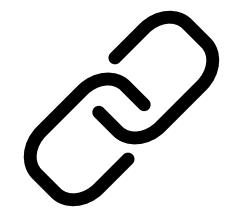


2024

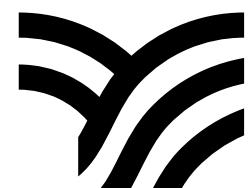
NeuralGrader

Quiz Checking Using LLMs



Web App

zohair.onrender.com



About Us

We are intrepid senior-year undergraduates at NUST, fueled by our fervent passion for the wonders of AI.

Our Team -



Muhammad Anser



Zuha Fatima



Zohair Shakeel

Problem

Assessing handwritten quizzes or assignments, can be a laborious endeavor, fraught with the potential for human error—especially when navigating diverse formats for example Python scripts, biological illustrations, and intricate circuit designs.

A desire arises for a tool that can assess and appraise various elements, identifying errors without necessitating specialized knowledge in any particular domain.

Solution

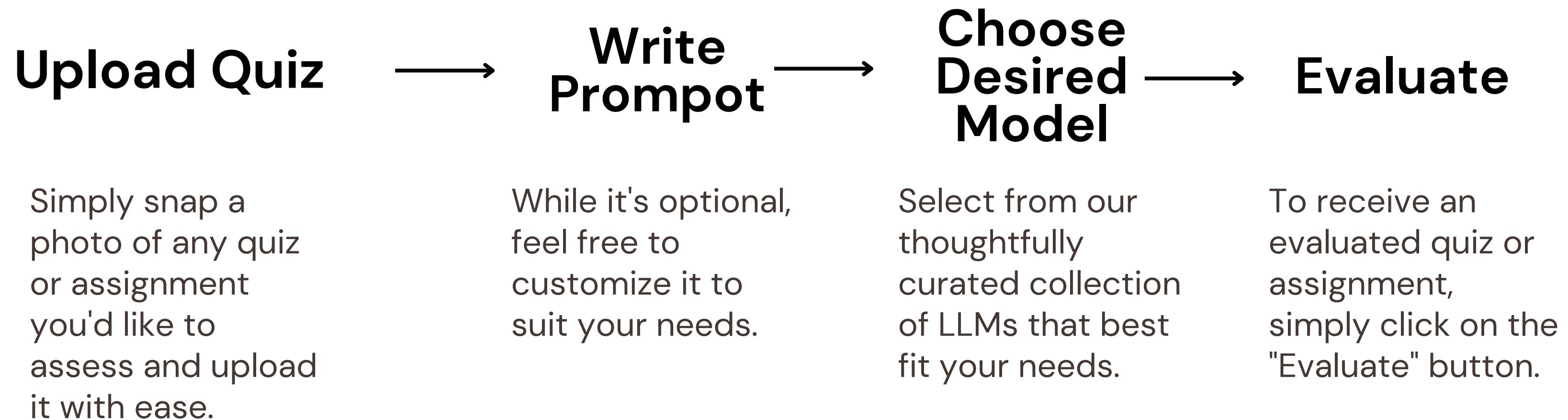
NeuralGrader, a user-friendly web application that automates quiz evaluation by utilizing advanced Large Language Models (LLMs) tailored for text and diagram analysis.

NeuralGrader conjures a remarkable time-saving spell, slashing efforts by over 90 percent. Best of all, it is a free enchantment and a premier alternative.

