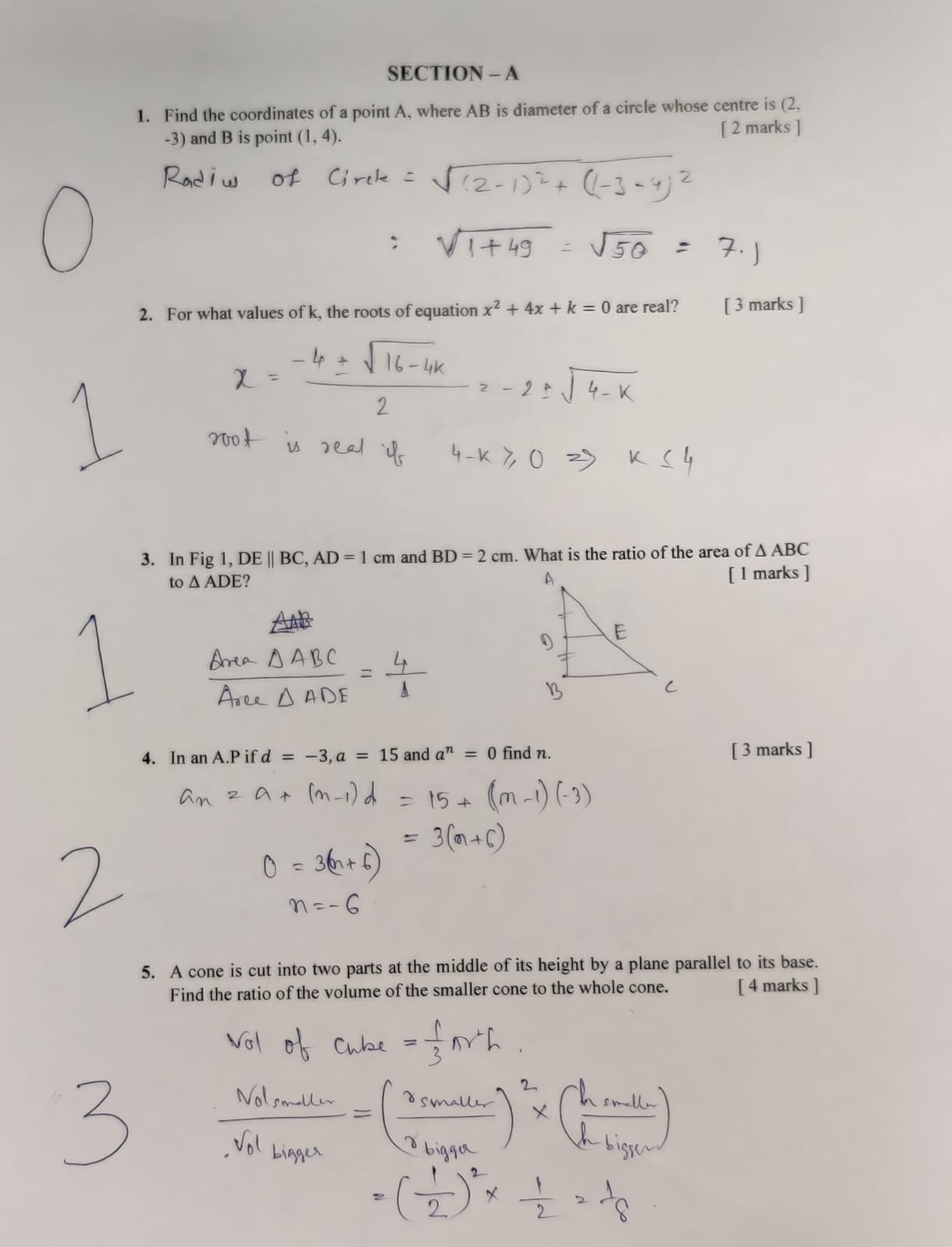
**Problem Statement**

To detect marks from a given answer sheet and display the topic of a subject where he needs to improve .

