

Beijing National Day School
Department of Mathematics

AP Computer Science Principles

Test 1: Python Syntax and Strings

English Name: _____

Pinyin Name: _____

Mr. Alwin Tareen, Fall 2018

Exam Record

Part1 _____ / 23 pts

Part2 _____ / 16 pts

Part3 _____ / 12 pts

Total: _____ / 51 pts

Grade: _____

Part I: Multiple Choice (23 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^{pt}) 1. Which of the following choices is a legal and legitimate Python variable name?
- | |
|------|
| |
| 1 pt |
- 2bad4you
 calvin&hobbes
 year2000
 #hammertime
- (1^{pt}) 2. You would like to set up a variable called `ounces` that has the value 16. What simple Python statement will accomplish this?
- | |
|------|
| |
| 1 pt |
- `ounces = 16`
 `16 = ounces`
 `def ounces(16):`
 `ounces(16)`
- (1^{pt}) 3. What does the following Python statement print out:
`print("123" + "abc")`
- | |
|------|
| |
| 1 pt |
- `"123" + "abc"`
 This is a syntax error because you cannot add strings.
 `123+abc`
 `123abc`
- (1^{pt}) 4. In Python, the float data type is used to store:
- | |
|------|
| |
| 1 pt |
- booleans
 decimal numbers
 strings
 integers
- (1^{pt}) 5. What is the result of the following Python statement:
`print(42%10)`
- | |
|------|
| |
| 1 pt |
- 1042
 420
 4
 2

- (1^{pt}) **6.** Which of the following choices is the correct assignment statement for a `string` data type?
- `greetings = [Hello]`
 - `greetings = @Hello@`
 - `greetings = "Hello"`
 - `greetings = #Hello#`
- (1^{pt}) **7.** What is the result of the following Python statement:
`print(17/4)`
- 4
 - 4.0
 - 4.3
 - 4.25
- (1^{pt}) **8.** What are the only values that are permissible in Python's boolean data type?
- Yes, No
 - On, Off
 - Right, Wrong
 - True, False
- (1^{pt}) **9.** Which of the following is a comment in Python?
- `/* This is a test */`
 - `// This is a test`
 - `# This is a test`
 - `% This is a test`
- (1^{pt}) **10.** Which of the following elements of a mathematical expression in Python is evaluated first?
- Multiplication `*`
 - Addition `+`
 - Parenthesis `()`
 - Subtraction `-`
- (1^{pt}) **11.** What will be the value of `x` when the following statement is executed: `x = int(98.6)`
- 99
 - 6
 - 98
 - 100
- (1^{pt}) **12.** What does the Python function `input()` do?
- Pause the program and read data from the user.
 - Take a screen shot from an area of the screen.
 - Read the memory of the running program.
 - Connect to the network and retrieve a web page.

7 pts

(1^{pt}) 13. Which Python keyword indicates the start of a function definition?

- sweet
- def
- continue
- return

1 pt

(1^{pt}) 14. 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^{pt}) 15. Which of the following is NOT a valid string method in Python?

- boldface()
- startswith()
- upper()
- strip()

1 pt

(1^{pt}) 16. 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^{pt}) 17. 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 pt

(1^{pt}) 18. 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 pt

6 pts

(1^{pt}) 19. What is the iteration variable in the following Python code?

```
for letter in "banana":  
    print(letter)
```

- letter
- print
- in
- "banana"

1 pt

(1^{pt}) 20. 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 pt

(1^{pt}) 21. 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 pt

(1^{pt}) 22. Consider the following string declaration:

```
grocery = "Mango"
```

Which of the following statements would cause an error(also known as a traceback)?

- dance = "T" + grocery[1:]
- person = grocery[:-2]
- several = grocery * 3
- grocery[0] = "T"

1 pt

(1^{pt}) 23. Consider the following Python code:

```
lunch = "pizza"  
dinner = lunch[:]
```

Note that the **start** and **stop** indexes are omitted from the square bracket notation. What is the technical term for the outcome of this kind of string slicing?

