Logo

Description automatically generated

**RIGA TECHNICAL UNIVERSITY**

**Faculty of Computer Science and Information Technology Institute of Applied Computer Systems**

**“ 9 Assignments ”**

TASK OF INDIVIDUAL WORK

**"DIP108-Algorithmization Practice”**

**Lect, Padmaraj Nidagundi**

**Accomplished:** Azhniuk Sofiia **Student Card No:** 221ADB207

2022./2023.

**Index**

[**Assignment**](#_heading=h.jr0x6fbv669p) **3**

[1. Develop four algorithms to improve math grades in school](#_heading=h.tbme7lcok9l) 3

[2. Write an algorithm to find all roots of a quadratic equation ax2+bx+c=0](#_heading=h.9ogsp8z8e7jn) 3

[3. Write an algorithm to convert temperature from Celsius to Fahrenheit](#_heading=h.z7a2ngucfgfg) 4

[4. Write an algorithm to find the largest among three different numbers entered by the user.](#_heading=h.cqrprkz94exw) 4

[5. Write an algorithm "Facebook friend suggest" feature](#_heading=h.vmei82gqned1) 6

[6. Write an algorithm "YouTube video suggests" feature](#_heading=h.crvltvuc3ze7) 6

[7. Write an algorithm "biometric face recognition " feature](#_heading=h.ueo7wkq4evte) 6

[8. Write an algorithm for making ice cream ( use your own ingredients )](#_heading=h.o6x90emy0xrl) 7

[9. Answer the following questions](#_heading=h.4s5zu5s5vyvy) 7

[A) What is an algorithm?](#_heading=h.o3pyv4tphig3) 7

[B) What is the Complexity of the Algorithm?](#_heading=h.tvvb545qypmk) 7

[Explain 1) Time complexity 2) Space complexity.](#_heading=h.6ijadj3l5ro9) 7

[C) What are the Asymptotic Notations?](#_heading=h.e37eano4ss8l) 7

[D) What are the algorithm characteristics?](#_heading=h.t4dah6tenbg) 7

[E) Explain Big-O complexities and their learning benefits for programming](#_heading=h.sk8tx97x3cgj) 7

[Bibliography](#_heading=h.3znysh7) **8**

# 

# Assignment

## Develop four algorithms to improve math grades in school

* Understand the Problem Bad grades
* Develop an Algorithm: Find additional courses and books ⇒ Contact teacher for consultation ⇒ Create good planning, dedicate more hours ⇒ Execute that regularly until getting the final result.

## 

## Write an algorithm to find all roots of a quadratic equation ax2+bx+c=0

**Algorithm:**

Step1: Ask for “a, b, c” numbers.

Step2: Find Discriminant

Step3: Check the value of the discriminant (<0, ==0, >0)

Step4: Calculate the value of x1 and x2 if possible.

Step5: Stop

**Program** (Python):

import math

print ("Write a b and c for the equation ax2+bx+c=0:")

a = int(input("a = "))

b = int(input("b = "))

c = int(input("c = "))

D=float(b\*\*2 - 4\*a\*c)

if D<0 :

print("No roots")

elif D==0:

print("Ok")

