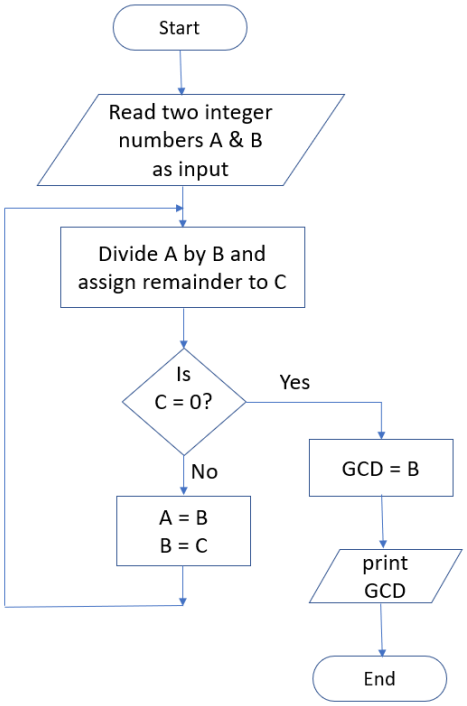
CSE101 – Midterm Exam 1

11-Apr-2019 **Total points: 50**

Name: _____ **Student ID #** _____

Instructions: Read the questions carefully before attempting to write the answer. Write the answers in the space provided below each question. Use of pencil is encouraged, so that you can erase and overwrite. Make sure that your handwriting is legible. Rough work sheet is provided at the end of the answer sheet to be used only for rough work, not for writing answers.

1. Given the following flowchart, write corresponding Python code (without using function) to calculate the greatest common divisor (GCD) of two integers A & B using Euclidian method. The Python code should be accurate and should compile and run successfully. (6 pts)



Answer

```
a = int(input("enter a:"))
b = int(input("enter b:"))
gcd = 0
while (a > 0 and b > 0):   #or while (true)
    c = a%b
    if c == 0:
        gcd = b
        break
    a = b
    b = c

print("gcd",gcd)
```

2. Suppose the following strings are defined in an interactive session:

```
>>> proverb = "Others' rice cakes always look bigger."
>>> wordlist = proverb.split()
>>> word = "cakes"
```

What will Python print as the value of the following expressions? (6 pts)

a. `len(wordlist)` 6

b. `print(wordlist)` ["Others' ", "rice", "cakes", "look", "bigger."]

c. `word in wordlist` True

d. `proverb.lower().count('o')` 3

e. `wordlist[:3]` ["Others' ", "rice", "cakes"]

f. `wordlist[-2:]` ["look", "bigger."]

3. Write two Python functions corresponding to following formulas for calculating and returning surface area and volume of a right circular cone. Both of the functions accept as input h (height of cone) and r (radius of cone) as shown in the following figure. Make use of math.pi value in your code. Return result upto two decimal places using function round. (8 pts)

Formula for volume of a cone:

$$V = \pi r^2 \frac{h}{3}$$

Formula for surface area of a cone:

$$A = \pi r \left(r + \sqrt{h^2 + r^2} \right)$$

Answer:

```
import math
def coneVolume(h, r):
```

v = math.pi*(r**2)*h*1/3
return round(v,2)

```
def coneSurfaceArea(h, r):
```

a = math.pi*r(r + (h**2 + r**2)**0.5)
return round(a,2)

```
#testcases
>>>print(coneVolume(10,5))
261.8
>>>print(coneSurfaceArea(10,5))
254.16
```

4. Given a variable x that refers to a string as follows, use string slicing to write the following expressions:

```
x = ' Seutobwochie'
```

a. An expression that is equivalent to the string containing the last 6 characters of x. Your expression should evaluate to `'wochie'`. (2 pts)

Answer: x[6:] or x[7:]

b. An expression that would extract the word `'Seutob'` from the above string x. (2 pts)

Answer: x[:6] or x[1:7]

c. An expression that would extract the word `'tobwoc'` from the above string x. (2 pts)

Answer: x[3:9] or x[4:10]

