There are 13 questions on this test. Record your answers to the first 10 questions below. Answer questions 11A, 11B, 12 and 13 on the designated attached pages. There are also scratch pages for the programming problems. The value of each question is:

1-10 multiple choice (4 pts each)
11-13 programming (20 pts each)

Allocate your time accordingly. We will give partial credit for questions 11A, 11B, 12 and 13. Answer them as completely as you can. If you finish early, use the extra time to double check your work. You may use the summary of Python language elements that is provided. You may not use notes, books or electronic devices of any sort. All cell phones and other mobile devices must be turned off during the exam. Be sure to sign the hand-in sheet when you hand in your exam.

Good luck!

Print Name _________________________________ Student ID ____________

Section ____________________________ (see section schedule below)

S2 TR 8:30; S4 WF 1:00; S6 WF 10:00; S8 MR 2:30; S10 TR 10:00
SECTION 1: Multiple choice.

Problems 1-10 are multiple choice (4 points each). Choose the letter that gives the output of the given code fragment. Record your answer on the front page of the answer packet.

Question 1
def fifthOneOdd(numlist):
    for i in range(0, len(numlist)):
        if i%5 != 0:
            continue
        if numlist[i]%2 != 1:
            return False
    return True

nums = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
print(fifthOneOdd(nums))

a. True
b. False
c. False True
d. SyntaxError: multiple return statements
e. None of the above

Question 2

def hasVowel(wordList):
    vowels = 'aeiouAEIOU'
    rtnList = []
    for word in wordList:
        for letter in word:
            if letter in vowels:
                if letter in vowels:
                    rtnList.append(word)
                break
    return rtnList

subwayAd = ['Rd', 'ths', 'to', 'gt', 'a', 'gd', 'jb']
print(hasVowel(subwayAd))

a. []
b. ['to']
c. ['to', 'a']
d. TypeError: string in list
e. None of the above
Question 3

```python
aString = 'Springsteen, Springsteen, oh, Springsteen!'
aList = aString.split()
bossCount = 0
for word in aList:
    if word == 'Springsteen':
        bossCount += 1
print(bossCount)
```

a. 0  
b. 1  
c. 2  
d. 3  
e. None of the above

Question 4

```python
titles = ['Ulysses', 'Typhoon', 'Eeyore', 'Brigadoon', 'Oo! Ah! Ee!']
double = []
for title in titles:
    for i in range(len(title)-1):
        if title[i] == title[i+1]:
            double.append(title)
print(double)
```

a. []  
b. ['Typhoon', 'Brigadoon']  
c. ['Ulysses', 'Typhoon', 'Brigadoon']  
d. ['Ulysses', 'Typhoon', 'Eeyore', 'Brigadoon', 'Oo! Ah! Ee!']  
e. None of the above
Question 5

```python
def vowelTest(testStr):
    vowels = 'aeiou'
    vowelsInTestStr = ''
    for letter in testStr:
        if letter in vowels:
            vowelsInTestStr += letter
    return vowelsInTestStr

print(vowelTest(t))
```

a. oueai
b. oueaiee
c. aeiou
d. aeeeiou
e. None of the above

Question 6

```python
order = [['1st', '2nd'], {'2nd': 'John', '1st': 'George'}, '1st', '2nd']
print(order[1][1])
```

a. SyntaxError
b. 0
c. 2
d. [0, 1, 2]
e. None of the above

Question 7

```python
presidents = {'Jefferson':3, 'Adams':2, 'Washington':1}
print(presidents[1])
```

a. Jefferson
b. Adams
c. Washington
d. KeyError: 1
e. None of the above
Question 8

```python
opVals = [not True, not not True, False and not True, not False and True]
falseCount = 0
for expr in opVals:
    if expr == False:
        falseCount += 1
print(falseCount)
```

a. 1
b. 2
c. 3
d. 4
e. None of the above

Question 9

The lines below are the content of the file named 'thunder.txt'.

```
Well the nights busting open
These two lanes will take us anywhere
We got one last chance to make it real
To trade in these wings on some wheels
```

After the execution of the following code, what is the content of the file 'thunderOut.txt'?

```python
inF = open('thunder.txt', 'r')
outF = open('thunderOut.txt', 'w')
for line in inF:
    if 'we' in line.lower():
        outF.write(line)
inF.close()
outF.close()
```

a. We got one last chance to make it real
b. Well the nights busting open
   We got one last chance to make it real
c. Well the nights busting open
   We got one last chance to make it real
   To trade in these wings on some wheels
d. thunderOut.txt is an empty file
e. none of the above
Question 10

```python
from turtle import *
t = Turtle()
for i in range(4):
    if i%3 == 0:
        t.forward(100)
        t.right(90)
    elif i%2 == 0:
        t.forward(100)
    else:
        t.forward(100)
```

a. Invalid syntax
b. One line of length 100
c. One line of length 200
d. Two perpendicular lines
e. None of the above
SECTION 2: Programming Problems.

Problems 11A, 11B, 12 and 13 are programming problems. Write your answers on the designated pages of the answer packet.

Question 11
Part a (10 points)
Write a function name tri() that takes two parameters:

1. t -- a turtle used for drawing and
2. size -- the length of a side of a triangle

tri() should draw an equilateral triangle in which the sides are of length size, leaving the turtle t in the initial position and orientation. For full credit, repeated operations must be performed by loops.

Question 11
Part b (10 points)

Write a function named trifecta() that takes three parameters:

1. size
2. angle
3. num

trifecta() should call tri() repeatedly so as to draw num triangles, each with sides of length size. Each triangle should be oriented angle degrees clockwise from the preceding triangle. trifecta() should create a turtle and pass it to tri() as a parameter. For full credit, repeated operations must be performed by loops.

For example, the function call trifecta(100, 25, 5) should produce the following output.

![Diagram of multiple triangles drawn by trifecta function call](image-url)
Question 12

Write a function named `makeWordList()` that takes two parameters:

1. `readFileName` is a string that is the name of a file containing text
2. `writeFileName` is a string that is the name of a file that `makeWordList()` writes its output to.

Each line in the output file should consist of one of the words in the input file, followed by a space, followed by the number of occurrences of that word in the input file, including both capitalized and lower case occurrences. You may assume that the file contains only letters – no numbers or punctuation marks.

For example, if the input file contains the following text

*You can steer yourself any direction you choose*

then the output file should have the following content:

```
any 1
yourself 1
choose 1
you 2
direction 1
can 1
steer 1
```

Question 13

Write a function named `vowelContent()` with the following input and output:

**Input:** a list parameter, `wordList`. The characters in the words in `wordList` are all lower case letters.

**Return:** a dictionary consisting of three key/value pairs. The keys are the strings 'mostly vowels', 'mostly consonants' and 'half vowels'. The value of each keys, respectively, is a list of words (without duplicates) consisting of, respectively, words in `wordList` that are more than half vowels, less than half vowels, and exactly half vowels. Vowels are the letters 'a', 'e', 'i', 'o', and 'u'.

For example, the following would be correct output.

```python
wList = ['its', 'a', 'death', 'trap', 'its', 'a', 'suicide', 'rap', 'we',
        'gotta', 'get', 'out', 'while', 'were', 'young']
print(vowelContent(wList))

{'half vowels': ['we', 'were'], 'mostly consonants': ['its', 'death', 'trap',
                                                    'rap', 'gotta', 'get', 'out', 'while', 'young'],
 'mostly vowels': ['a', 'suicide', 'out']}
```