COMPUTER PROGRAMMING I
Introduction To Python

BIL2205

Dokuz Eylul University, Faculty of Science,
Department of Statistics

Data Types in Python

□ *Python Collections:*

□ **List** is a collection which is ***ordered, indexed*** and ***changeable***. Allows duplicate members.

□ **Tuple** is a collection which is ***ordered , indexed*** and ***unchangeable***. Allows duplicate members.

□ **Set** is a collection which is ***unordered*** and ***unindexed***. No duplicate members.

□ **Dictionary** is a collection which is ***unordered, changeable*** and ***indexed***. No duplicate members.

Lists in Python

- **A list** is an *ordered* sequence of values and may contain values of any data type.
- In Python lists may be heterogeneous (may contain items of different data types).
- Ex:
  - `scores = [78, 93, 80, 68, 100, 94, 85]`
  - `colors = [‘red’, ‘green’, ‘blue’]`
  - `mixed = [‘purple’, 100, 90.5]`

Lists in Python

- Lists
  - Python lists are written with square brackets.
- Ex:

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
print (sınıf)
```

Lists in Python

- □ List Length
  - ▣ `len(the_list)`
- □ Ex:

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
print (len(sınıf))
```

Lists in Python

- List Indexing
- Ex:

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
print (sınıf[1])
```

- Indexing begins with 0
  - The first item is the 0th item.

Lists in Python

- □ List Indexing

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
```

- □ Ex: Print the first item in the list

```
print (sınıf[0])
```

- □ Ex: Print the last item in the list

```
print (sınıf[len(sınıf)-1])
print (sınıf[-1])
```

Lists in Python

□ List Indexing

□Ex: Print the whole list

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
for i in range(len(sınıf)):
    print (sınıf[i])

for eleman in sınıf:
    print (eleman)
```

Lists in Python

- □ List Indexing – Slices
- □ Ex:

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
print (sınıf[0:2])
print (sınıf[:2])
print (sınıf[2:5])
print (sınıf[2:])
print (sınıf[:-1])
print (sınıf[-2:])
```

Lists in Python

- List Indexing
- Ex: Change the value of an element

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
sınıf[1] = "Hüseyin"
print (sınıf)
```

Lists in Python

- □ List Indexing
- □ Ex: Change the value of an element

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
sınıf[1] = ["Hüseyin","Mehmet","Leyla"]
print (sınıf)

sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
sınıf[1:1] = ["Hüseyin","Mehmet","Leyla"]
print (sınıf)
```

Lists in Python

- □ List – Adding an item to a list
- □ `the_list.append(new_item)`

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
sınıf.append("Jale")
print (sınıf)
```

Lists in Python

- □ List – Adding multiple items to a list
- □ the_list.**append**(new_item1, new_item2)

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
sınıf.append("Hüseyin","Mehmet","Leyla")

sınıf.append(["Hüseyin","Mehmet","Leyla"])
print (sınıf)
```

Lists in Python

□ List – Adding multiple items to a list

□ the_list.**extend**(new_list)

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
sınıf.extend(["Hüseyin","Mehmet","Leyla"])
print (sınıf)
```

Lists in Python

□ List – Inserting an item to a specific index in a list

□ `the_list.insert(index, new_item)`

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
sınıf.insert(1, "Osman")
print (sınıf)

sınıf.insert(3, "Hamdi")
print (sınıf)
sınıf.insert(345, "Nalan") # ???
print (sınıf)
```

Lists in Python

- □ List – Removing an item in a list
- □ the_list.**remove**(item_to_remove)

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
sınıf.remove("Veli")
print (sınıf)

sınıf.extend(["Ali", "Fatma"])
sınıf.remove("Ali") # which "Ali"?
print (sınıf)
```

Lists in Python

- □ List – Removing an item in a list
- □ `the_list.pop(index_of_item_to_remove)`

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
sınıf.pop(2)
print (sınıf)

sınıf.pop() # ???
print (sınıf)
```