5. Suppose the following list is defined in an interactive What is the output of each of the following code fragments? There are no syntax errors in the code. (6 pts)

```
>>> a = ["achim", "jeong-o", "jeonyeog", "bam"]
```

a. `print(len(a))` 4

b. `print(len(a[1]))` 7

c. `print(a[2][1:5])` eony

d. `print(a[3]*3)` bambambam

e. `print(a[0] + a[3])` achimbam

f. `print(a[1][-3:])` g-o

6. What value will be stored in the variable result after the following code has been executed? (2 pts)

```
nums = [6, 5, 4, 8, 10, 12, 5, 7, 0, 9, 11]
result = []
for n in range (int(len(nums) / 2)):
      result.append(nums[n])
```

a. [6, 5, 4, 8, 10, 12]
b. [6, 5, 4, 8, 10]
c. [6, 5, 4, 8]
d. None of the above

7. What is the output of the following code fragment? (2 pts)

```
name = "Madagascar island"
result = ""
for ch in name:
    if ch.upper() not in 'AS':
          result += ch
print(result)
```

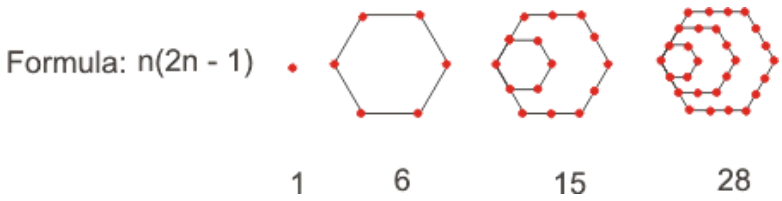
a. Madagascar island
b. Mdgscr islnd
c. Mdgcr ilnd
d. Madagacar iland

8. What is the output of the following code fragment? (2 pts)

```
d = 10
e = 12
f = 8
if e - f == f - 1:
      print('X')
elif d - f > e:
      print('Y')
else:
      print('Z')
```

