <code>treasure = {"gold":50, "silver":100} print("gold" in treasure)</code> Answer: True	<div>1 pt</div>

- (1<sup>pt</sup>) 9. What is the output of the following code:

```
inventory = {  
    "pocket": "lint",  
    "canteen": "water",  
    "pouch": "flint",  
    "backpack": ["shovel", "bedroll", "rope"]  
}  
print(inventory["backpack"])
```

Answer: ["shovel", "bedroll", "rope"]

1 pt

- (1<sup>pt</sup>) 10. What is the output of the following code:

```
fortune = {"gold": 500}  
fortune["gold"] += 50  
print(fortune)
```

Answer: {"gold": 550}

1 pt

- (1<sup>pt</sup>) 11. What is the output of the following code:

```
inventory = {  
    "gold": 500,  
    "backpack": ["xylophone", "dagger", "bedroll"]  
}  
inventory["backpack"].sort()  
print(inventory["backpack"])
```

Answer: ["bedroll", "dagger", "xylophone"]

1 pt

- (1<sup>pt</sup>) 12. What is the output of the following code:

```
grocery = {"kiwi": 5, "grape": 12}  
del grocery["kiwi"]  
print(grocery)
```

Answer: {"grape": 12}

1 pt

- (1<sup>pt</sup>) 13. Consider the following dictionary:

```
salad = {"caesar": 1, "garden": 2}
```

Write an assignment statement that modifies this dictionary to become the following:

```
salad = {"caesar": 1, "vegetable": 3, "garden": 2}
```

Answer: salad["vegetable": 3]

1 pt

- (1<sup>pt</sup>) 14. What is the output of the following code:

```
singer = {"justin": "bieber", "taylor": "swift", "ed": "sheeran"}  
print(singer.get("swift", "guitar"))
```

Answer: guitar

1 pt

- (1<sup>pt</sup>) 15. What is the output of the following code:

```
sports = {"tennis": 43, "football": 78, "badminton": 52}  
result = list(sports.keys())  
print(result)
```

Answer: ["tennis", "football", "badminton"]

1 pt

7 pts

**Part III: Python Programming** (12 points)

- Show all of your work. Remember that program segments are to be written in the Python programming language.

(3<sup>pts</sup>) 1. Consider the following partially completed function called `inventory()` that manages the total quantity of products in a grocery store. It contains two dictionaries:

- `prices` gives the cost of each product.
- `stock` indicates the quantity of each product in the store.

Complete the implementation of the function `inventory()` so that it returns the total value of all the products in the store. The function should return a `float`.

- If the following statements are executed, the output of your program should be: 129.5

```
result = inventory()
print(result)
```

```
def inventory():
```

```
    """
```

```
    returns a float, the total value of all the products.
```

```
    """
```

```
    prices = {"banana":4, "apple":2, "orange":1.5, "pear":3}
```

```
    stock = {"banana":6, "apple":7, "orange":31, "pear":15}
```

```
    # YOUR CODE HERE
```

3 pts

3 pts



## (3pts) 2. DNA to RNA Transcription

3 pts

- A strand of DNA is composed of a long sequence of molecules called nucleotides or bases. Only four distinct bases are used: *adenine*, *cytosine*, *guanine*, and *thymine*, which are respectively abbreviated as A, C, G, and T.
- An organism uses DNA as a model when constructing a complementary structure called RNA. The process of creating RNA from DNA is known as *transcription*. The RNA is then used to make proteins.
- RNA also consists of four nucleotides, three of them being A, C, and G, and a fourth one *uracil*, which is abbreviated as U.
- Transcription creates an RNA sequence by matching a complementary base to each original base in the DNA, using the following substitutions:

DNA	→	RNA
A	→	U
C	→	G
G	→	C
T	→	A

Consider the following partially completed function called `transcription(dna)`. It contains a single dictionary called `mapping` which gives each DNA base, and its corresponding RNA substitution:

```
mapping = {"A": "U", "C": "G", "G": "C", "T": "A"}
```

Complete the implementation of the function `transcription(dna)` so that it returns the transcribed RNA which corresponds to the DNA sequence being read in by the parameter `dna`. The function should return a string.

- If the following statements are executed, the output of your program should be:  
UCCGAUGCA

```
result = transcription("AGGCTACGT")
print(result)
```

Write your answer on the next page.

3 pts

```
def transcription(dna):  
    """
```

```
    dna, a string representing a DNA sequence  
    returns a string, the transcribed RNA sequence  
    """
```

```
    mapping = {"A": "U", "C": "G", "G": "C", "T": "A" }
```

(6pts) 3. Numbers in Mandarin follow 3 simple rules:

- There are words for each of the digits from 0 to 10.
- For numbers 11 to 19, then number is pronounced as “ten digit,” so for example, 16 would be pronounced(using Mandarin) as “ten six.”
- For numbers between 20 and 99, the number is pronounced as “digit ten digit,” so for example, 37 would be pronounced(using Mandarin) as “three ten seven.” If the digit is a zero, it is not included.

Consider the following partially completed function called `convert_to_mandarin(eng)`. It contains a single dictionary called `nums` which expresses the Mandarin translation for each of the numerical digits from 0 to 10:

```
nums = {"0": "ling", "1": "yi", "2": "er", "3": "san", "4": "si",  
        "5": "wu", "6": "liu", "7": "qi", "8": "ba", "9": "jiu", "10": "shi"}
```

Complete the implementation of the function `convert_to_mandarin(eng)` so that it converts an English number(between 0 and 99), **given as a string**, into the equivalent Mandarin. The following Python statements demonstrate the expected outputs from this function:

- `print(convert_to_mandarin("36"))` will return `san shi liu`
- `print(convert_to_mandarin("20"))` will return `er shi`
- `print(convert_to_mandarin("16"))` will return `shi liu`

Write your answer on the next page.

6 pts
-------

6 pts
-------

```
def convert_to_mandarin(eng):  
    """  
    eng, a string representing an English number 0 to 99  
    returns the string mandarin representation of eng  
    """  
    nums = {"0": "ling", "1": "yi", "2": "er", "3": "san", "4": "si",  
            "5": "wu", "6": "liu", "7": "qi", "8": "ba", "9": "jiu", "10": "shi"}  
    # YOUR CODE HERE
```