

Beijing National Day School  
Department of Mathematics & Computer Science

# AP Computer Science Principles

Semester 1 Exam

Location: Library, 6th Floor, Aspiration Building

Date: Monday, January 15th, 2018

Start Time: 8:00AM

End Time: 10:00AM

**NO CALCULATORS PERMITTED**

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

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

Mr. Alwin Tareen, January 2018

## Exam Record

Multiple Choice \_\_\_\_\_ / 30 pts

Short Answer \_\_\_\_\_ / 20 pts

Free Response \_\_\_\_\_ / 20 pts

Total: \_\_\_\_\_ / 70 pts

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

**Section I: Multiple Choice (30 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. In Python, the float data type is used to store:

booleans

decimal numbers

strings

integers

1 pt

(1<sup>pt</sup>) 2. What are the only values that are permissible in Python's boolean data type?

Yes, No

On, Off

Right, Wrong

True, False

1 pt

(1<sup>pt</sup>) 3. Which Python keyword indicates the start of a function definition?

sweet

def

continue

return

1 pt

(1<sup>pt</sup>) 4. Consider the following function definition:

```
def circlearea(radius):
```

In this context, what is the formal name for the variable `radius`?

expression

logical deduction

parameter

condition

1 pt

(1<sup>pt</sup>) 5. What does the following Python program print out?

```
str1 = "Hello"  
str2 = "there"  
greet = str1 + str2  
print(greet)
```

Hello there

Hellothere

there

Hello

1 pt

- (1<sup>pt</sup>) **6.** How would you use the index operator to print out the letter "q" from the following string?  
`x = "From marquard@uct.ac.za"`
- `print(x[9])`  
 `print(x[8])`  
 `print(x[-1])`  
 `print(x[q])`
- (1<sup>pt</sup>) **7.** How would you use string slicing to print out "uct" from the following string?  
`x = "From marquard@uct.ac.za"`
- `print(x[14+17])`  
 `print(x[15:18])`  
 `print(x[14:17])`  
 `print(x[14:3])`
- (1<sup>pt</sup>) **8.** What is the iteration variable in the following Python code?  
`for letter in "banana":`  
 `print(letter)`
- `letter`  
 `print`  
 `in`  
 `"banana"`
- (1<sup>pt</sup>) **9.** How would you print out the following string in all upper case in Python?  
`greet = "Hello there"`
- `puts greet.ucase;`  
 `print(uc($greet))`  
 `print(greet.upper())`  
 `console.log(greet.toUpperCase());`
- (1<sup>pt</sup>) **10.** What does the following Python program print out?  
`data = "From stephen.marquard@uct.ac.za"`  
`pos = data.find(".")`  
`print(data[pos:pos+3])`
- `uct`  
 `mar`  
 `.ma`  
 `ste`
- (1<sup>pt</sup>) **11.** 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<sup>pt</sup>) **12.** Which of the following Python statements would print out the length of a list stored in the variable `fruit`? 

1 pt
- `print length(fruit)`
  - `print fruit.length()`
  - `print len(fruit)`
  - `print strlen(fruit)`
- (1<sup>pt</sup>) **13.** What type of data is produced when you call the `range()` function? For example, consider the statement: `nums = range(5)`

1 pt
- A list of characters
  - A list of integers
  - A list of words
  - A string
- (1<sup>pt</sup>) **14.** What does the following Python code print out? 

