

*THE UNIVERSITY OF WESTERN ONTARIO
LONDON, CANADA*

Computer Science 1026a

MIDTERM EXAMINATION

OCTOBER 27, 2018

2 Hours: 3:00pm-5:00pm

SOLUTIONS

Question	Out of	Mark
<i>Section 1: Scantron</i>		
1. True/False	15	
2. Multiple Choice	40	
<i>Section 1 Total Marks</i>	<i>55</i>	
<i>Section 2: On Exam</i>		
3. Logic Errors	12	
4. A Little Code	33	
<i>Section 2 Total Marks</i>	<i>45</i>	
<i>Total</i>	<i>100</i>	

SECTION 1

Answer questions in PART 1 and PART 2 on the scantron provided. Any markings you make on the pages on the question booklet for PART 1 and PART 2 **WILL NOT** be graded

Question 1: True/False – 15 Marks (1 each)

For the following questions, please circle (or indicate as specified by the question) your answer directly on the exam sheet. Note that questions are each worth one point unless otherwise indicated.

use the name to reach the location.

- 1) A variable in Python has a name and a location (memory). True False
- 2) Python can have variables that hold integer values. True False
- 3) Boolean variables can only have a value of True or False. True False
- 4) Compilers translate source code into byte code. True False
- 5) The first position in a string in Python has the index 0. True False
- 6) The symbol '#' is used in Python to indicate a comment. True False
- 7) X3yZ is a valid variable name in Python. True False
- 8) 9.7E05 is a floating number in Python. True False
- 9) The keyword **def** is used to define a function in Python. True False
- 10) The assignment operator in Python is =. True False
- 11) To divide two integers to get an integer result, you can use //. True False
- 12) The keyword **elseif** can be used in if-statements in Python. True False
- 13) The operator + can be used to concatenate two strings together. True False
- 14) In Python, lists are mutable. True False
- 15) 45 % 8 produces the result 5. True False

$$48/8 = 6$$

$$45/8 = 5$$
$$45 \% 8 = 5$$

Question 2: Multiple Choice – 40 Marks (2 each)

- 16) Which of the following statements is/are TRUE about the CPU?
- a. CPU stands for Central Processing Unit
 - b. The CPU is what performs computation
 - c. The CPU processes machine language
 - d. **At least two of the above statements are true**
 - e. None of the above are true
- 17) What are two of the most important benefits of the Python language?
- a. Advanced mathematical equations and fast programs
 - b. Ease of use and fast programs
 - c. **Ease of use and portability**
 - d. Fast programs and smaller programs
- 18) Which statement(s) allows us to initialize the list `numbers` with 10 elements all set to zero?
- a. `numbers = [0]`
 - b. `numbers[10] = 0`
 - c. **`numbers = [0] * 10`**
 - d. `numbers[10] = [0] * 10`
- 19) Which of the following subtracts a variable `x` from a variable `y`, divides their difference by 3 and adds 11 to the result:
- a. $((x - y) / 3) + 11$
 - b. $x - y / 3 + 11$
 - c. $y - x / 3 + 11$
 - d. **$(y - x) / 3 + 11$**
 - e. None of the above are true
- Handwritten note for question 19: $(x - y) / 3 + 11$
- 20) What will be the values of the variables `num1` and `num2` after the execution of the following assignments?
- ```
num1 = 21
num2 = 18
num1 = num1 + num2 // 2
num2 = num1
```
- a. `num1` is 21, `num2` is 21
  - b. **`num1` is 30, `num2` is 30**
  - c. `num1` is 30, `num2` is 21
  - d. `num1` is 30, `num2` is 18
  - e. None of the above.

21) Which statement correctly creates a list that contains four elements?

