1. What is the output?
x=3.45
print("A{0:.3f}B".format(x))
a
b
c
d
e none of the above

2. What is the output?
h="happy"
s="sad"
print("{0:<6} {1:>10}".format(h,s))
a
b
c
d
e none of the above

3. What is the output?
if 4>10:
    print("cat", end="")
else:
    print("lion", end="")
print("panther")
a. cat
b. lion
c. catpanther
d. lionpanther
e. none of the above
4. What is the output?

```python
a=45
b=36
if a==45:
    print("fish", end="")
elif b <= 36:
    print("whale", end="")
else:
    print("eel", end="")
```

a. fish  b. whale  c. eel  d. fishwhale  e. none of the above

5. What is the output?

```python
if "BABY" == "baby":
    print("small")
else:
    print("large")
```

a. small  b. large  c. smalllarge  d. none of the above

Questions 6 through 8 refer to the following functions:

```python
def f(t):
    ans = t*2+1
    return ans

def g(s):
    print(s*3)
    return 100

def h(s):
    b=f(s)
    return b+10
```

6. What is the output?

```python
x=f(5)
print("x=", x)
```

a. x= x  
b. x= 11  
c. x=t*2+1  
d. none of the above
7. What is the output?

```python
k="M"
y = g(k)
print("k=", k)
```

- a. MMM
- b. k = 100
- c. k = MMM
- d. k = 100
- e. none of the above

8. What is the output?

```python
b=h(10)
print("b=", b)
```

- a. b = 21
- b. b = 20
- c. b = 31
- d. 21
- e. none of the above

Problems 9 to 11 refer to this file and this code:

Here is the file numbers.txt:

```
10 20
30 40
```

Code:

```python
infile=open("numbers.txt", "r")
big=infile.readlines()
small = big[0]
x=small.split()
```

9. What is the value of `big`?

- a. "10 20
\n30 40"
- b. [10, 20, 30, 40]
- c. ["10 20", "30 40"]
- d. ["10 20\n", "30 40"]
- e. none of the above

10. What is the value of `small`?

- a. 1
- b. "1"
- c. "10 20"
- d. "10 20\n"
- e. none of the above
11. What is the value of x?
   a. 1
   b. [10, 20]
   c. ["10", "20"]
   d. ["10", "20\n"]
   e. none of the above

12. x="abcdefg"
    What is the value of x[2:5]?
   a. "bcde"   b. "bcd"   c. "cdef"   d. "cde"   e. none of the above

13. x="abcdefgh"
    What is the value of x[4:]?
   a. "efgh"   b. "defgh"   c. "abcd"   d. "defg"   e. none of the above

14. y= "cdefg"
    What is the value of y[-1]?
   a. "b"   b. "g"   c."f"   d."defg"   e. none of the above

15. z="10,20,30"
    What is the value of z.split(,"") ?
   a. "10 20 30"   b.["10" "20" "30"]   c. ["10", "20", "30"]   d.102030   e. none of the above

16. x= "I am Sam."
    What is the value of len(x)?
   a. 1   b. 9   c. 6   d. 7   e. none of the above

17. y=[4, 3, 5]
    What is the value of len(y)?
   a. 1   b.2   c. 3   d. 4   e. none of the above

18. What is chr(ord('A')) ?
   a. "A"   b. 65   c. I don't know, I didn't memorize the Unicode table.

19. What is the output?
   for x in range(ord("a"), ord("f")):
       print(chr(x), end="", )
   a. a, b, c, d, e, f,   b. a, b, c, d, e,   c. a, b, c, d, e   d. none of the above

20. What is the output?
   for x in "abc"
       print(chr(ord(x)+1), end='-')
   a. a-b-c-d-   b. b-c-d-   c. a+1-b+1-c+1-   d. none of the above
True or False
21. A string always contains a single line of text.
T      F

22. Python lists are mutable, but strings are not.
T      F

23. The add method can be used to add an item to the end of a list.
T      F

24. Accessing a single character out of a string is called:
   a. slicing       b. concatenation   c. assignment   d. indexing

25. Which of the following is the same as s[0:-1]?
   a. s[-1]            b. s[: ]             c. s[ : len(s)-1]   d. s[0: len(s)]

26. Which of the following cannot be use to turn a string of digits into a number?
   a. int              b. float             c. str              d. eval

27. Which string method turns all letters of a string to upper case?
   a. capitalize       b. title             c. toUpper         d. upper

28. What is the output?
   for k in range(5):
       sum=0
       sum=sum+k
   print(k)
   a. 10          b. 4          c. 5          d. 15          e. none of the above

True/False
29. Every Python function must have a parameter.  T      F
30. Every Python function returns some value.         T      F
31. One reason for writing functions is to make programs faster.  T      F
32. All functions are called by main.                T      F
33. A function can send information back to a program with a(n)
   a. return             b. print             c. assignment   d. send
34. Which of the following is not a reason to use functions?
   a. to reduce code duplication
   b. to make a program more readable
   c. to make a program more self-documenting
   d. to demonstrate intellectual superiority

35. A function can modify the value of an actual parameter only if it's
   a. mutable   b. a list   c. an int   d. a variable

36. Consider the function
   ```python
def cube(x):
    answer=x*x*x
    return answer
   ```
   Consider this code that uses this function:
   ```python
answer=4
result = cube(3)
print(answer, result)
   ```
   What is the output?
   a. 27 27   b. 4 27   c. none of the above?

37. What is the output?
   ```python
x=2
y=40
if x>5:
    print("red", end="")
elif y<20:
    print("green", end="")
   ```
   a. red   b. green   c. there is no output

38. An expression that evaluates to either true or false is called
   a. operation   b. Boolean   c. simple   d. compound

39. In Python, the body of a decision is indicated by
   a. indentation   b. parentheses   c. curly braces   d. a colon

40. True or False
   a) 4>5 or 2<3   b) 10<20 or 3<4   c) 10<20 and 3<4
   d) not ( 1>0)
   a) T   b) T   c) T   d) T

41. Consider the file
   ```
num=txt
30 40
   ```
```python
myin = open("nums.txt", "r")
first = myin.readline()
for line in myin:
    print(line, end=', ')
```

What is the output?

a) 30, 40, 20, 10
b) 30 40, 20 10
c) 20 10,
d) none of the above

42. aList is a list of string versions of numbers. For example it could ["1", "3"] or it could be ["2.4", "1.2", "44.6", "23.4", "300"]. Write a function
```python
def sumEntries(aList):
    pass
```
that will add all the numerical values for the entries of the list. It should print nothing, but it should return the sum.

43. reportScore(name) is a function that returns a score on a test, when you give it a student's name.
getCutoff(grade) is a function that reports the lowest score that will earn the grade of grade. For example, getCutoff('A') might be 90.

Write a program that uses these two functions. It should ask a student her name and tell her (print) both her score and her grade, well labeled. Assume the possible grades are A, B, C, D, and F. DO NOT WRITE THE FUNCTIONS reportScore or GetCutoff.

44. What is the output?
```python
for i in ["a", "b"]:
    for t in range(2):
        print(i + str(t) +" ", end="")
print()
```