Answer Sheet :



Syllabus:



Requirements

!sudo apt install tesseract-ocr

!pip install pytesseract

!pip install sentence-transformers

Imports:

import cv2

import torch

from PIL import Image

import numpy as np

from google.colab.patches import cv2\_imshow

import pandas as pd

import pytesseract

from keras.datasets import mnist

from keras.layers import Dense, Flatten

from keras.layers.convolutional import Conv2D

from keras.models import Sequential ,load\_model

from keras.utils import to\_categorical

import glob

Sentence Transformers:

from sentence\_transformers import SentenceTransformer, util

sent\_model = SentenceTransformer('all-MiniLM-L6-v2')

df = pd.read\_csv('/content/java\_syllabus.csv')

contents = [content for content in df['content'].values]

topics = [topic for topic in df['topic'].values]

# display(topics)

It cleans the topics and removes English punctuations and repeated words

Model of Number Recognition

This model is trained by MNIST data and the model is saved after training and this trained model is used for number detection

model = load\_model("/content/mnist\_model.h5")

Loading of Answer Sheet:

img\_rgb = cv2.imread('/content/desc2.jpeg')

img\_gray = cv2.cvtColor(img\_rgb, cv2.COLOR\_BGR2GRAY)

wid, hgt = img\_gray.shape[::-1]

cv2\_imshow(img\_rgb)

ret, thresh1 = cv2.threshold(img\_gray, 0, 255, cv2.THRESH\_OTSU |

                                          cv2.THRESH\_BINARY\_INV)

rect\_kernel = cv2.getStructuringElement(cv2.MORPH\_RECT, (12, 5))

dilation = cv2.dilate(thresh1, rect\_kernel, iterations = 2)

cv2\_imshow(dilation)

contours, hierarchy = cv2.findContours(dilation, cv2.RETR\_EXTERNAL,

                                            cv2.CHAIN\_APPROX\_NONE)

im2 = img\_rgb.copy()

Separating Obtained Marks and Max Marks

list = []

c=0

max\_marks =[]

cropped = ""

que = []

for cnt in contours:

    x, y, w, h = cv2.boundingRect(cnt)

    # Draw the bounding box on the text area

    cropped = im2[y:y + h, x:x + w]

    cropped = img\_gray[y:y + h, x:x + w]

    if(x< (wid\*0.1)):

      c+=1

      list.append([x,y,w,h])

    if(x> (wid\*0.7)):

      text = pytesseract.image\_to\_string(cropped)

      max\_marks.append([text,x,y,w,h])

    else :

      que.append([x,y,w,h])

final=[]

for i in max\_marks:

  x="".join(ch for ch in i[0] if ch.isalnum())

  if x.isalnum():

    if x.\_\_contains\_\_("mark"):

    # num = [int(ch) for ch in x.split() if ch.isdigit()]

      final.append([int(x[0]), i[1] , i[2],i[3],i[4]])

# in this block we are coreleating max marks coordinates with question cooordinates and taking final output as

# coordinates of a question and max marks for a question as list

f\_que = []

for m in final:

  comp = m[2]

  for q in que:

    if (abs(comp-q[1])) in range(0,40) and ((q[2]\*q[3])>500) and (q[0]<(wid \* 0.20)):

      cv2\_imshow(im2[q[1]:q[1] + q[3], q[0]:q[0] + q[2]])

      f\_que.append([q,m[0]])

for i in range(len(f\_que)-1):

  if (f\_que[i][0][1]-f\_que[i+1][0][1]) in range(0,70):

    f\_que.pop(i)

#In this block we coreleate coordinates of question and marks attained

# we can have question coordinates , marks obtained coordinates , max marks in one list

ans = []

for l in list:

  comp = l[1]

  for m in f\_que:

    if(l[1]-m[0][1] > 0 and l[1]-m[0][1]<comp):

      ans.append([l,m[0],m[1]])  # in ans list we have coordinates of marks obtained, question , max marks

      comp= l[1]-m[0][1]

c=0

for i in range(len(ans)):

  mark\_img = im2[ans[i][0][1]:ans[i][0][1] + ans[i][0][3], ans[i][0][0]:ans[i][0][0] + ans[i][0][2]]

  mark\_img = cv2.resize(mark\_img, (28,28))

