Test ONE

NOTE: There are two programming problems at the end that go in your bluebook. There are several questions near the end of this document that require programming-thinking.

A good strategy would be to skip any questions you cannot do quickly to get to programming questions, then go back through completing the shorter ones.

Closed book, closed notes, NO electronic devices of any type

You will have ONE HOUR AND 45 MINUTES to complete this test

Do NOT tear any pages out of your Blue Book.

Do NOT tear any pages from this document.

Be sure to hand in all 9 pages of this test.

Do not remove this page.

Place your answers for questions 1 through 21 (inclusive) in this document.

If you write any of these answers in your Blue Book you will get NO CREDIT for them.

Place your answer for questions 22 and 23 in your Blue Book.

Put your name and ID on each page in this document after this page.

CIRCLE YOUR LAST NAME

If you need "scratch" paper, use your Blue Book but cross out anything you do not want graded.

There are 23 questions totaling 189 points

There are no optional questions. You must earn 189 points to score a 100%

Do not begin until you are instructed to do so

If you continue writing after time has been called you will have earned a ZERO.

If your cell phone rings during this test or if you answer your cell phone, you will receive a ZERO.

All cell phones must be inside bags or pockets during the entire test.
No cell phones are allowed on your desk at any time during the test.

Turn off all cell phones before the test starts

(If you are expecting a call, BEFORE the test starts leave your phone with the test monitor)
For the following questions, use these variable definitions.

\[
\begin{align*}
a &= 37 \\
b &= 10 \\
c &= 2.1 \\
d &= '3' \\
\end{align*}
\]

What is the type and value of each of the following expressions or, if it won't compile, circle that answer

<table>
<thead>
<tr>
<th></th>
<th>type</th>
<th>value</th>
<th>circle if will not compile or there will be a runtime error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>a // b</td>
<td></td>
<td>error will occur</td>
</tr>
<tr>
<td>2</td>
<td>a % b</td>
<td></td>
<td>error will occur</td>
</tr>
<tr>
<td>3</td>
<td>(c * 4) / 2.1</td>
<td></td>
<td>error will occur</td>
</tr>
<tr>
<td>4</td>
<td>float(4) - 2 ** 3</td>
<td></td>
<td>error will occur</td>
</tr>
<tr>
<td>5</td>
<td>bin( 5 )</td>
<td></td>
<td>error will occur</td>
</tr>
<tr>
<td>6</td>
<td>d * int(d)</td>
<td></td>
<td>error will occur</td>
</tr>
<tr>
<td>7</td>
<td>str( c )</td>
<td></td>
<td>error will occur</td>
</tr>
<tr>
<td>8</td>
<td>a &lt; c &lt; b or b == 1</td>
<td></td>
<td>error will occur</td>
</tr>
<tr>
<td>9</td>
<td>b + c</td>
<td></td>
<td>error will occur</td>
</tr>
<tr>
<td>10</td>
<td>b &lt; 4 or not</td>
<td></td>
<td>error will occur</td>
</tr>
<tr>
<td>11</td>
<td>b if a &lt; b else c</td>
<td></td>
<td>error will occur</td>
</tr>
</tbody>
</table>

12. (2 pts) (on the next several lines)

```
'hi
lo'''
```

will not compile
13. (3 pts) Given these constants, complete the function, implementing its docstring.

```python
INVALID_INPUT_PROMPT = "Sorry, try again"
UNKNOWN_ERROR = 33333333
MANAGERS_NAME = 'JOE'
SENTINEL_VALUE = -99999

def killTheProgramOnUnkownError():
    ''' Ends the program using the correct error code.
    Should be called only when an unknown error has occurred.
    Explains that the program will end and why. '''
    print( "Ending program. A unknown error has occurred." )
```

A programmer has been hired by a contest giving company that has very strict and well publicized age limits for entering their contests. The have hired this programmer to write a function for that takes a user's age and returns whether or not the user is old enough to enter their contest (they have only one contest).

Note that this question is NOT about branching.

Do NOT write this function. ONLY answer this question:

14. (10 pts) What are the pieces of information this programmer needs?
   Give the source of information for each (if no information is needed, just write NO INFO).
   Note that we are NOT asking for the implementation.
   Consider both good design as well as the fact that the contest has a very specific age limit.
   Use the terminology for your answer that is used in class and in the website.
   If a piece of information might have two sources, state both.