a. X
b. Y
c. Z
d. None of the above

9. A **hexagonal numbers series** along with a formula for n^{th} term is given as follows:



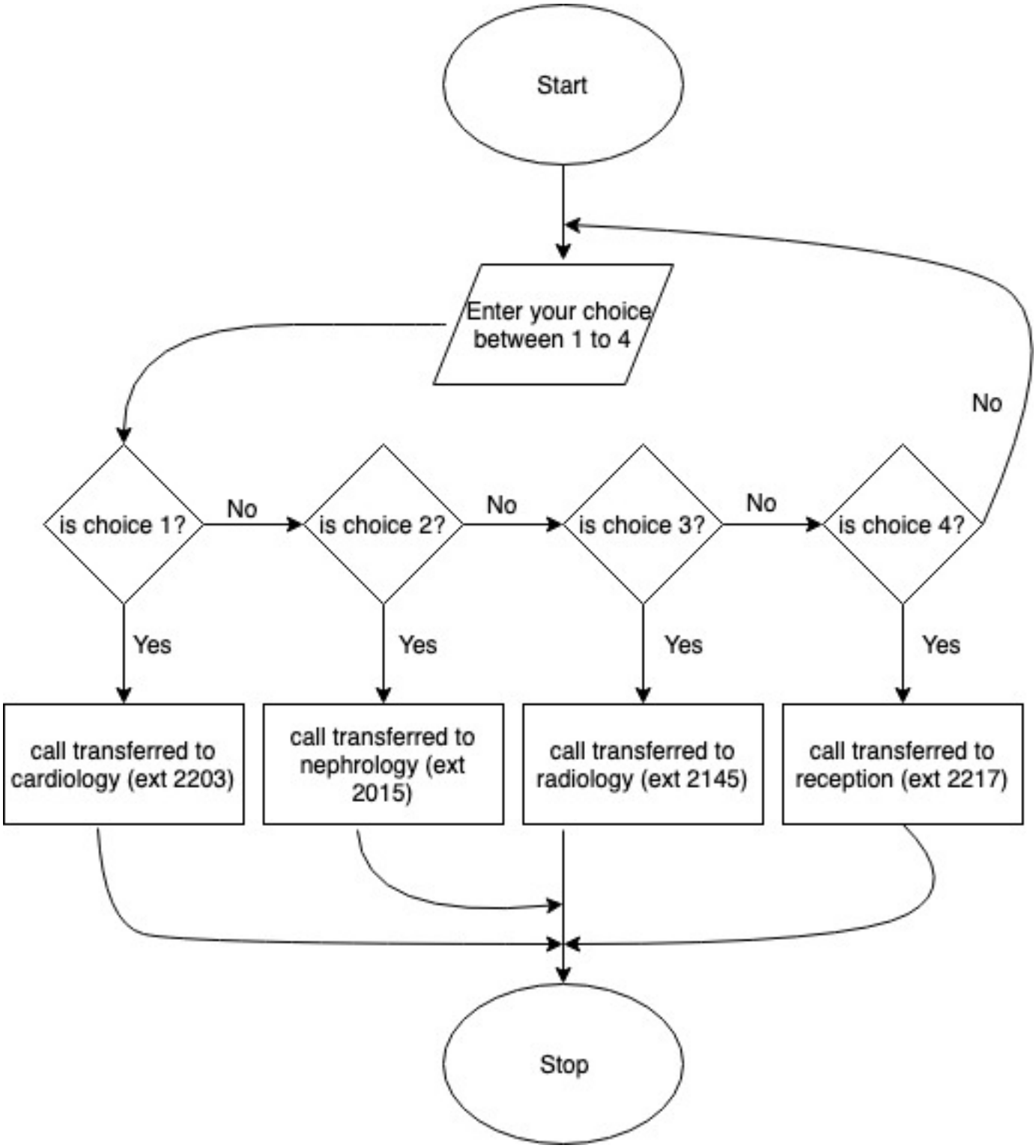
Write a function with the following definition to return a list of n elements in a hexagonal number series where n is the number of terms in a series. (6 pts)

```
def hexagonalSeries(n):
        res = []
        for i in range(1,n+1):
            res.append(i*(2*i-1))

        return res
```

```
#testcase
>>> print(hexagonalSeries(6))
[1, 6, 15, 28, 45, 66]
```

10. Create a flowchart for the interactive voice response program in a hospital that receives a user call and asks user for input between numbers 1 to 4 and transfers calls to appropriate department depending on the user input 1, 2, 3 or 4. The departments to be connected are: input 1 -> Cardiology (ext. 2203), input 2 -> Nephrology (ext. 2015), input 3 -> Radiology (2145), input 4 -> Receptionist (ext. 2217). If the user presses inputs other than 1 to 4, the program asks user for valid input and continues till the user has entered valid input. (6 pts)


Start
Enter your choice between 1 to 4
is choice 1?
No
is choice 2?
No
is choice 3?
No
is choice 4?
No
Yes
Yes
Yes
Yes
call transferred to cardiology (ext 2203)
call transferred to nephrology (ext 2015)
call transferred to radiology (ext 2145)
call transferred to reception (ext 2217)
Stop

Rough Work

. 1: Linear Search and insertion sort

A. Suppose a list is defined with this statement:

(3 pts)

```
>>> languages = ['French', 'Korean', 'English', 'Nepali',
'Tagalog']
>>> from PythonLabs.IterationLab import isearch
```

Describe what would be printed in your shell window if you were to type these expressions:

1. `isearch(languages, 'English')` **2**
2. `isearch(languages, 'Korean')` **1**
3. `isearch(languages, 'Hindi')` **(Nothing, None, -1)**

B. Suppose a list a is defined with this statement:

(3 pts)

```
>>> a = [12, 5, 8, 14, 6, 7, 4, 10, 2, 9]
```

How many comparisons will be made by the following searches using the linear search method?