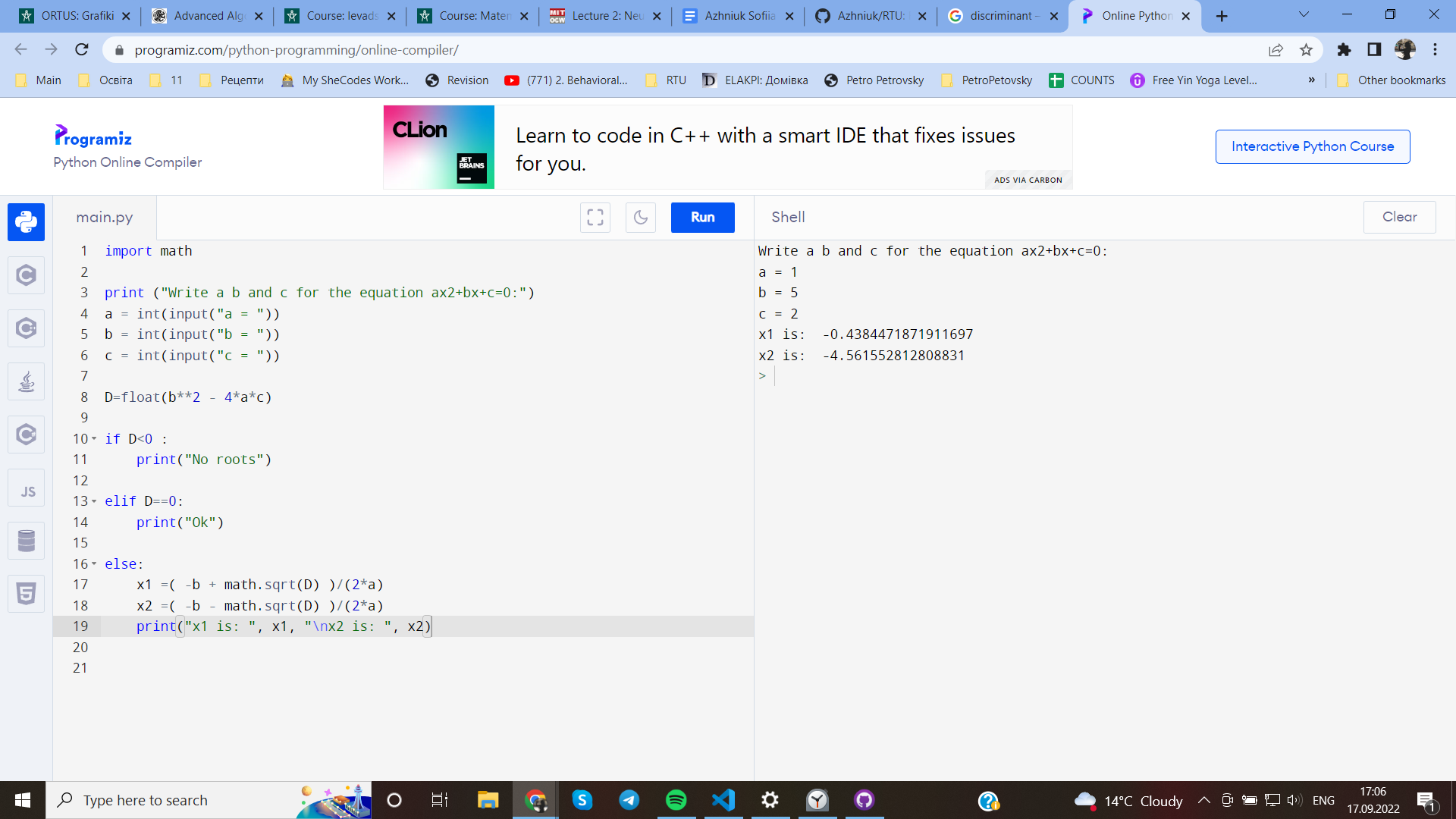
else:

x1 =( -b + math.sqrt(D) )/(2\*a)

x2 =( -b - math.sqrt(D) )/(2\*a)

print("x1 is: ", x1, "\nx2 is: ", x2)