1 pt
- ```
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<sup>pt</sup>) **15.** Which of the following slicing operations will produce the list [12, 3]? 

|      |
|------|
|      |
| 1 pt |
- ```
nums = [9, 41, 12, 3, 74, 15]
```
- `nums[1:3]`
  - `nums[2:4]`
  - `nums[2:2]`
  - `nums[12:3]`
- (1<sup>pt</sup>) **16.** Which list method adds a new item to the end of an existing list? 

1 pt
- `add()`
  - `append()`
  - `index()`
  - `push()`
- (1<sup>pt</sup>) **17.** Which of the following Python functions deletes an element from a list? 

1 pt
- `push()`
  - `pop()`
  - `invalidate()`
  - `split()`

- (1<sup>pt</sup>) **18.** What will the following Python code print out?
- ```
friends = ['Joseph', 'Glenn', 'Sally']
friends.sort()
print friends[0]
```
- Glenn  
 Joseph  
 friends  
 Sally
- (1<sup>pt</sup>) **19.** 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<sup>pt</sup>) **20.** 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<sup>pt</sup>) **21.** 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<sup>pt</sup>) **22.** Which of the following Python functions breaks a string into a list of words?
- `split()`  
 `join()`  
 `remove()`  
 `extend()`
- (1<sup>pt</sup>) **23.** Which of the following Python methods would you use to create a separate and distinct copy of a dictionary?
- `double()`  
 `duplicate()`  
 `copy()`  
 `clone()`

- (1<sup>pt</sup>) **24.** 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<sup>pt</sup>) **25.** 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<sup>pt</sup>) **26.** 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<sup>pt</sup>) **27.** 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<sup>pt</sup>) **28.** 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<sup>pt</sup>) **29.** 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>) **30.** 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 |

**Section II: Short Answer** (20 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 Python code:
- ```
lunch = "cheeseburgers"
print(lunch[6:12])
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 2. What is the output of the following Python code:
- ```
greeting = "Hello, world!"
newgreeting = "J" + greeting[1:]
print(newgreeting)
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 3. What is the output of the following Python code:
- ```
print("cola" in "chocolate")
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 4. What is the output of the following Python code:
- ```
print("seed" in "banana")
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 5. What is the output of the following Python code:
- ```
fruit = "strawberry"
bigfruit = fruit.upper()
print(bigfruit)
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 6. What is the output of the following Python code:
- ```
vegetable = "cauliflower"
index = vegetable.find("u")
print(index)
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 7. What is the output of the following Python code:
- ```
line = "Please have a nice day"
print(line.startswith("Please"))
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 8. What is the output of the following Python code:
- ```
meal = "fresh pizza is the best pizza"
print(meal.replace("pizza", "salad"))
```
- Answer:
- 1 pt



- (1<sup>pt</sup>) 9. What is the output of the following code:
- ```
cheeses = ["Cheddar", "Edam", "Gouda"]
print(cheeses[0])
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 10. What is the output of the following code:
- ```
print([0] * 4)
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 11. What is the output of the following code:
- ```
snacks = ["pizza", "burger"]
snacks.append("fries")
print(snacks)
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 12. What is the output of the following code:
- ```
drinks = ["tea", "soda", "cola", "juice"]
drinks.sort()
print(drinks)
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 13. What is the output of the following code:
- ```
dinner = ["salad", "bread", "steak", "potato"]
del dinner[1]
print(dinner)
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 14. What is the output of the following code:
- ```
nums = [3, 41, 12, 9, 74, 15]
print(max(nums))
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 15. What is the output of the following code:
- ```
food = {"pizza":3}
food["fries"] = 10
print(food)
```
- Answer:
- 1 pt
- (1<sup>pt</sup>) 16. What is the output of the following code:
- ```
treasure = {"gold":50, "silver":100}
print("gold" in treasure)
```
- Answer:
- 1 pt

(1<sup>pt</sup>) 17. What is the output of the following code:

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

Answer:

|      |
|------|
|      |
| 1 pt |

(1<sup>pt</sup>) 18. What is the output of the following code:

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

Answer:

|      |
|------|
|      |
| 1 pt |

(1<sup>pt</sup>) 19. What is the output of the following code:

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

Answer:

|      |
|------|
|      |
| 1 pt |

(1<sup>pt</sup>) 20. What is the output of the following code:

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

Answer:

|      |
|------|
|      |
| 1 pt |

**Section III: Free Response** (20 points)

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

- (2<sup>pts</sup>) 1. Write a Python function that evaluates and returns the result of the following quadratic expression:

$$a \cdot x^2 + b \cdot x + c$$

2 pts

Your function should be called `evalquadratic(a, b, c, x)`, which takes in four parameters. The function should return a numeric value.

