Simulated Exam Python 1

1. What is a programming language?
   - ☐ Language spoken in everyday life.
   - ☐ Language created by humans to write a program.
   - ☐ Language used by the secret service
   - ☐ Language that organizes work.

2. Which action is **not** part of a computer algorithm:
   - ☐ input and output
   - ☐ Mathematical calculation
   - ☐ Conditional execution and repetition
   - ☐ Reflexion and criticism

3. How does Python perform arithmetic operations: what is the result of the following expression?
   2+(6+2)/(2**2)=

   × 2.5 × 4 × 6 × 5

4. What is **not** a Python value type**?**

   × *Int* and *float* Pay × List × *Boolean* values and *None* × *String*

5. What is the type of the following variable (x):

```
x='3'
print(type(x))
```

   × Int × float × str × boolean

6. What is the value of x in the following program:

```
x=3
y=4
if x==y:
    print(x)
```

   × 3 × 4 × 1 × 7

7. Let a=50 and b=25. What are the values for the following logical operations:
   a>40 and b>40
   a>0 and b>0
   a>0 or b<0
   b<0 or b<0
   - ☐ True, True, False, False
   - ☐ False, True, False, True
   - ☐ False, True, True, False
   - ☐ False, False, True, True

8. Why is the function "float()" used in the following program?

```
import math
r=float(input('radius'))
V=(3/4)*math.pi*r**3
print(V)
```

- ☐ Because Variable *r* is an integer.
- ☐ Because we want to write a text with the result.
- ☐ Because *Input()* returns the input as a String value.
- ☐ Because we multiply variable *r* with a *float* multiply value.

9. Which output does the program produce?

```
a=2
a+=1
print(a)
```

× 2 × 1 × 3 × 4

10. What output does the program produce?

```
x = ''
for i in range(10 - 6):
    x += str(i)
print(x)
```

× '0123' × '4444' × 'iiii' × 6

11. What output does the program produce?

```
i=1
while i<3:
    i+=1
print(i)
```

× 1 × 2 × 3 × 4

12. What are the results of the following expressions?

- 1+1.0
- 1+'1'
- '1'+'1'

× 2, 11, 2 × 2.0, 'SyntaxError', '11' × 1, 1, '2' × 2, 'SyntaxError', 2

13. Why is there a 'SyntaxError' with the program:

```
def countdown(n)
while n>0:
print(n)
n=n-1
print('Bang!')
```

- ☐ because the program has no indentation.
- ☐ because the print function is missing a parenthesis.
- ☐ because variable n is a string.
- ☐ because *while* is in an infinite loop.

14. What is the output of the following program?

```
my_list = ['x', 'y', 'z']
my_list.append('a')
empty_string = ''
for i in my_list:
        empty_string += i
print(empty_string)
```

× 'xyz' × 'a' × 'xyza' × ' '

15. Which output does the program produce?

```
print(type('a') == str and 3.5 > 1)
```

× False × SyntaxError × 'a' and 3.5 ×True

16. The following program should return the lexicographically largest letter of a word to be passed. Why does the following program **not** work?

```
word = int(input('Which word should be checked? '))
def greatestChar(long_word):
        maxChar = 2.5
        for c in long_word:
        if c > maxChar:
        c = maxChar
        else:
        break
 greatest_char = greatestChar(word)
```

- ☐ because the maxChar is a float value.
- ☐ because the for loop is dead.
- ☐ because the variable *word* should be a string value.
- ☐ because the variable *word* should be a float value.

17. What does the following function do?

```
word=input('Word')
word=word.lower()
i=0
for s in word:
    if s in [‘a’,’e’,’i’,’o’,’u’]:
        i+=1
print(i)
```

- ☐ The program says how long the word is.
- ☐ The program says how many letters are lowercase.
- ☐ The program says how many vowels the word has.
- ☐ The program pushes the letters into the list.

18. Given the following List=[['python', 'refugeeks','Monty'],[25,63,41]], which element does the statement **List[1][1]** express?

× 'Python' × 'refugeeks' × 25 ×63

19. In which case the *for* loop does **not** work:

- ☐ `for w in word:`
- ☐ `for w in ['a','b','c']:`
- ☐ `for w in 4:`
- ☐ `for w in range(4):`

20. Which result does the function express with input b equals **20**.

```
def bmi_interpreter(b):

    if b < 17.5:
        print(b,' : underweight')
    elif 18.5<b<25:
        print(b,' : normal')
    else:
        print(b, ' : overweight')
```

- ☐ 20 : normal
- ☐ 20: underweight
- ☐ 78.4 : overweight
- ☐ 5.1 : underweight