**Output:**



## Write an algorithm to convert temperature from Celsius to Fahrenheit

**Algorithm:**

Step1: Ask for Celsius Temperature

Step2: Multiply the temperature by 1.8 and add 32

Step3: Print the result

Step4: Stop

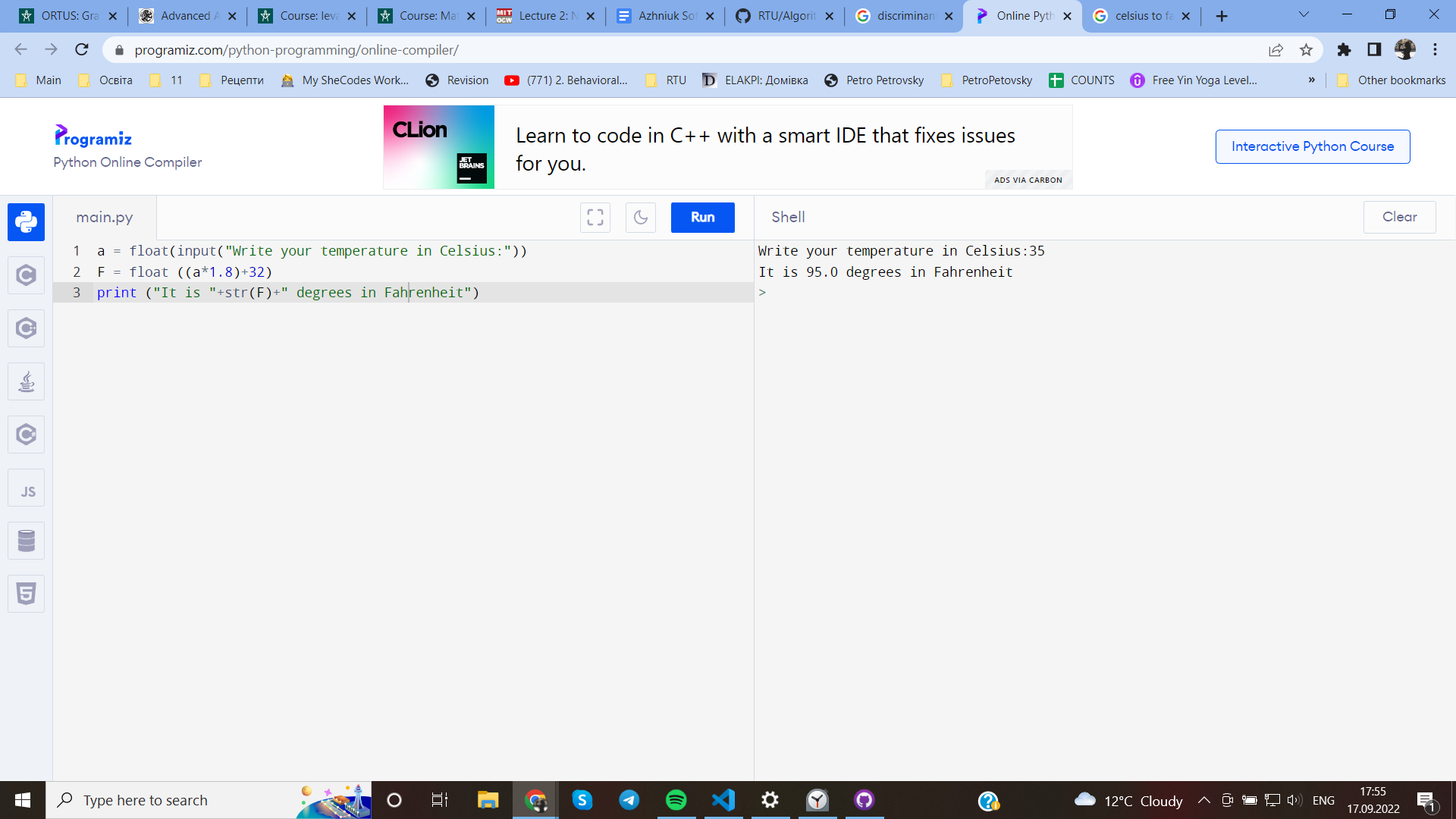
**Program:**

a = float(input("Write your temperature in Celsius:"))

F = float ((a\*1.8)+32)

print ("It is "+str(F)+" degrees in Farenheit")

**Output:**



## Write an algorithm to find the largest among three different numbers entered by the user.

**Algorithm:**

Step1: Ask for three numbers

Step2: Create the array with that numbers

Step3: Sort the array

Step4: Return the last number

Step5: Stop.