Lists in Python

- List – Removing a slice of items in a list
- `the_list[a:b] = [ ]`

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
sınıf[1:4]= []
print (sınıf)
```

Lists in Python

- Empty a list
- `the_list.clear()`

```
sınıf = ["Ali","Veli","Ayşe","Fatma","Hasan"]
sınıf.clear()
print (sınıf)
print (len(sınıf))
```

Lists in Python

- Creating an empty list

```
sınıf = []
print (sınıf)
print (len(sınıf))
```

- Creating a list of n items

```
sınıf = [0] * 25
print (sınıf)
print (len(sınıf))
```

Lists in Python

- Joining lists:

```
kızlar = ["Ayşe", "Fatma", "Hayriye", "Leyla"]
erkekler = ["Ahmet", "Mehmet", "Hüseyin"]
sınıf = kızlar + erkekler
print (sınıf)
print (len(sınıf))
```

- List membership test:

```
print ("Fatma" in sınıf)
print ("George" in sınıf)
```

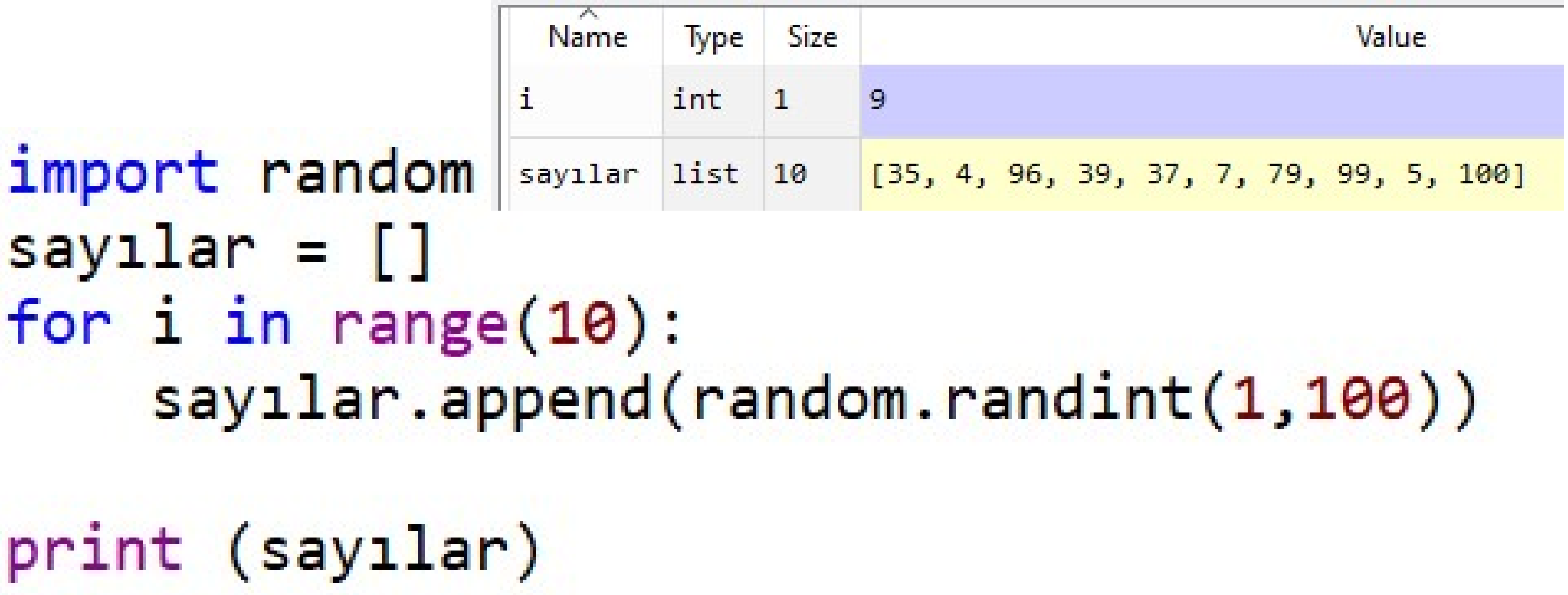
Lists in Python

□ **Lists in lists:**