1. `isearch(a, 7)` **6**
2. `isearch(a, 2)` **9**
3. `isearch(a, 19)` **10**

C. Assume a list is defined with this statement:

(4 pts)

```
>>> gases = ['He', 'Ar', 'Ne', 'Kr', 'Xe', 'Rn']
>>>from PythonLabs.IterationLab  import isort
```

Explain how the list would be sorted by a call to isort. Here are the first two lines, to get you started:

```
>>> isort(gases) #isort stands for insertion sort
['He'] ['Ar', 'Ne', 'Kr', 'Xe', 'Rn']
['Ar', 'He'] ['Ne', 'Kr', 'Xe', 'Rn']
['Ar', 'He', 'Ne'] ['Kr', 'Xe', 'Rn']
['Ar', 'He', 'Kr', 'Ne'] ['Xe', 'Rn']
['Ar', 'He', 'Kr', 'Ne', 'Xe'] ['Rn']
['Ar', 'He', 'Kr', 'Ne', 'Rn', 'Xe']
```

CSE 101 Midterm 1 (Practice Version) KEY

- Please place **ALL** of your final answers on the answer sheet that you were given at the start of the exam. Answers written on this question packet will **NOT** be graded.
- The problems in this exam are not necessarily ordered by difficulty, so be sure to look through the entire exam before starting.
- This exam is **CLOSED-BOOK, CLOSED-NOTES**. You may **NOT** use any computers, calculators, smartphones, or any other electronic devices during the exam. The presence of any such devices or notes will be treated as evidence of academic dishonesty.
- Any **CHEATING** will result in a 0 for the exam, as well as charges of academic dishonesty. Do not:
  1. interact with (for example, speak to) your neighbors during the exam, or
  2. place anything in a location where it may appear that you are sharing information or copying (or allowing others to copy your work)!
- Please raise your hand or come to the front of the room if you have any questions during the exam.
- When you are finished, please return both your answer sheet **AND** your question packet to the instructor and TAs.
- You must present photo ID (student ID, driver's license, etc.) when submitting your final work.

DO NOT BEGIN WORK UNTIL YOU ARE TOLD TO DO SO!

(2 pts) 1. Consider the following Python code fragment:

```
product = 0

for i in range(1, x):
    product = product * i

print(product)
```

If the variable `x` is initialized to 5, this code will:

A. print the value 5

B. print the value 24

C. print the value 120

D. print the value 0

E. print an error message

(2 pts) 2. Which of the following is **NOT** required for an algorithm?

A. A precise statement of the starting conditions

B. A specification of the algorithm's termination condition

C. An electronic computing platform

D. A series of simple yet detailed steps

E. All of the above are required to create and implement an algorithm

(2 pts) 3. The value of the arithmetic expression `5 - 3 * 2` is

A. -1

B. 4

C. 16

D. 0

E. None of the above

(2 pts) 4. The value of the arithmetic expression `17 // 5` is

A. 3.4

B. 3

C. 4

D. 5

E. None of the above

(2 pts) 5. The value of the arithmetic expression `14 / 3` is

A. 4

B. 2

C. 4.6667

D. 0.667

E. None of the above

(2 pts) 6. Which of the following is **NOT** a valid Python arithmetic operator?

A. *

B. **

C. /

D. //

E. #

(2 pts) 7. Which of the following is **NOT** a legal variable name in Python?

A. length

B. CaptainAmerica

C. _measure

D. 99balloons

E. None of the above; all four are valid variable names

(2 pts) 8. What output will the following Python code fragment produce?

```
def mystery (f):
    f = f * 2
    print(f)

f = 5
mystery(f)
print(f)
```

A.
```
5
5
```

B.
```
5
10
```

C.
```
10
10
```

D.
```
10
5
```

E. Nothing; an error message will be printed instead

(2 pts) 9. Suppose that the variable `income` has the value 4001. What will be displayed by the following code?

```
if income > 3000:
    print("Income is greater than 3000")
if income > 4000:
    print ("Income is greater than 4000")
```

A. Nothing will be displayed

B. *Income is greater than 3000*

C. *Income is greater than 3000* followed by *Income is greater than 4000*

D. *Income is greater than 4000*

E. *Income is greater than 4000* followed by *Income is greater than 3000*

(2 pts) 10. Suppose that the variable `income` has the value 4500. What will be displayed by the following code?

```
if income > 3000:
    print("Income is greater than 3000")
elif income > 4000:
    print ("Income is greater than 4000")
```

A. Nothing will be displayed

B. *Income is greater than 3000*

C. *Income is greater than 3000* followed by *Income is greater than 4000*

D. *Income is greater than 4000*

E. *Income is greater than 4000* followed by *Income is greater than 3000*

(2 pts) 11. Which of the following is the correct "does not equal" operator in Python?