**Program:**

def sort(array\_name):

for i in range(len(array\_name)):

swapped = False

for j in range(0, len(array\_name) - i - 1):

if array[j] > array[j + 1]:

a = array[j]

array[j] = array[j+1]

array[j+1] = a

swapped = True

if not swapped:

break

array = []

number\_of\_items = int(input("How many numbers? "))

print("Write number+enter:")

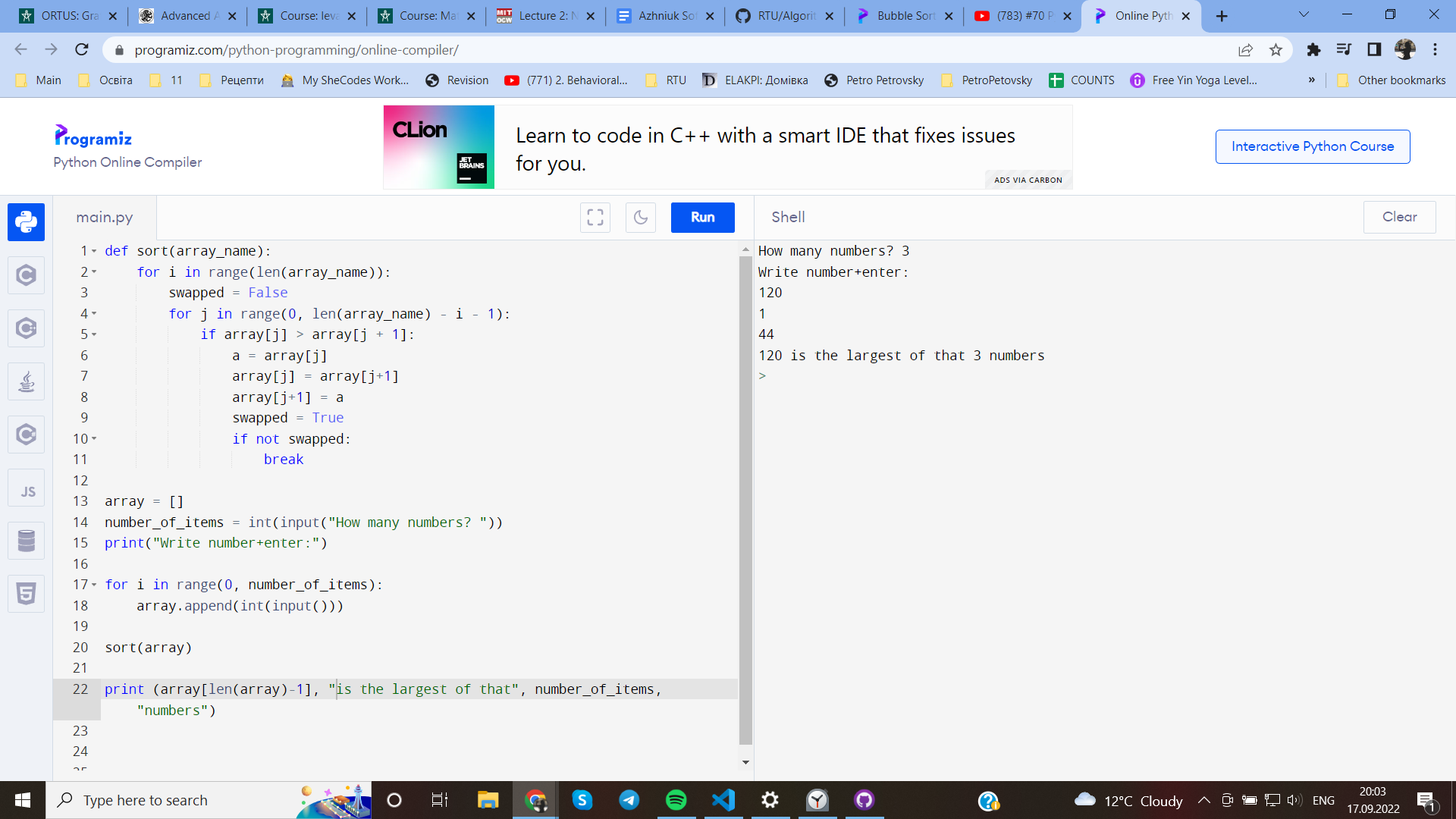
for i in range(0, number\_of\_items):

array.append(int(input()))

sort(array)

print (array[len(array)-1], "is the largest of that", number\_of\_items, "numbers")

**Output:**



## Write an algorithm "Facebook friend suggest" feature

Step 1: Start

Step 2: Look for friends of friends and suggest them

Step 3: Collect friends with the same interests as the user(for example knitting, football, music, etc)