```
kızlar = ["Ayşe", "Fatma", "Hayriye", "Leyla"]
erkekler = ["Ahmet", "Mehmet", "Hüseyin"]
sınıf = [kızlar, erkekler]
print (sınıf)
print (len(sınıf))
print (sınıf[0])
print (sınıf[1])
print (sınıf[1][2])
```

Lists in Python – Exercises

- Create a list of 10 random integer (between 1 and 100)



Name	Type	Size	Value
i	int	1	9
sayılar	list	10	[35, 4, 96, 39, 37, 7, 79, 99, 5, 100]

```
import random
sayılar = []
for i in range(10):
    sayılar.append(random.randint(1,100))

print (sayılar)
```

Lists in Python – Exercises

- Generate 10 random integer (between 1 and 100), store them in a list.
- Find the average of the numbers in the list.
- Find the minimum value in the list.
- Find the maximum value in the list.
- Find the range of the numbers in the list.
- Find the standart deviation of the numbers in the list.
- Find the number of integers greater than 50 in the list.
- Find the median of the numbers in the list.

Lists in Python – Exercises

- □ Ask the user to enter integers, -1 to stop.
- □ Find the average of the numbers entered.
- □ Find the minimum value entered.
- □ Find the maximum value entered.
- □ Find the range of the numbers entered.
- □ Find the standart deviation of the numbers entered.
- □ Find the number of odd and even numbers entered.

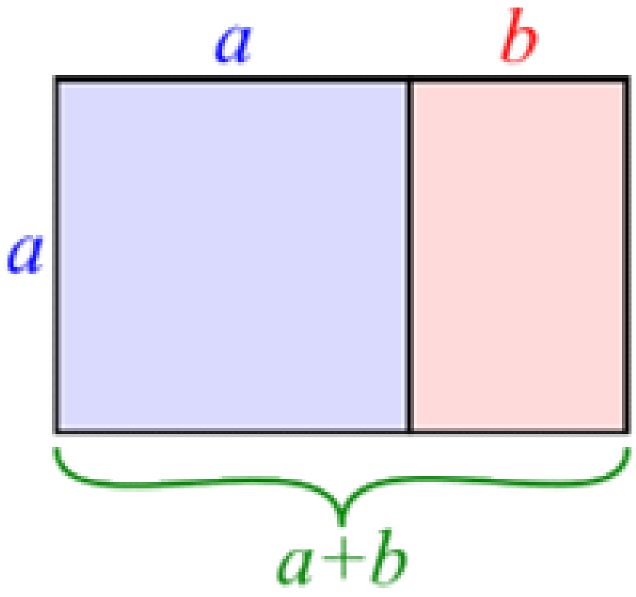
Exercise

- Fibonacci Series :
  - First two terms are 0 and 1
  - The next term is the sum of previous two terms.
- Question:
  - Find the first 24 terms of Fibonacci Series
