1 What will be the output of the following code snippet?

```
def func(a, b): return b if a == 0 else func(b % a, a)
print(func(30, 75))
```
 a) 10
 b) 20
c) 15
d) 0

Answer: c) 15

The initial call to the function is '**func(30, 75)**'. Here's how the recursion unfolds:
a = 30, **b = 75**: Since **a** is not equal to 0, the function calls itself with (**75 % 30, 30)**, which becomes **func(15, 30)**.
a = 15, **b = 30**: Again, **a** is not equal to 0, so the function calls itself with (**30 % 15, 15)**, resulting in **func(0, 15).**
a = 0, **b = 15**: This time, since **a** is equal to **0**, the function returns the value of **b**, which is **15**.
Therefore, the output of the code snippet is **15.**

2) numbers = (4, 7, 19, 2, 89, 45, 72, 22)
sorted_numbers = sorted(numbers)
even = lambda a: a % 2 == 0
even_numbers = filter(even, sorted_numbers)
print(type(even_numbers))
a) Int
b) Filter
c) List
d) Tuple

Answer: b)Filter

3) As what datatype are the *args stored, when passed into
a) Tuple
b) List
c) Dictionary
 d) none
Answer:Tuple

4) set1 = {14, 3, 55} set2 = {82, 49, 62} set3={99,22,17} print(len(set1 + set2 + set3))
 a) 105
b) 270
c) 0
d) Error
Answer:Error

Here concatenate sets using the + operator, but sets in Python do not support concatenation in the same way as lists or strings. To combine sets, we can use the union method or the | operator.

5.What keyword is used in Python to raise exceptions?
a) raise
b) try
c) goto
d) except
Answer:Raise

In Python, the raise keyword is used to explicitly raise exceptions. We can use it in conjunction with the raise statement to signal that an error or exceptional situation has occurred in our code.

6.Which of the following modules need to be imported to handle date time computations in Python?
a) timedate
b) date
c) datetime
d) time
Answer:datetime

The **datetime** module is part of the Python standard library and does not require additional installation.

7) What will be the output of the following code snippet?
print(4**3 + (7 + 5)**(1 + 1))
a) 248
 b) 169
c) 208
d) 233
Answer:208

8) Which of the following functions converts date to corresponding time in Python?
a) strptime
b) strftime
c) both a) and b)
d) None
Answer:b) strftime

9) The python tuple is _____ in nature.
a) mutable
b)immutable
c)unchangeable
d) none
Answer: b) immutable

In Python, a tuple is an immutable data type, meaning its elements cannot be changed or modified after the tuple is created

10) The __ is a built-in function that returns a range object that consists series of integer numbers, which we can iterate using a for loop.
A. range()
B. set()
C. dictionary{}
D. None of the mentioned above
 Answer: A.range()

11) Amongst which of the following is a function which does not have any name?
A. Del function
B. Show function
C. Lambda function
D. None of the mentioned above
Answer: C.Lambda function

12)The module Pickle is used to __.
A. Serializing Python object structure
B. De-serializing Python object structure
C. Both A and B
D. None of the mentioned above
Answer: C. Both A and B

13)Among which of the following is / are the method of convert Python objects for writing data in a binary file?
A. set() method
B. dump() method
C. load() method
D. None of the mentioned above
Answer: B.dump() method

14)Amongst which of the following is / are the method used to unpickling data from a binary file?
A. load()
B. set() method
C. dump() method
D. None of the mentioned above
Answer:A.load()

15)A text file contains only textual information consisting of ___.
A. Alphabets
B. Numbers
C. Special symbols
D. All of the mentioned above
Answer:D. All of the mentioned above

16) Which Python code could replace the ellipsis (...) below to get the following output? (Select all that apply.)
captains = {
"Enterprise": "Picard",
"Voyager": "Janeway",
"Defiant": "Sisko",
}
Enterprise Picard,
Voyager Janeway
Defiant Sisko

a) for ship, captain in captains.items(): print(ship, captain)
b) for ship in captains: print(ship, captains[ship])
c) for ship in captains: print(ship, captains)
d) both a and b
 Answer: d) both a and b

17)Which of the following lines of code will create an empty dictionary named captains?
 a) captains = {dict}
b) type(captains)
c) captains.dict()
d) captains = {}
Answer: d) captains = {}

18) Now you have your empty dictionary named captains.
 It's time to add some data!
Specifically, you want to add the key-value pairs
"Enterprise": "Picard", "Voyager": "Janeway", and "Defiant": "Sisko".
Which of the following code snippets will successfully add these key-value pairs
to the existing captains dictionary?

a) captains{"Enterprise" = "Picard"}
captains{"Voyager" = "Janeway"}
captains{"Defiant" = "Sisko"}
b) captains["Enterprise"] = "Picard"
captains["Voyager"] = "Janeway"
captains["Defiant"] = "Sisko"
c) captains = { "Enterprise": "Picard", "Voyager": "Janeway",
  "Defiant": "Sisko", }
d) None of the above

Answer: b) captains["Enterprise"] = "Picard"
captains["Voyager"] = "Janeway"
captains["Defiant"] = "Sisko"

19) You're really building out the Federation Starfleet now!
Here's what you have:
captains = { "Enterprise": "Picard", "Voyager": "Janeway", "Defiant": "Sisko",
"Discovery": "unknown", }
Now, say you want to display the ship and captain names contained in the
dictionary, but you also want to provide some additional context.
How could you do it?
a) for item in captains.items():
print(f"The [ship] is captained by [captain].")
b) for ship, captain in captains.items():
print(f"The {ship} is captained by {captain}.")
c) for captain, ship in captains.items():
print(f"The {ship} is captained by {captain}.")
d) All are correct

Answer: b) for ship, captain in captains.items():
print(f"The {ship} is captained by {captain}.")

20) You've created a dictionary, added data, checked for the existence of keys, and iterated over it with a for loop.
 Now you're ready to delete a key from this dictionary:
captains = { "Enterprise": "Picard", "Voyager": "Janeway", "Defiant": "Sisko", "Discovery": "unknown", }
What statement will remove the entry for the key "Discovery"?
 a) del captains
b) captains.remove()
c) del captains["Discovery"]
d) captains["Discovery"].pop()

Answer: b) captains.remove()