**Funções:**

**calculateResult**: Calcula o valor da expressão no display, verificando se o último caractere é um operador (caso seja, ele o ignora) e exibe o resultado no display.

**deleteLastDigit**: Apaga o último dígito do display. Caso o último caractere seja um ponto decimal, permite que ele seja reutilizado.

**writeOnDisplay**: Escreve no display o número ou operador clicado. Se um operador for inserido após outro operador, o último operador será substituído pelo novo. Também controla a quantidade de pontos decimais permitidos (máximo de um ponto por número).

**verifyDecimalPoint**: Verifica se o valor digitado é um ponto decimal.

**verifyOperator**: Verifica se o valor digitado é um operador válido (+, -, \*, ou /).

**calculateArray**: Avalia a expressão matemática no display, separando e processando operações em uma ordem de precedência (soma, subtração, multiplicação e divisão).

**multiplyArray**: Calcula o produto de uma lista de números.

**divideArray**: Calcula a divisão de uma lista de números, dividindo o primeiro pelo restante.

**subtractArray**: Subtrai uma lista de números, subtraindo os valores subsequentes do primeiro valor.

**sumArray**: Calcula a soma de uma lista de números.

Etapas:

### **1. Preparação do HTML e Seleção dos Elementos**

* Certifique-se de ter um campo de exibição para mostrar os resultados.
* Identifique cada botão de operação e número na interface, pois precisará acessá-los individualmente no código.
* Crie IDs únicos para cada botão de número, operador e funções adicionais como limpar e calcular o resultado.

### **2. Organização dos Botões e Eventos**

* Crie um array para armazenar os botões e agrupe-os em categorias: números, operadores e funções auxiliares.
* Adicione um evento de escuta a cada botão do array para capturar o clique e disparar a função correspondente.

### **3. Implementação do Display**

* Implemente uma função para adicionar números e operadores ao display, verificando se um operador substitui o anterior quando necessário.
* Controle a quantidade de pontos decimais permitidos para evitar múltiplos pontos em um único número.

### **4. Cálculo e Exibição do Resultado**

* Crie uma função para realizar o cálculo da expressão matemática. Antes de processar, remova qualquer operador que possa estar no final da expressão.
* Divida a expressão de acordo com os operadores, respeitando a precedência (primeiro multiplicação e divisão, depois soma e subtração).
* Exiba o resultado final no display, garantindo que valores inválidos sejam tratados.

### **5. Funções Auxiliares para Manipulação do Display**

* Implemente uma função para apagar o último dígito no display, considerando que um ponto decimal apagado deve liberar o limite para um novo ponto.
* Adicione uma função para limpar completamente o display e reiniciar o contador de pontos decimais.

### **6. Validação de Operadores e Ponto Decimal**

* Implemente uma função para validar operadores e outra para verificar se o caractere digitado é um ponto decimal.
* A validação dos operadores ajuda a evitar duplicações e a garantir que operadores consecutivos sejam substituídos corretamente.

### **7. Funções para Cálculo das Operações**

* Desenvolva funções separadas para cada operação matemática (soma, subtração, multiplicação e divisão) que aceitem listas de valores.
* Essas funções irão processar a lista para que cada parte da expressão seja resolvida com a precedência correta dos operadores.

### **8. Integração do Cálculo Final**

* Monte uma função que utiliza as funções de soma, subtração, multiplicação e divisão para resolver expressões matemáticas completas.
* Implemente uma estrutura de repetição e uma lógica de divisão das operações para garantir que a expressão seja processada corretamente em cada nível de precedência.

### **9. Testes e Ajustes Finais**

* Teste cada funcionalidade para verificar se os operadores funcionam como esperado e se o display exibe os valores corretos.
* Ajuste o comportamento do display para garantir que, ao pressionar os botões de limpar ou apagar, o resultado seja resetado corretamente.