- concatenation
- immutable
- clone
- iteration

1 pt

5 pts

Part II: Short Answer (16 points)

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

- (1^{pt}) 1. What is the output of the following Python code:
`print(3 > 4 or (2 < 3 and 9 > 10))`
Answer: 1 pt
- (1^{pt}) 2. What is the output of the following Python code:
`lunch = "cheeseburgers"`
`print(lunch[6:12])`
Answer: 1 pt
- (1^{pt}) 3. What is the output of the following Python code:
`breakfast = "pineapple"`
`print(breakfast[:4])`
Answer: 1 pt
- (1^{pt}) 4. What is the output of the following Python code:
`flavor = "strawberry"`
`print(flavor[5:])`
Answer: 1 pt
- (1^{pt}) 5. What is the output of the following Python code:
`icecream = "vanilla"`
`print(icecream[:])`
Answer: 1 pt
- (1^{pt}) 6. What is the output of the following Python code:
`drink = "soda"`
`print(drink[:-1])`
Answer: 1 pt
- (1^{pt}) 7. What is the output of the following Python code:
`beverage = "water"`
`print(beverage * 3)`
Answer: 1 pt
- (1^{pt}) 8. What is the output of the following Python code:
`greetings = "Hello, world!"`
`newgreetings = "J" + greetings[1:]`
`print(newgreetings)`
Answer: 1 pt

 8 pts

- (1^{pt}) 9. What is the output of the following Python code:
- ```
print("cola" in "chocolate")
```
- Answer:
- (1<sup>pt</sup>) 10. What is the output of the following Python code:
- ```
print("seed" in "banana")
```
- Answer:
- (1^{pt}) 11. What is the output of the following Python code:
- ```
fruit = "kiwi"
bigfruit = fruit.upper()
print(bigfruit)
```
- Answer:
- (1<sup>pt</sup>) 12. What is the output of the following Python code:
- ```
citrus = "ORANGE"
smallcitrus = citrus.lower()
print(smallcitrus)
```
- Answer:
- (1^{pt}) 13. What is the output of the following Python code:
- ```
vegetable = "cauliflower"
index = vegetable.find("u")
print(index)
```
- Answer:
- (1<sup>pt</sup>) 14. What is the output of the following Python code:
- ```
line = "Please have a nice day"
print(line.startswith("Please"))
```
- Answer:
- (1^{pt}) 15. What is the output of the following Python code:
- ```
meal = "fresh pizza is the best pizza"
print(meal.replace("pizza", "salad"))
```
- Answer:
- (1<sup>pt</sup>) 16. What is the output of the following Python code:
- ```
def choose(x, y, z):
    if x:
        return y
    else:
        return z
print(choose(False, 2, 3))
```
- Answer:

Part III: Python Programming (12 points)

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

(2^{pts}) 1. Assume that `sample` is a string of lower case text characters. Write a Python function that counts the number of vowels that are contained in the string `sample`. Valid vowels are: "a", "e", "i", "o", "u". Your function should be called:

```
def countvowels(sample):
```

The function should return an integer which is the total quantity of vowels in the string.

- If the following statements are executed:

```
result = countvowels("azcbobobegghakl")
print(result)
```

Then the output of your program should be: 5

```
def countvowels(sample):
    // YOUR CODE HERE
```

2 pts

2 pts

- (2pts) 2. Write a Python function that takes in a string as a parameter, and generates a new string, which is made up of three copies of the last two characters of the original string. Your function should be called:

```
def extraend(word):
```

The function should return a string.

- If the following statements are executed:

```
result = extraend("hello")  
print(result)
```

Then the output of your program should be: lololo

```
def extraend(word):  
    // YOUR CODE HERE
```

2 pts

2 pts

- (4pts) **3.** In this question, you will write a Python function that performs the multiplication operation, but with a technique that the Ancient Egyptians used. The algorithm for Ancient Egyptian Multiplication can be expressed as follows. Assume that `grow` and `shrink` are the numbers to be multiplied together:

4 pts

- Create an integer variable called `product` to hold the solution.
- Check to see if `shrink` is an odd number.
- If `shrink` is odd, then add the number `grow` to the variable `product`.
- Multiply the number `grow` by 2.
- Divide the number `shrink` by 2 (*Note*: Use integer division).
- Continue until the number `shrink` becomes zero.

Write a Python function that takes in two integer values, `grow` and `shrink`, as parameters, and calculates their multiplicative product using the Ancient Egyptian Multiplication algorithm. Your function should be called:

```
def multiply(grow, shrink):
```

Note: This function returns an integer value.

- If the following statements are executed:

```
result = multiply(23, 58)
print(result)
```

Then the output of your program should be: 1334

Write your solution on the next page.

4 pts

```
def multiply(grow, shrink):  
    // YOUR CODE HERE
```

(4pts) 4. 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:

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

The function should return a string which is the Pig Latin translation of `word`.

- If the following statements are executed:

```
result = piglatin("orange")
print(result)
```

Then the output of your program should be: `orangehay`

- If the following statements are executed:

```
result = piglatin("peach")
print(result)
```

Then the output of your program should be: `eachpay`

Write your solution on the next page.

--

4 pts

--

4 pts

```
def piglatin(word):  
    // YOUR CODE HERE
```

0 pts

This page is left intentionally blank.