Step 4: Collect people from the same city (or ever country)

Step 5: Collect people from your contacts

Step 6: Collect accounts with content, that user likes( again for example knitting, football, music, etc)

Step 7: Suggest them based on collected data

Step 8: Stop

## Write an algorithm "YouTube video suggests" feature

Step 1: Start

Step 2: Collect the information about the user

Step 3: Process the information and:

Step 4: Find a video based on previous likes, views etc and suggest it

Step 5: Collect information about user`s interests and suggest the similar videos

Step 6: Suggest videos that are popular in user`s city/country

Step 7: Mostly form suggestions with the subscriptions of a user

Step 8: Show the most recent videos

Step 9: Show different kinds of video (translations, shorts and usual videos)

Step 10: Stop

## Write an algorithm "biometric face recognition " feature

Step 1: Start

Step 2: First, the system captures an image of the user's face using a camera or other imaging device.

Step 3: Program finds an ovals, wich could be the face(s)

Step 4: The system then extracts facial features from the processed image, such as the distance between the eyes, the shape of the nose and chin, and the overall geometry of the face, the oval of the face.

Step 5: These extracted features are then compared to a database of known faces, using a machine learning algorithm to determine the closest match.

Step 6: If a match is found with a high degree of confidence, the system identifies the user as the person associated with the known face in the database.

Step 7: If no match is found, or the confidence level is too low, the system can prompt for additional information or authentication.

Step 8: Stop

## Write an algorithm for making ice cream ( use your own ingredients )

Step 1: Start

Step 2: Go to the shop

Step 3: Buy milk and special powdered mixture chocolate “Lody”

Step 4: Go home

Step 5: Get a bowl, milk and powder

Step 6: Mix all together with a mixer for 10 minutes

Step 7: Get forms for an ice-cream

Step 8: Put the mixture into the form

Step 9: Close the form in the freeze

Step 10: Wait for 24 hours

Step 11: Stop

## Answer the following questions

## A) What is an algorithm?

An algorithm is the sequence of planned actions/steps we need to solve a problem.

**What is the need for an algorithm?**

To solve some problems. For better performance, the economy of the time.

## B) What is the Complexity of the Algorithm?

Complexity of Algorithm is the computations of the amount of time and space required by an algorithm for an input of size

## Explain 1) Time complexity 2) Space complexity.

1. Time complexity is the time required towards the execution of that algorithm
2. Space complexity is the memory space required towards the execution of that algorithm

## C) What are the Asymptotic Notations?

Asymptotic Notations are the notations to describe the running time of an algorithm (how much time an algorithm takes with a given input)

## D) What are the algorithm characteristics?

**Unambiguity, fineness, effectiveness, and language independence**

**Unambiguous** – clear algorithm. All steps should be obvious and lead to only one meaning

**Finiteness** – algorithms must terminate after a finite number of steps

**Effectiveness** - all those steps that are required to get to output must be feasible with the available resources.

**Language independence** – an algorithm should have step-by-step directions, which should be independent of any programming language

## E) Explain Big-O complexities and their learning benefits for programming

In computer science, the Big-O notation is used to describe the performance or complexity of an algorithm. It provides a way to express how the number of operations required by an algorithm grows as the size of the input grows.

Big-O is the one of the asymptotic notations. It is used for the worst case running time.

For programming it is good to measure the scalability of an algorithm. It specifies the maximum number of operations taken by an algorithm to give the output according to the amount of data on which the program must work. For execution the algorithm, Big O uses a huge amount of the performance, which leads to long waiting times

# Bibliography

**1.** **Lect. Padmaraj Nidagundi’s Class Presentation Slides and Study Materials (available on Ortus)**

**2. The Big-O explanations (**[**https://www.linkedin.com/pulse/what-big-o-notation-why-important-computer-md-mafujul-islam?trk=articles\_directory**](https://www.linkedin.com/pulse/what-big-o-notation-why-important-computer-md-mafujul-islam?trk=articles_directory)**)**

**3. Advanced Algorithms and Data Structures (WS 2021/22) (https://www.cosy.sbg.ac.at/~held/teaching/aads/aads\_print.pdf)**