CSE-101, Introduction to Programming
Midterm Exam, 2018

Name: _____ Marks: 25
Roll Number:_____ Time: 60 minutes
Section: _____
Group: _____

Instructions:
1. You will be expected to write Python code in this exam. We recommend that you draw vertical lines to make your indentation clear.
2. Assume the use of Python3 in all of the questions below.
3. Write your details on both the question paper and the answer sheet. Only the answers written in the answer sheet will be evaluated.
4. There are 2 sheets in this question paper printed both sides. There are a total of 4 questions and Q1 and Q2 have subparts. Mention the question number and the subpart number clearly.

Question 1: Write the output of the following programs:

a. (4 Marks)

```
x = 'Intro to programming' < = 'intro to programming'
y = not True and False or not False
z = 2**-3**3 >= -512
print('x= ' + str(x))
print('y= ' + str(y))
print('z= ' + str(z))
if (x and y and z):
    x= y > z
    print ('x='+ str(x))
else:
    x = y and z
    print ('x='+ str(x))
```

Ans:
x= True
y= True
z= True
x=False

b. (1 Mark)

```
x = "foo"
y = "bar"
print (x[-1:1]*2 + y[1:-1]*4) # Use '_' for one space
```

Ans:
aaaa

Question 2: Please read the question carefully and answer the following:

a) (4 Marks)

Consider the code given below. What should the values for variables ***start***, ***end***, ***first*** and ***last*** be assigned at the beginning of this code if the required output is:

15 20 25 30
18 24 30 36
21 28 35 42

```
#variable assignment lines
start = _______
end = ________
first = ________
last = _______
m = start+5
while m  <= end-1:
        n= first
        while n < last:
            print ( " %5d " %(m*n), end = ' ')
            n += 1
        m += 1
        print()
```

Ans:
start = 0
end = 8
first = 3
last = 7

b) (2 Marks)

Can we assign ***integer*** values to variables x and y so that the string "CSE101" is printed out. If yes, what are those values for x and y?

```
x=_______
y=_______
if (x>=1) and (0<=y<=3):
    print("CSE103")
elif not(y<=4 or y>=1):
    print("CSE102")
elif x<=1 or x>=3:
    print("PSY03")
else:
    print("CSE101")
```

Ans:

x=2

y can take all integral values except 0,1,2,3

c) (2 Marks)

What would be the output of the function call: `method_3c(1325476)?`

```
def method_3c(number):
    x= number
    count = 0
    while (x > 0):
        x= int(x/10)
        count += 1
    i = 2
    while ( i < int(count/3)):
        number = int(number /10)
        i += 2
        count= count+1
    return number%10
```

Ans:

6

d) (2 Marks)

Consider the following script:

```
k=10
while k <=100:
    print(k)
    k= k + 5
```

Write an equivalent script that makes effective use of a for-loop instead of while loop.

Ans:

```
for k in range(10, 101, 5):
    print(k)
```

Question 3: (5 Marks) Write a function isNumberOkay(number) that returns True if the parameter number $a_1a_2a_3a_4....a_n$ of ***n*** digits satisfies the following pattern and False otherwise:

- Number $a_1a_2a_3a_4....a_n$ is a positive integer, and
- For all 1 <= i <= int(n/2), both a_i and a_{n-i+1} are either both even or odd.

```
def isNumberOkay(number):
""" Returns: True if number satisfies following two conditions
1. number a1a2a3a4....an is a positive integer, and
2. For all 1 <= i <= int(n/2), both ai and an-i+1 are either both
even or odd.
Example: Inputs 41770, 30387, 7777, 6752 will return True.
Precondition: number is an integer object with at least 2
digits. """
```

Ans:

```
def isNumberOkay(number):
        num=str(number)
        numdig= len(num)
        if number < 0 :
                return False
        i=1
        while i<=int(numdig/2) :
                if int(num[i-1])%2 != int(num[numdig-i])%2 :
                        return False
                i=i+1
        return True
```

Question 4: (5 Marks) In the US, 10-digit telephone numbers are typically represented in one of the two following styles:
“Parenthetical”: (555) 666-1110
“Dashed”: 555-666-1110
There is no whitespace in a Dashed phone number: they are all exactly 12 characters long. There is only one space in a Parenthetical phone number, and it is after the “)”; they are all exactly 14 characters long, counting the space.
Implement the following function according to its specification.

```
def phone_to_paren(s):
""" Returns: a string representing the phone number s in
Parenthetical form.
Precondition: s is a non-empty string that *would* be a valid
Dashed phone number EXCEPT that it possibly has spaces around
the dashes.
Examples of valid input:
555-666-1110
555 - 666 - 1110
555 - 666-1110
,etc., ... all yield the same output:
(555) 666-1110
"""
```

Ans:

```
def phone_to_paren(s):
    phno = str(s)
    if len(phno) >= 12:
        i= 0
        out= '('

        for c in phno:

            if c.isdigit() == True :
                out = out + c
                i= i+1

            if i == 3:
                out = out + ') '
                i= i+1
```

```
            if i == 7:
                out = out + '-'
                i= i+1

            if i == 14:
                break

        return out

    else:

        out = "Invalid Input"
        return out
```