- `0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946, 17711, 28657`

Golden Ratio / Golden Section

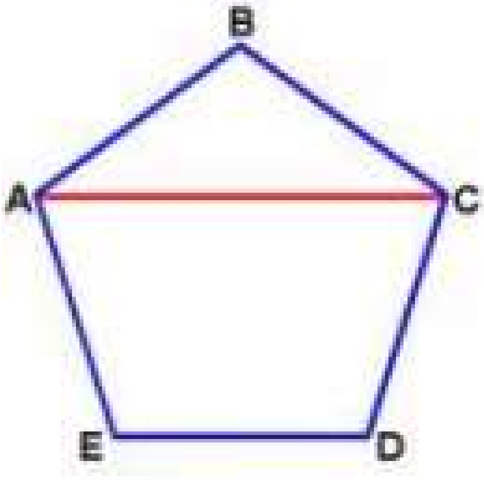
- □Divine Proportion
- □Phidias (490-430 BC)
- □Plato (427-347 BC)
- □Euclid (325-265 BC)
- □Fibonacci (1170-1250)

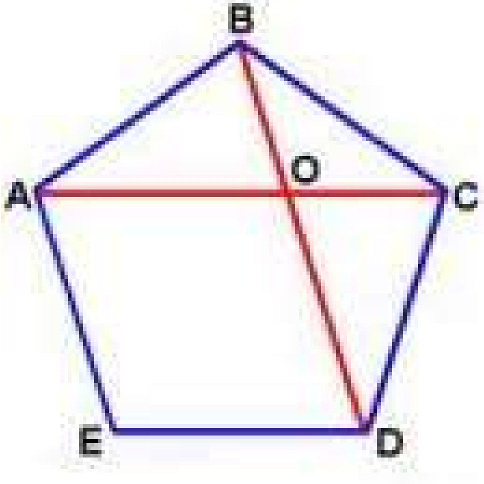
$$\varphi = \frac{1+\sqrt{5}}{2} = 1.6180339887\ldots.$$

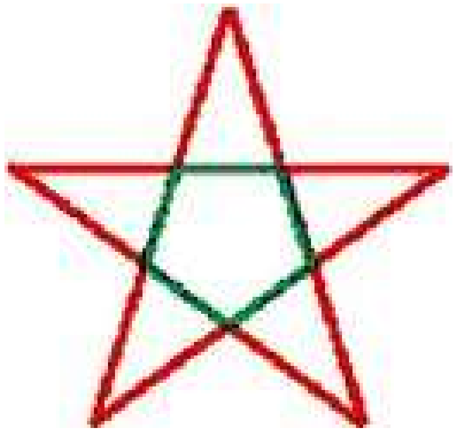


Golden Ratio / Golden Section

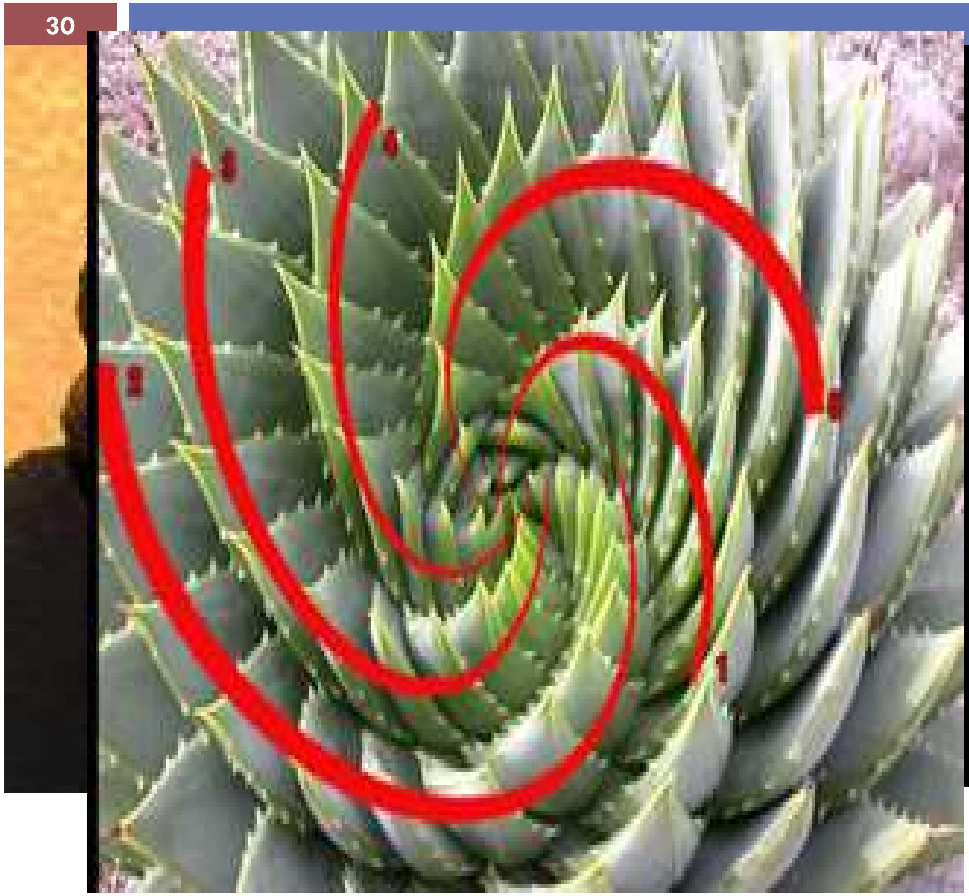
Golden Ratio - Geometry