<table>
<thead>
<tr>
<th>INFORMATION NEEDED</th>
<th>SOURCE</th>
</tr>
</thead>
</table>

15. (3 pts) What is the name of the exception that will be raise at runtime?
   In no exception will be raised, write NO ERROR.

```python
def hmm( value, min = 100 ):
    ''' ... '''
    min = min = min
    diff += abs( min - value )
    return diff
```
16. (8 pts) When you want to use a function written by someone else, what must you read? (circle only one letter)

a) the `def` line (the first line of the function definition)  
b) the docstring  
c) the body of the function  
d) the call of the function  
e) all of these  
f) a) and b) only  
g) a), b) and c) only  
h) none of these

17. (5 pts) What is returned when this function is called? (circle only one letter)

```python
def hmm( ):
    pass
```

a) Nil  
b) None  
c) Null  
d) nothing will be returned because there is no `return`  
e) nothing will be returned because this won't even compile  
f) nothing will be returned because `NoReturnValueError` will be raised at run time

18. (10 pts)
In a game, there are 20 possible prizes. The winning user's prize has already been chosen. Now it's time to print a line on the screen telling the winning user which prize they won.

Which **problem form** is controlling the output?

If you can't answer which of the problem forms, for fewer points you may answer by stating which python statement should be used to control the screen display.

**ANNOUNCED DURING THE TEST:** "NOT EVERYONE WINS"

19. (8 pts) Which of the calls below will **NOT** work? (circle the letters of ALL correct answers)

```python
def gimmeMaybe( usuallyIsTen = 10, seldomTheSameValue = 10 ):
    ''' does something with parameters '''
    # code to do something with parameters
```

a) `gimmeMaybe( )`  
b) `gimmeMaybe( 7 )`  
c) `gimmeMaybe( , 7 )`  
d) `gimmeMaybe( 7, 7 )`  
e) `gimmeMaybe( 10 )`  
f) `gimmeMaybe( , 10 )`  
g) `gimmeMaybe( 10, 10 )`  
h) `gimmeMaybe( 7, 7, 7 )`
20. PARTS ONE, TWO and THREE (15 pts together, 5 pts each) PART FOUR is at the bottom
   Draw arrows to show which values the parameter p refers
   to after lines 5 and 7 and which value the local variable x refers
   to after lines 8. after each line of the code below is executed.
   If you need to show more numbers or change the values in RAM, do so.
   The main call is in PART FOUR, below – all this page is the same one program.

This code does compile and run correctly.

```python
def callThee( ):
    ''' … '''
0.     print( 'THEE' )
1.     x = 5
2.     p = callMe( x )
3.     print( x )
4.     return p

def callMe( p ):
    ''' … '''
6.     print( 'ME' )
7.     p = 3
8.     x = p + 2
9.     return 4
```

For example:

This arrow shows where
x refers to after line 1.

Write the values that will be in RAM
.
.
.
.
.
6
5
.
.
.
.
.
.

PART ONE (5 pts)
Draw the arrow showing
where p refers after line 5.

PART TWO (5 pts)
Draw the arrow showing
where p refers after line 7.

PART THREE (5pts)
Draw the arrow showing
where x refers after line 8.

PART FOUR 8 pts)

What is the output when main is called?
The functions in Q20 are part of this program.

```python
def main():
    print( callThee( ) )
main()
```

write your output here
21. (15 pts) There is a built-in function you have never used before named oblatt.

Here is a SnakePit session showing the help for that function.

```
IDLE 2.7.1  >>> help( oblatt )
Help on built-in function oblatt in module __builtin__:
oblatt(...)  
oblatt(object, object, object=1) -> str

The first two parameters must be strings, third must be int
Returns the str made of the first and second parameters concatenated together and repeated the third parameter number of times
E.g.: "Hi", "Lo", 3 yields the string: "HiLoHiLoHiLo" (3 "HiLo''s)

Obviously you cannot define this function because it is built in.
Your job is to complete drawFigureWithNames by writing the needed parameters and python statements that do what the docstring says it will do.

Even though it's not the best, the only functions you can call oblatt and other built-in functions.
You do not need to show any testing for your function.
Do NOT write a main function (obviously) or a complete program (obviously).
ONLY complete the definition for drawFigureWithNames (don't forget the parameters!)
```

def drawFigureWithNames(                           ):
    ''' Given two names, draws a pattern on the screen
        made of the names jammed together in a nice way.
        Here is an example of the output when the names are "John" and "Mary"
        There are always these four lines and always each name these 8 times:
        JohnMaryJohnMaryJohnMary
           -MaryJohn-
           -JohnMary-
        MaryJohnMaryJohnMaryJohn
        The two middle lines start one before where the 1st pair ends.
        Assume both parameters are strs that are names.
    '''