- If the following statements are executed:

```
result = evalquadratic(5, 8, 3, 2)
print(result)
```

- Then the output of your program should be:

```
39
```

- (2<sup>pts</sup>) 2. In Python you can use `+` to concatenate two lists. Write a Python function that takes in a list as a parameter, and returns a new list with the integer 6 concatenated to the end. Your function should be called `cat6(data)`, which takes in a single parameter, `data`. The function should return a list.

- If the following statements are executed:

```
result = cat6([3, 8, 4])
print(result)
```

- Then the output of your program should be:

```
[3, 8, 4, 6]
```

2 pts

- (2<sup>pts</sup>) **3.** Write a Python function that takes in a list as an parameter, and returns a new list consisting of alternating elements of the original list, starting with the element at index 0. Your function should be called `everyother(data)`, which takes in a single parameter, `data`. The function should return a list.

2 pts

- If the following statements are executed:

```
result = everyother(["tea", "water", "coffee", "soda", "milk"])
print(result)
```

- Then the output of your program should be:

```
["tea", "coffee", "milk"]
```

- (2<sup>pts</sup>) **4.** The mean(or average) of a list of `n` numbers is the sum of the numbers divided by `n`. Write a Python function that takes as a parameter a list of numbers(of any length) and returns the mean. Your function should be called `mean(data)`, which takes in a single parameter, `data`. The function should return a numeric value.

2 pts

- If the following statements are executed:

```
result = mean([2, 7, 3, 9, 13])
print(result)
```

- Then the output of your program should be:

```
6.8
```

- (2pts) 5. Write a Python function that takes two lists as parameters, and returns a list that gives the intersection of the two lists, that is, a list of elements that are common to both lists. *Note:* The ordering of your outputs does not matter, [3,2] is the same as [2,3]. Your function should be called `listintersection(alist, blist)`, which takes in two parameters, `alist` and `blist`. The function should return a list. *Note:* You may assume that there are no repeated occurrences of an element within each of the lists.

2 pts

- If the following statements are executed:

```
result = listintersection([1, 3, 6, 9], [10, 14, 3, 72, 9])
print(result)
```

- Then the output of your program should be:

```
[3, 9]
```

(5<sup>pts</sup>) **6.** Pig Latin is a type of slang language that is easy to learn and understand. An English word can be translated into Pig Latin by following these two simple rules:

- If the English word begins with a vowel, then the corresponding Pig Latin word is generated by appending the letters "hay" to the end of the word. For example, "orange" becomes "orangehay".
- If the English word begins with a consonant, then the corresponding Pig Latin word is generated by moving the first letter to the end of the word, then appending the letters "ay". For example, "peach" becomes "eachpay".

Write a Python function that takes in an English word as a parameter, and translates that word to Pig Latin. Your function should be called `piglatin(word)`, which takes in a single parameter, `word`. The function should return a string which is the Pig Latin translation of `word`.

```
def piglatin(word):
```

|       |
|-------|
|       |
| 5 pts |

(5pts) 7. The following are some facts regarding the double helix structure of DNA.

- The DNA molecule is made up of two **strands**, running in opposite directions.
- Each base bonds to a base in the opposite strand. *Adenine* always bonds with *Thymine*, and *Cytosine* always bonds with *Guanine*. The **complement** of a base is the opposite base to which it always bonds.
- The two strands are twisted together into a long spiral staircase structure called a **double helix**.
- If we know the order of bases on one strand, we can immediately deduce the sequence of bases in the complementary strand. These bases will run in the opposite order, to match the fact that the two strands of DNA run in opposite directions.
- In DNA strings, the bases *Adenine* and *Thymine* are complements of each other, and they are represented by the letters A and T respectively.
- Also, the bases *Cytosine* and *Guanine* are complements of each other, and they are represented by the letters C and G.

5 pts

Write a **Python** function that takes in a string of DNA base symbols as a parameter, and returns the complementary strand that corresponds to it. You will have to reverse the order of the symbols in the DNA string, then find the complement base of each of those symbols. Your function should be called `reversecomplement(dna)`, which takes in a single parameter, `dna`. The function should return a string.

- If the following statements are executed:

```
result = reversecomplement("AAAACCCGGT")
print(result)
```

- Then the output of your program should be:

```
ACCGGGTTTT
```

**Write your answer on the next page.**

**def** reversecomplement(dna):