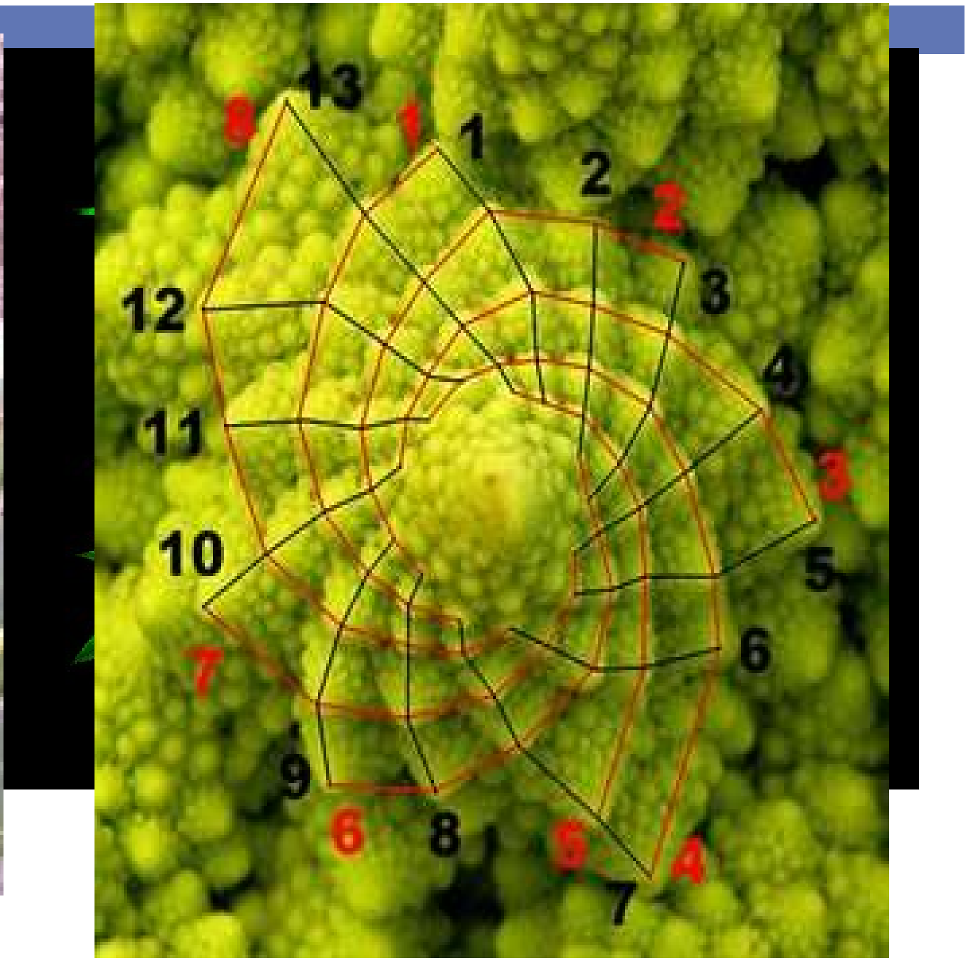


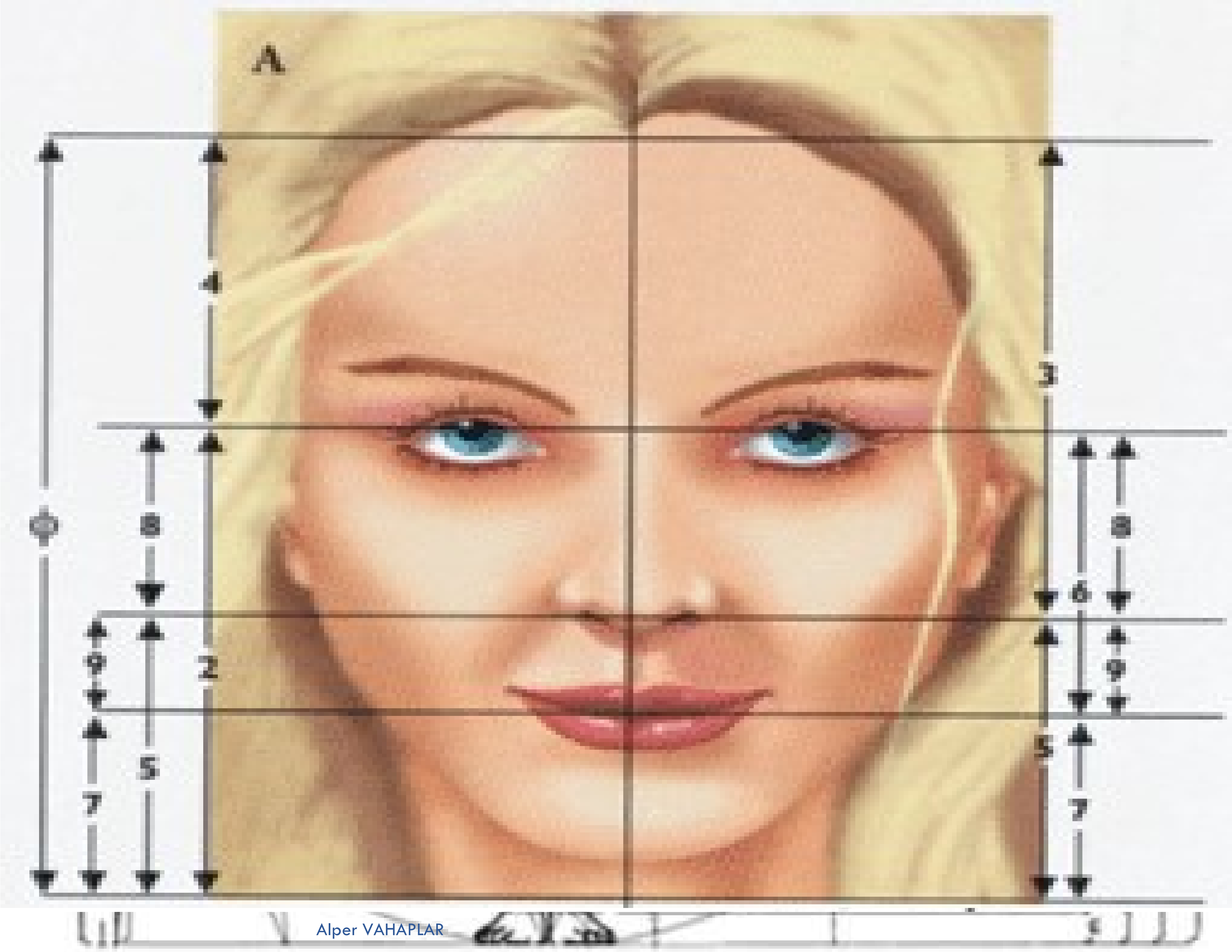


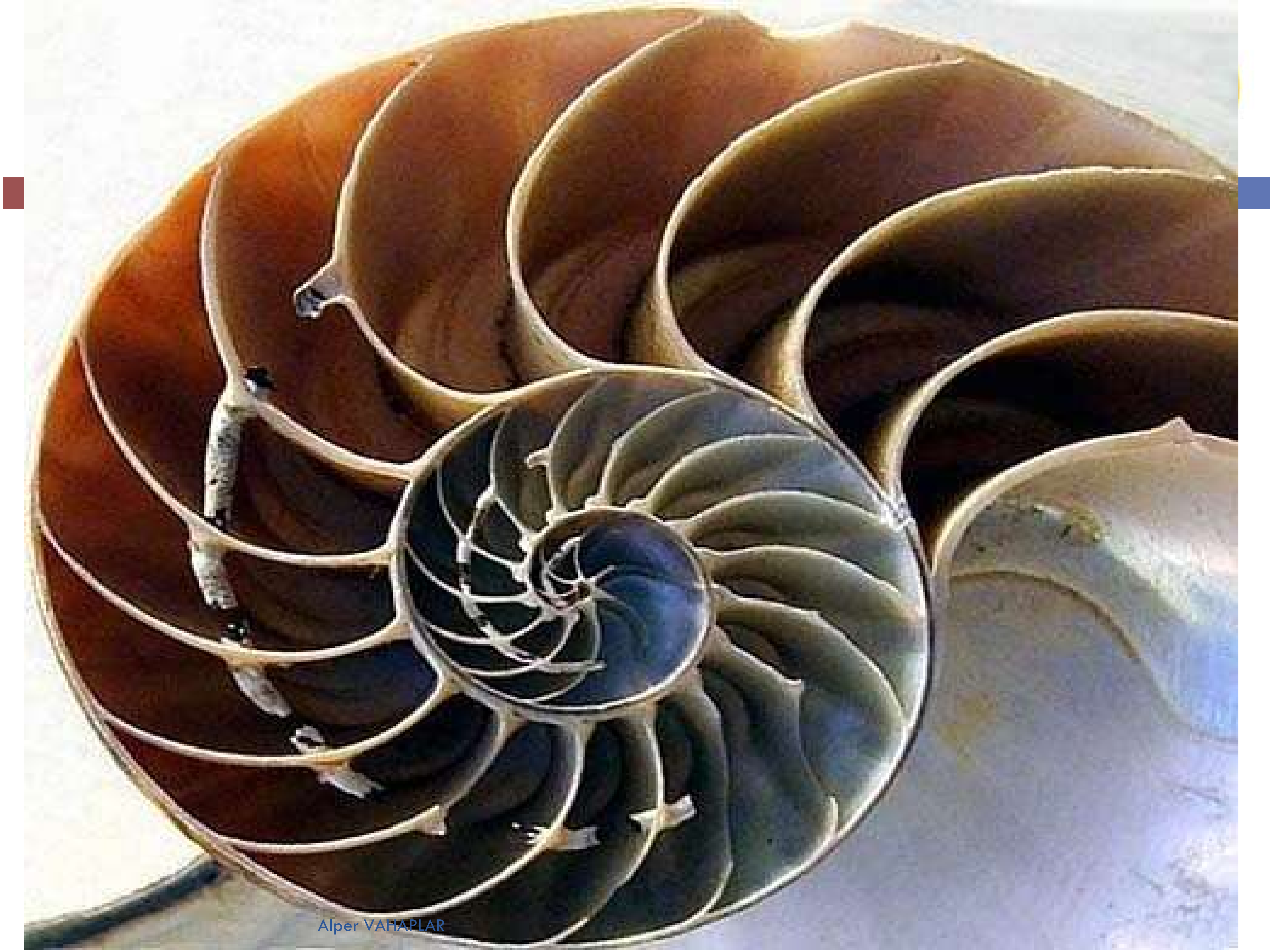


Golden Ration - Life






A
4
3
8
8
6
9
9
2
5
5
7
7



S.Ozan Dertli

Exercise

- Fibonacci Series:
  - First two terms are 0 and 1
  - The next term is the sum of previous two terms.
- Question:
  - Find the first 24 terms of Fibonacci Series
- `0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946, 17711, 28657`
- Find the ratio of fib(n)/fib(n-1) for the first 24 terms.