- a. values[4]
- b. values = [4]
- c. values = [1, 2, 3, 4]
- d. value[4] = [1, 2, 3, 4]

*[x, y, z, R, [1, 2, 3, 4]]*

22) The following code snippet contains an error. What is the error?

```
cost = int(input("Enter the cost: "))
if cost > 100
 cost = cost - 10
print("Discounted cost:", cost)
```

- a. Logical error: use of an uninitialized variable
- b. Syntax error: missing colon after if statement
- c. Syntax error: missing an else statement
- d. Logical error: error in converting input

*python 语法错误*

23) Which of the following for loops will run the loop body 5 times?

- a. for i in range(13, 9, -1) :
- b. for i in range(14, 10, -1) :
- c. for i in range(15, 9, -1) :
- d. for i in range(14, 9, -1) :

*14, 13, 12, 11, 10*

*不运行*

24) Which of the following checks to see if there is a comma anywhere in the string variable name?

- a. if name.contains(",") :
- b. if "," not in name :
- c. if name.startswith(",") :
- d. if "," in name :

*有逗号吗??*

25) Which of the following statements is true about functions and strings:

- a. A function can be called with a string as an argument.
- b. A function can return a string.
- c. Only a. is true.
- d. Only b. is true.
- e. Both a. and b. are true.

26) What does the following code snippet output?

```
a = 7
b = 8
def fun(b,a):
 a=9
 b=8
 return a

fun(a,b)
print(a,b)
```

输出a但无修改数据.

- a. 7 8
- b. 8 9
- c. 9 9
- d. 8 8
- e. None of the above

27) What is the value of `names` after the following code segment has run?

```
names = []
names.append("Amy")
names.append("Bob")
names.pop()
names.append("Peg")
names[0] = "Cy"
names.insert(0, "Ravi")
names.insert(4, "Savannah")
```

名字未位置

- a. ["Amy", "Ravi", "Cy", "Peg", "Savannah"]
- b. ["Ravi", "Cy", "Peg", "Savannah"]
- c. ["Cy", "Bob", "Peg", "Savannah",]
- d. ["Ravi", "Amy", "Bob", "Savannah", "Peg"]

28) What is the output of the code snippet given below?

```
s = "zyxwv"
length = len(s)
i = 1
while i <= length // 2 :
 print(s[i-1], s[length - i])
 i = i + 1
```

- a. zx  
yw
- b. zy  
wv
- c. zx  
yv
- d. zv  
yw

29) What is printed by the following code snippet?

```
name = "This is London Ontario"
name = name.lower()
name = name.replace("o", "#")
name.upper()
print(name)
```

- a. "THIS IS LONDON ONTARIO"
- b. "THIS IS L#ND#N #NTARI#"
- c. "This is L#nd#n #ntari#"
- d. "this is l#nd#n #ntari#"

30) What is printed to the screen when this loop executes?

```
for i in range(24, 3, -7) :
 print(i, end = "-")
```

- a. 24-17-10-3-
- b. 24-17-10-3
- c. 24 17 10-
- d. 24-17-10-



31) What is printed from the following code snippet?

```
prices = [[1.0, 3.50, 7.50],
 [10.0, 30.50, 70.50],
 [100.0, 300.50, 700.50],
 [1000.0, 3000.50, 7000.50]]
print(prices[2][1])
```

- a. 10.0
- b. 30.50
- c. 100.00
- d. **300.50**

**Use the following code for the next three questions (i.e. questions 32 – 34 )**

```
num1 = int(input("Enter a number: "))
num2 = int(input("Enter a number: "))
num3 = int(input("Enter a number: "))
if not (num1 > num2 and num1 >= num3) :
 print("First num is", num1)
elif not (num2 > num1 and num2 > num3) :
 if num2 % 10 == 0:
 print("The value is", num2)
 elif num1 % 10 == 0:
 print("The value is", num1)
 else:
 print("The value is", num3)
elif not (num3 > num1 or num3 > num2) :
 print(num3)
```

32) Assuming a user enters 5, 10, and 15 as the input, what is the output of the above code snippet?

- a. The value is 10
- b. **First num is 5**
- c. 15
- d. The value is 15

33) Assuming a user enters 15, 10, and 5 as the input, what is the output of the above code snippet?