  # cv2\_imshow(mark\_img)

  cv2.imwrite("/content/obtained\_marks/mark"+str(c)+".jpeg",mark\_img)

  c+=1

Function for Number Recognition

def predict(path):

    image = cv2.imread(str(path))

    grey = cv2.cvtColor(image.copy(), cv2.COLOR\_BGR2GRAY)

    grey =cv2.bitwise\_not(grey)

    ret, thresh = cv2.threshold(grey.copy(), 100, 255, cv2.THRESH\_BINARY)

    # cv2\_imshow(thresh)

    resized\_digit = cv2.resize(thresh, (18,18))

    digit = np.pad(resized\_digit, ((5,5),(5,5)), "constant", constant\_values=0)

    prediction = model.predict(digit.reshape(1, 28, 28, 1))

    # plt.imshow(digit.reshape(28, 28), cmap="gray")

    # plt.show()

    return np.argmax(prediction)

Printing Output:

contents\_embeddings = sent\_model.encode(contents)    # Embedding contents data

for i in range(len(ans)):

  ques\_img = im2[ans[i][1][1]:ans[i][1][1] + ans[i][1][3], ans[i][1][0]:ans[i][1][0] + ans[i][1][2]]  #taking question from its coordinates

  cv2\_imshow(ques\_img)

  quest\_txt = pytesseract.image\_to\_string(ques\_img)     #converting image to text

  question\_embedding = sent\_model.encode(quest\_txt)    ## Embedding question text

  scores = util.dot\_score(question\_embedding, contents\_embeddings)

  max\_index = torch.argmax(scores)

  # print(quest\_txt)

  obt\_marks\_img = im2[ans[i][0][1]:ans[i][0][1] + ans[i][0][3], ans[i][0][0]:ans[i][0][0] + ans[i][0][2]]

  # cv2\_imshow(obt\_marks\_img)

  cv2.imwrite("/content/mark.jpeg",obt\_marks\_img)

  mark=predict("/content/mark.jpeg")

  # if((mark/ans[i][2])<=0.9):

  print('Question: ', quest\_txt)

  print('Topic: ', topics[max\_index])

  print("Max Marks",ans[i][2],"\n")

  print("Obtained Marks",mark,"\n")

  cv2\_imshow(obt\_marks\_img)

  print('----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------')

Output:



1/1 [==============================] - 0s 31ms/step

Question: 1. Find the coordinates of a point A, where AB is diameter of a circle whose centre is (2.

Topic: CIRCLES

Max Marks 2

Obtained Marks 0





1/1 [==============================] - 0s 22ms/step

Question: 2. For what values of k, the roots of equation x? + 4x + k = 0 are real?

Topic: QUADRATIC EQUATION

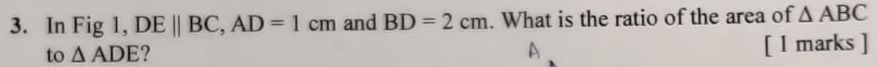
Max Marks 3

Obtained Marks 1

A black line on a white surface

Description automatically generated with low confidence

--------------------------------------------------------------------------------



1/1 [==============================] - 0s 22ms/step

Question: 3. In Fig 1, DE || BC, AD = 1 cm and BD = 2 cm. What is the ratio of the area of A ABC

to A ADE? ‘ [ I marks ]

Topic: TRIANGLES

Max Marks 1

Obtained Marks 1



--------------------------------------------------------------------------------



1/1 [==============================] - 0s 24ms/step

Question: 4. InanA.Pifd = —3,a = 15 anda" = 0 findn.

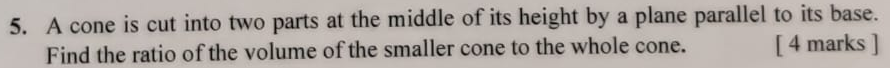
Topic: ARITHMETIC PROGRESSION

Max Marks 3

Obtained Marks 2

A picture containing hook

Description automatically generated with medium confidence



1/1 [==============================] - 0s 395ms/step

Question: 5. Acone is cut into two parts at the middle of its height by a plane parallel to its base.

Find the ratio of the volume of the smaller cone to the whole cone. [4 marks }

Topic: SURFACE AREAS AND VOLUMES

Max Marks 4

Obtained Marks 3

A number on a white surface

Description automatically generated with low confidence