A. !==

B. !=

C. =/=

D. !

E. None of the above

(2 pts) 12. Given the Python statement below, what value would be returned by `len(s)`?

```
s = "I'm a little teapot!"
```

A. 15

B. 17

C. 18

D. 20

E. 22

(2 pts) 13. The index of the last character in the Python string `text` is equal to:

A. 0

B. `len(text) - 1`

C. `len(text)`

D. `text - 1`

E. None of the above

(2 pts) 14. Consider the following Python code fragment. What values will it print?

```
for x in range(1, 11, 3):
    print(x)
```

A. 1, 3, 6, 9

B. 1, 4, 7, 10

C. 2, 5, 8

D. 2, 5, 8, 11

E. 3, 6, 9

(2 pts) 15. Assume that the string "candy" has been assigned to the Python variable `c`. What will the following Python code fragment print?

```
for i in range(4, 1, -1):
    print(c[i], end="")
print()
```

A. cand

B. andy

C. ydn

D. ydna

E. ydnac

(2 pts) 16. What value will the following Python function return for the input "pandemonium"?

```
def mystery(text):
    result = ""
    for x in text:
        if x == 'p':
            result += "pop"
        elif x == 'm':
            result = ""
        else:
            result += x
    return result
```

A. `popandemonium`

B. `pande`

C. `pandemonium`

D. `""` (the empty string)

E. `pandeoniu`

(2 pts) 17. The `for` loop below is designed to add up all of the even integers (and *only* the even integers) in the range 6 through 104 (inclusive) and save the result into a variable named `total`. What code fragment should go in the blank space below?

```
total = 0

____________________ : # your code goes here
    total = total + x
```

A. `for x in range(6, 105)`

B. `for x in range(6, 104, 2)`

C. `for x in range(6, 105, 2)`

D. None of the above

(2 pts) 18. Given two Python strings defined as follows:

```
s1 = "heuristic"
s2 = "algorithm"
```

Which of the following expressions will return the value `"urgor"`?

A. s1[2:3] + s2[2:5]

B. s1[2:4] + s2[2:4]

C. s1[3:5] + s2[3:6]

D. s1[2:4] + s2[2:5]

E. None of the above

(2 pts) 19. Consider the following function:

```
def blah(n):
    d = 0
    while n > 0:
        d = d * 10
        d = d + n % 10
        n = n // 10
    return d
```

What value will the function call `blah(30792)` return?

A. 27903

B. 32097

C. 29730

D. 29703

E. None of the above

(2 pts) 20. What is the result of translating the base 10 value 38 into base 6?

A. 26

B. 201

C. 102

D. 123

E. None of the above

(2 pts) 21. Translate the three values below from their indicated current bases into base 10, and then identify the one whose value is different from the others (or select "None of the above" if all three base 10 values are identical).

A. 111 (base 8)

B. 141 (base 6)

C. 331 (base 4)

D. None of the above (all three values are identical)

111 base 8 = 64 + 8 + 1 = 73 base 10

*141 base 6 = 36 + 4*6 + 1 = 61 base 10*

*331 base 4 = 3*16 + 3*4 + 1 = 61 base 10*

(2 pts) 22. The code below prints out a pattern of asterisks. How many asterisks in all will be printed to the screen?

```
for x in range(6):
    for y in range(x+1):
        print("*", end="")
    print()
```