- a. First num is 15
- b. **The value is 10**
- c. 5
- d. The value is 15

34) Assuming a user enters 7, 7, and 6 as the input, what is the output of the following code snippet?

- a. **First num is 7**
- b. The value is 7
- c. 6
- d. The value is 6

35) What is the output of the following code snippet.

```
def myCalculator(n):
```

```
 i = 4
```

```
 x = 3
```

```
 y = 2
```

```
 while i > 0:
```

```
 y = y + n
```

```
 x = x + y % 3
```

```
 i = i-1
```

```
 return x
```

```
print(myCalculator(3))
```

- a) 10
- b) **11**
- c) 12
- d) 13

## SECTION 2

Answer questions in PART 3 AND PART 4 in **this** booklet. Answers recorded in any other location **WILL NOT** be graded

### Question 3: Logic Errors - Correcting Code Segments – 12 Marks

The function `substring` determines the position in a given string where another string occurs as a substring. For example, the string "ab" occurs in the string "abab" at positions 0 and 2. The function takes two parameters: `str` which is the given string and `s` which is the substring to look for. The function returns a list of the positions in `str` in which `s` occurs.

The function is syntactically correct but has four (4) incorrect lines of code which contain logic errors that prevent it from computing correctly. Identify the lines and correct them Note: a line may contain more than one logic error. Examples of a main program that uses the function, *when it is correctly implemented*, and sample output are also provided. (12 Marks)

#### Each error is 3 marks

```
Function to determine positions of a substring
in another string
def substring(str,s):
 posns = []
 for i in range(len(str)-len(s)+1):
 k = i
 j = 1 # j = 0
 done = False
 while j < len(s)-1 and not done: # j < len(s)
 if str[k] == s[j]:
 k += 1
 j += 1
 else:
 done = True
 if done: # not done
 posns.append(k) # i
 return posns

#
main program
str = "abab"
s = "ab"
lst = substring(str,s)
print(lst)

str = "abbbaab"
s = "bb"
lst = substring(str,s)
print(lst)
```

#### Output

```
[0, 2]
[1, 2]
```

## Question 4: A Little Code - 33 Marks

- 4.1) Create a function `countZeros` that counts the number of zeros in an integer. For example, if the integer is 90120, then the function would return 2. Provide your code for the function below (12 Marks).

**NOTE: This is ONE solution – there are a number of ways to provide this code!**

```
def countZeros(n):

def countZeros(n):
 strInt = str(n) # 4 converting integer
 count = 0
 for ch in strInt: # 2 for loop
 if ch == "0": # 4 for if and count
 count = count + 1
 return count # 2 for return
```

- 4.2) Create a function `removePunc` that takes a word containing letters, digits and punctuation and removes all the punctuation and returns a new word with no punctuation. The punctuation to consider is defined in the constant `PUNC` below. For example, if the word is "London's.", then the function would return "Londons". Provide your code for the function below (10 Marks).

**NOTE: This is ONE solution – there are a number of ways to provide this code!**

```
PUNC = ". , ; ' \" ? ! "

def removePunc(w):

def removePunc(w):
 neww = "" # 1 for initialization
 for ch in w: # 2 for loop
 if ch not in PUNC: # 3 for if
 neww = neww + ch # 2 for correct assignment
 # and concatenation
 return neww # 2 for correct return
```

- 4.3) Create a function `wordList` that takes a line of text (a sentence) and splits it into words and returns a list of words where each word has no punctuation. This function should make use of the function `removePunc` that was defined above (4.2). Provide your code for the function below (11 Marks).

**NOTE: This is ONE solution – there are a number of ways to provide this code!**

```
def wordList(line):
```

```
def wordList(line):
 words = [] # 1 for initialization
 lst = line.split() # 2 for splitting line
 for w in lst: # 1 for loop through list
 w = removePunc(w) # 3 correct use
of "removePunc"
 words.append(w) # 3 for correct append
 return words # 1 for return
```