

**TIME : 2hr**

**TOTAL MARK : 50**

## **PART A**

1. **Create a React component called `Counter` that:**
    - Displays a number starting from 0
    - Has buttons to increment, decrement, and reset the counter
    - Ensure the counter never goes below 0  
(10 marks)
  2. **Create a React component `ToggleText` that shows a paragraph of text.**
    - Include a button that toggles the visibility of the text
    - Use conditional rendering  
(5 marks)
  3. **Create a React component `FruitSelector` that:**
    - Has a dropdown with fruits: Apple, Banana, Mango
    - Displays the selected fruit below the dropdown dynamically  
(5 marks)
- 

## **PART B**

1. **Create a React component `TodoList`:**
  - Users can add tasks using an input and a button
  - Display tasks in a list with a "Delete" button for each task
  - Use `useState` to manage the list  
(8 marks)

2. Create a React component **ColorChanger** that:

- Has 3 buttons: Red, Green, Blue
- Clicking a button changes the background color of a `<div>`
- Display the current selected color as text inside the div  
(4 marks)

3. Create a React component **Timer** that:

- Displays a count that increases by 1 every second using `useEffect` and `setInterval`
  - Include a "Stop Timer" button to stop the counter  
(8 marks)
- 

## **PART C**

1. Create a parent component **Parent** and a child component **Child**:

- Pass a string from the parent to the child via props
- The child should display the string inside an `<h3>`  
(3 marks)

2. Create a React form component **UserForm**:

- Fields: Name (text), Age (number), Email (email)
- Display the submitted data below the form after clicking "Submit"
- Validate that Name and Email are not empty  
(3 marks)

3. Create a React component **Accordion**:

- Display 3 sections with headings: "Section 1", "Section 2", "Section 3"
- Only one section can be open at a time; clicking a heading toggles its content
- Use state to manage which section is open  
(4 marks)