

## Latihan Persiapan UAS

1. Buatlah sebuah fungsi rekursif python yang akan me-list semua permutasi dari sebuah kata. Permutasi merupakan pengaturan dari huruf-huruf yang menyusun sebuah kata.

Contoh: Permutasi dari "xyz" ada 6 yaitu :

"xyz", "xzy", "yxz", "yzx", "zxy", "zyx"

2. Buatlah list yang berisi 100 angka kelipatan 2 terkecil dengan menggunakan comprehension list

Contoh:

[2, 4, 6, 8, 10, ....., 200]

3. Perhatikan potongan kode berikut

```
var_x = [10,20,30,40,50]
def misteri(var_a):
    var_a[2] = 100
    var_a[4], var_a [1] = var_a [1], var_a [4]
    var_a = 5
misteri(var_x)
var x [3] = 15
print(var_x)
```

Output dari potongan kode di atas adalah

4. Buatlah sebuah fungsi yang menghitung banyaknya kemunculan angka dalam sebuah string

Contoh:

```
>>> HitungAngka ("Pada tahun 2015, terjadi peningkatan yang signifikan yaitu sebesar 28 persen dari tahun sebelumnya menjadi 896.377 miliar.")
```

12

```
>>> HitungAngka ("Sandra sedang pusing 7 keliling")
```

1

```
>>> HitungAngka ("Tujuh kali tujuh sama dengan empat sembilan")
```

0

5. Carilah output dari code berikut ini

```
def main():
    line = input()
    wordList = line.split()
    hasil = 1
    for element in wordList:
        try:
            val = int(element)
            hasil *= val
        except ValueError:
            print ('Value Error')
            break
    finally:
        print (element)
    print(hasil)

main()
```

Bila user memasukkan input berikut ini maka code akan memberikan output?

15 tahun 5 bulan 4 hari sudah rautmu murung mematung tanpa ujung

6. Perhatikan kode berikut ini

```
class Animal(object):
    def __init__(self, name = "Animal", talk = ""):
        print ("Animal __init__()")
        self.name = name
        self.talk = talk
    def getName(self):
        return self.name
    def getTalk(self):
        return self.talk
class Cat(Animal):
    def __init__(self, name, talk = "meow"):
        super().__init__()
        print ("Cat __init__()")
        self.name = name
        self.talk = talk
    def getName(self):
        return self.name
    def getTalk(self):
        return self.talk
class Dog(Animal):
    def __init__(self, name, talk = "Bark"):
        super().__init__()
        print ("Dog __init__()")
        self.name = name
        self.talk = talk
    def getName(self):
        return self.name
    def getTalk(self):
        return self.talk

a = Animal(talk = "Animal")
c = Cat(name="Cat")
d = Dog(name="Dog")

print (c.getName())
print (d.getTalk())
```

Apakah output dari program tersebut?

7. Apakah output dari code berikut ini

```
class X(object):
    def __init__(self, param1 = 10):
        print("calling init in class X")
        self.param1 = param1
    def method(self):
        print("calling method in X")
        self.param1 //= 2
class Y(object):
    def __init__(self, param2 = 20):
        print("calling init in class Y")
        self.param2 = param2
    def method(self):
        print("calling method in Y")
        self.param2 //= 5

class Z(X, Y):
    def __init__(self, param2 = 100, param1 = 50):
        print("calling init in class Z")
        X.__init__(self)
        Y.__init__(self)
    def method(self):
        X.method(self)
        Y.method(self)
        print("calling method in Z")
        self.param2 *= 5

z = Z()
z.method()
print(z.param1)
print(z.param2)
```

8. Apakah output dari potongan code di bawah ini?

```
d = { 'a': 1, 'b': 2, 'c': 3 }
print (d.items())
```