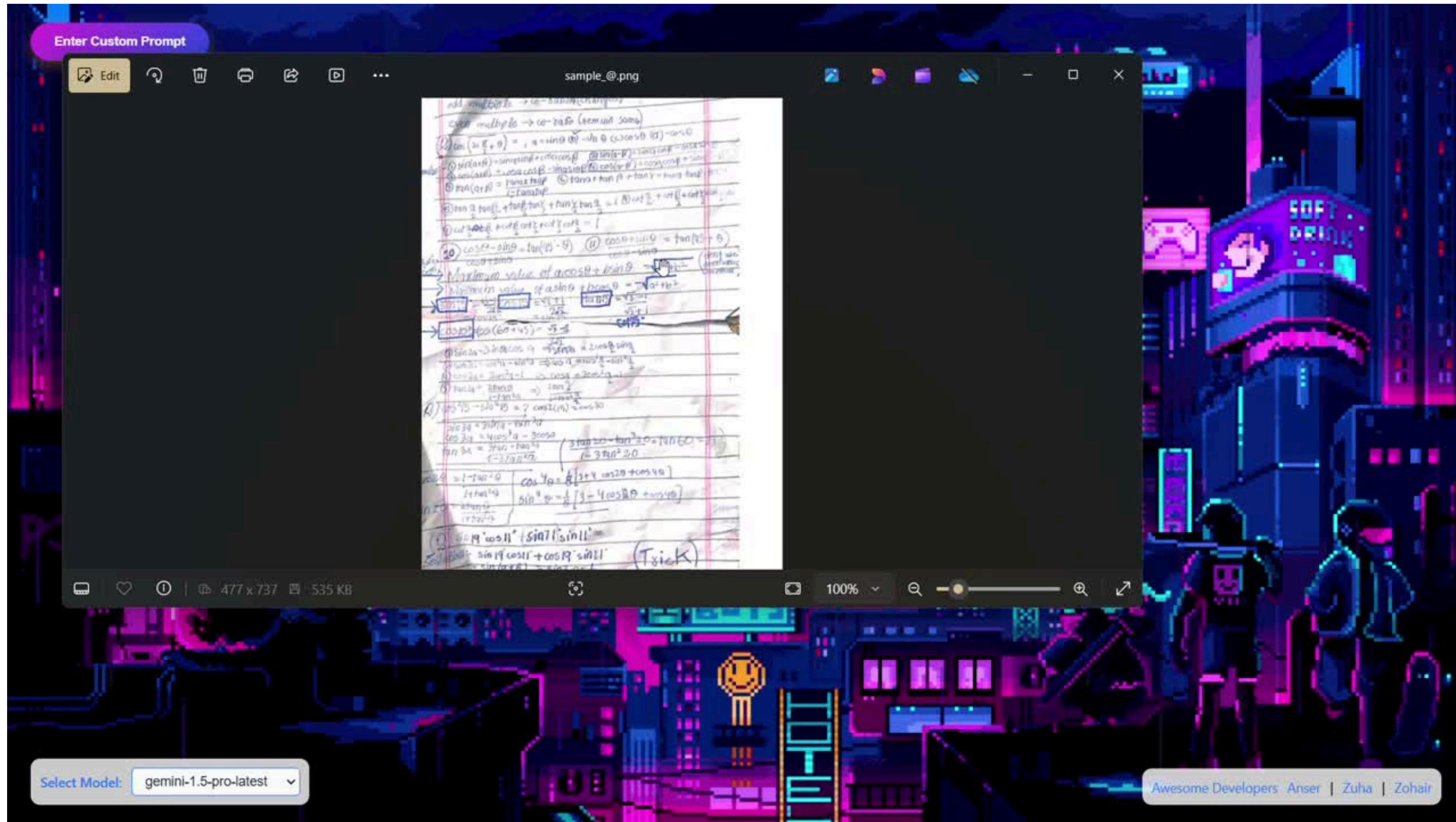
Essential tools / premier offerings

- Frontend: HTML, Flask (UI/UX optimized for ease of use).
- Backend: Google-Gemini-Flash-1.5-Pro via Google Cloud API for OCR.
- LLMs: Integrated thoughtfully handpicked LLMs for diverse quiz types like Python programs, biological diagrams, and circuit logic.

Workflow



Grading a chaotic handwritten quiz in half a minute.



Potential Impacts

Time Efficiency:

- Dramatically diminishes the hours spent on manual grading.

Scalability:

- Perfectly suited for educational realms and virtual learning landscapes.

Accessibility:

- Empowers educators to navigate intricate assessments effortlessly, enriching the learning journey.

No Need for Expertise:

- With NeuralGrader, mastery in a specific domain is no longer a prerequisite.

Tests



Handwritten Python Programs

BASIC PYTHON PROGRAMS

1]. Python Program to print all prime number in given interval.