A. 5

B. 6

C. 21

D. 36

E. None of the above

(12 pts) 23. A student wrote the `roomArea()` function below, in order to collect the dimensions of a rectangular room, calculate its area, and return the answer. **BRIEFLY** identify **ALL** of the errors (in terms of both Python syntax and overall program logic) that you can find in this function.

```
def roomArea ():
    area = length ** width
    length = input("Please enter the length of the room in feet: ")
    return length
    width == input("Please enter the width of the room in feet: ")
    return area
```

- 'length' and 'width' should be read in before calculating the area
- The line that reads in 'length' should use int() to convert the input into an integer value
- The line that reads in 'width' should use int() to convert the input into an integer value
- The line that reads in 'width' uses the comparison operator (==) instead of assignment (=)
- The area calculation uses exponentiation (**) instead of multiplication (*)
- The function returns the value of 'length' (improperly) before it returns the area of the room

(6 pts) 24. Consider the following Python function definition and determine what value will be returned by the function call that follows.

```
def calculate(value, src):
    result = 0
    val_string = str(value)
    size = len(val_string)
    mult = src ** (size - 1)

    for digit in val_string:
        d = int(digit)
        result += d * mult
        mult = mult // src

    return result

calculate(93, 2)
```

21

Quick (summarized) explanation:

1. 'size' is set to 2 (the number of digits in 'value')
2. 'mult' is set to 2 (the value of 'src'), raised to the first power ('size' - 1)
3. In the loop, 'd' is first set to 9. We add 9 * 2, or 18, to 'result'. Then 'mult' is divided by 2 using floor division, giving us 1.
4. The second round of the loop sets 'd' to 3. We add 3 * 1, or 3, to 'result', giving us a final answer of 21.

(6 pts) 25. Write the decimal (base 10) equivalent of the hexadecimal (base 16) value 2B. Be sure to show your work in order to receive partial credit!

Solution 1: *2B = (2 * 16^1) + (11 * 16^0) = 32 + 11 = 43.*

Solution 2: *2B is equivalent to 00101011 in binary. 00101011 in binary is equal to 32 + 8 + 2 + 1, or 43.*

(8 pts) 26. Complete the Python `encode()` function below, which takes a string as input and returns a new string (or "error") according to the following process:

1. Start with `d` = 1 and `s` = `""`.
2. Process the input string one character at a time. If you encounter a '*' character, multiply `d` by 2. If you encounter any other character, append it to the end of `s` and subtract 1 from `d`.
3. As soon as `d` reaches 0, return `s`.
4. If you run out of input characters (i.e., reach the end of the string argument) before `d` reaches 0, return the string "error".

```
def encode (input):
    d = 1
    s = ""

    for character in input:
        if character == "*":
            d = d * 2
        else:
            s = s + character
            d = d - 1

        if d == 0:
            return s

    return "error"
```

(6 pts) 27. Consider a geometric sequence defined by the terms

$$a, ar, ar^2, ar^3, \ldots, ar^{n-1}$$

The first term of the sequence (meaning n is 1) is just the value of a. The second term (n is 2) is a times r, and so on. Complete the `nthTerm()` function below, which takes three integer arguments (a, r, and n) and returns the value of the nth term of this sequence.

```
def nthTerm (a, r, n):
    if n == 1:
        return a
    else:
        return a * (r ** (n-1))
```

(8 pts) 28. Complete the Python function below, which recognizes (matches) strings that belong to a special language. The rules of this language are as follows:

- All strings in this language contain only the letters a and b
- A string only belongs to this language if it has equal numbers of as and bs
- At any given point in the string, the number of bs that have been encountered must be less than or equal to the number of as seen so far

For example, *aabb* is a valid string in this language, but *abba* is not (once the third character has been read, we have seen more bs than as).

```
# Returns True if input belongs to the language, and False otherwise
# You may assume that input only contains 'a' and 'b' characters

def isInLanguage (input):
    num_as = 0
    num_bs = 0

    for letter in input:
        if letter == "a":
            num_as = num_as + 1
        else:
            num_bs = num_bs + 1

        if num_bs > num_as:
            return False # b's outnumber a's, so we abort processing

    if num_as == num_bs:
        return True # equal numbers of a's and b's
    else:
        return False
```