✓ **Code:**

```
ATUL KUMAR (LINKEDIN).  
NOTES GALLERY (TELEGRAM).  
start = 11  
end = 75  
Print("prime numbers between ", start, "and ",  
      end, "are:")  
For (i in range (start, end + 1)):  
    If i > 1:  
        For j in range (2, i):  
            If (i % j == 0):  
                Break  
        Else  
            Print(i)  
  
Output:  
Prime numbers between 11 and 75 are:  
11 , 13 17 19 23 29 31 37 41 43  
47 59 61 67 71 73
```

2]. Python program to print square of given number:

✓ **Code:**

```
num = int(input ("enter number: "))  
Square = n * n  
print("Square is : ", Square)  
Output:  
enter number : 10  
Square is : 100
```

NeuralGrader

Drag and drop quiz document or click to upload

Evaluate Quiz

*Accuracy: The code has a logical error; the condition `if (i % j == 0)` should be `if (i % j == 0)`, using the modulo operator (%) instead of integer division (/). Also, 19, 29, 37 and 71 are incorrectly marked in the output as prime and 53, 59 are missed out.
*Completeness: The code includes the basic structure for prime number identification, but lacks robustness due to the logical error mentioned above.
*Clarity: Indentation is inconsistent, which is crucial in Python. Variable naming (i, j) could be more descriptive (e.g., `num`, `divisor`).

Score: 4/10

NeuralGrader

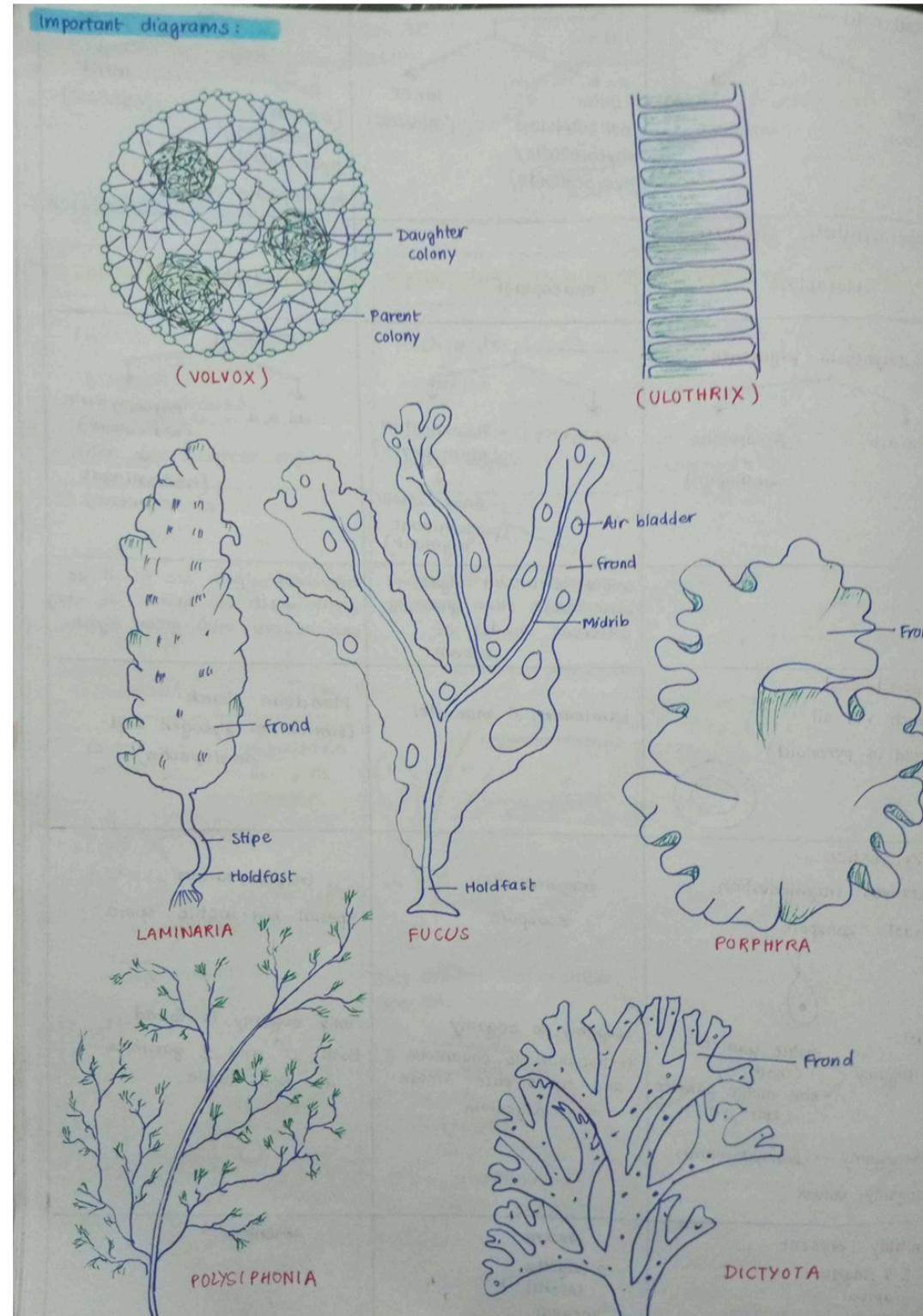
Drag and drop quiz document or click to upload

Evaluate Quiz

*Accuracy: The code calculates the square correctly in logic, but not in code. Variable 'n' is not defined.
*Completeness: The core components are present (input, calculation, output), but with the error mentioned above.
*Clarity: Clear and concise, but could be improved by using a more descriptive variable name instead of 'num' (e.g., `number_to_square`). Also, logic needs to be correct to show the right output.

Score: 6/10

Biological Diagrams



NeuralGrader

Drag and drop quiz document or click to upload

Evaluate Quiz

Quiz Analysis

The diagrams are correctly drawn and labelled. The concept of showing different algae types and their structures is well-executed.

Score: 10/10

Flip Flops

10.4 Level-Triggered and Edge-Triggered Flip-Flops

In a level-triggered flip-flop, the output responds to the data present at the inputs during the time the clock pulse level is HIGH (or LOW). That is, any changes at the input during the time the clock is active (HIGH or LOW) are reflected at the output as per its function table. The clocked R-S flip-flop described in the preceding paragraphs is a level-triggered flip-flop that is active when the clock is HIGH.

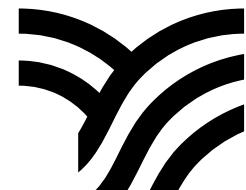
In an edge-triggered flip-flop, the output responds to the data at the inputs only on LOW-to-HIGH or HIGH-to-LOW transition of the clock signal. The flip-flop in the two cases is referred to as positive edge triggered and negative edge triggered respectively. Any changes in the input during the time the clock pulse is HIGH (or LOW) do not have any effect on the output. In the case of an edge-triggered flip-flop, an edge detector circuit transforms the clock input into a very narrow pulse that is a few nanoseconds wide. This narrow pulse coincides with either LOW-to-HIGH or HIGH-to-LOW transition of the clock input, depending upon whether it is a positive edge-triggered flip-flop or a negative edge-triggered flip-flop. This pulse is so narrow that the operation of the flip-flop can be considered to have occurred on the edge itself.

Figure 10.23 shows the clocked R-S flip-flop of Fig. 10.21 with the edge detector block incorporated in the clock circuit. Figures 10.24 (a) and (b) respectively show typical edge detector circuits for positive

The screenshot shows the NeuralGrader platform. At the top, the logo 'NeuralGrader' is displayed in blue with a brain icon. Below the logo is a dashed purple rectangular area with a cloud icon and the text 'Drag and drop quiz document or click to upload'. A purple button labeled 'Evaluate Quiz' is located below this area. A large white text box contains a positive evaluation of the submitted explanation: 'The explanation of level-triggered and edge-triggered flip-flops is accurate and complete. It correctly distinguishes between the two types based on their response to the clock signal (level vs. edge). The description of the edge-detector circuit and its role in creating a narrow pulse is also well-explained. The text is clear, concise, and well-structured.' At the bottom of the text box, the score 'Score: 10/10' is shown. The background of the interface features a blurred image of a city skyline at night.



Thank You!



Questions? Reach Out Here:
ansersohaib.study@gmail.com

