Prof Dr P Kantek (pkantek@gmail.com) VIVXh91a V: 1.02 CARLOS EDUARDO ANTAO

Máquina de troco

24FRO405 - 1

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, $\ensuremath{\mathrm{CSS}}$ e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- · Forneça a série de moedas existentes no país
- Depois, forneça o valor do troco a ser produzido

```
    O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5
                                                          (Moedas, separadas por vírgulas)
Valor a pagar: 59
 Calcular
Moedas a liberar: 50,5,2,2
                                                                     (Moedas, separadas por vírgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
}
```

Avaliação:

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

71 13 98 50 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76001 - /

Prof Dr P Kantek (pkantek@gmail.com) VIVXh91a V: 1.02 DANILO DEPETRIS SOARES

Máquina de troco

24FRO405 - 2

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, CSS e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- · Forneça a série de moedas existentes no país
- Depois, forneça o valor do troco a ser produzido

```
    O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5
                                                          (Moedas, separadas por vírgulas)
Valor a pagar: 59
 Calcular
Moedas a liberar: 50,5,2,2
                                                                     (Moedas, separadas por vírgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
}
```

Avaliação:

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

10 5 40 6 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76199 - /

Prof Dr P Kantek (pkantek@gmail.com) VIVXh91a V: 1.02 EDUARDO CAVALHEIRO DIAS

Máquina de troco

24FRO405 - 3

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, CSS e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo, embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- · Forneça a série de moedas existentes no país

O seu programa indicará a sequência de :	
Vamos lá	
Preencha a data a seguir:	
Moedas: 1,35,50,2,5	(Moedas, separadas por vírgulas)
Valor a pagar: 59 (nnn)	
Calcular	
Moedas a liberar: 50,5,2,2	(Moedas, separadas por vírgulas)
Para testar	
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,1 1, 5, 10, 25, 50, 100 troco para 51 é: 50,1 1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1 1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2 1, 7, 11, 20, 66 troco para 78 é: 66,11,1	0,1,1,1

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
}
```

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

98 3 45 20 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76018 - /

CEP-UFPR-UTFPR-PUC/Pr-UP Sistemas de Informação 30/08/2024 - 11:25:00.2 Matemática aplicada Prof Dr P Kantek (pkantek@gmail.com) Máquina de troco VIVXh91a V: 1.02 67025 ERIC CORDEIRO 24FRO405 - 4

Máquina de troco

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, CSS e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

 $\bf O$ problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50\}$. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo, embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
  ordenar as moedas em ordem decrescente
  enquanto valor > 0
    i = 1
        enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
        escreva (moeda[i])
        valor = valor - moeda[i]
        fim{enquanto}
        i + i + 1
        fim{enquanto}
        fim{enquanto}
        fimfenquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort() ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente
console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

O uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);

Exemplo com strings:
const frase = "01å, mundo!";
// Extrair a palavra "mundo"
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- · Forneça a série de moedas existentes no país
- · Depois, forneça o valor do troco a ser produzido

```
• Depois, romeça o vaior do troco a ser produzido
• O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá

Preencha a data a seguir:

Moedas: 1,35,50,2,5 (Moedas, separadas por vírgulas)

Valor a pagar: 59 (Moedas, separadas por vírgulas)

Calcular

Moedas a liberar: 50,5,2,2 (Moedas, separadas por vírgulas)

Para testar

1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 79 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
}
```

1	Avaliação:					

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

66 40 3 2 1

e com o valor do troco igual a

71

Responda aqui:

Moedas a devolver



405-76025 - /

CEP-UFPR-UTFPR-PUC/Pr-UP 11:25:00.2 Sistemas de Informação 30/08/2024 - 11:25:00.2 Matemática aplicada Prof Dr P Kantek (pkantek@gmail.com) Máquina de troco VIVXh91a V: 1.02 76032 FELIPE EIJI KANO 24FRO405 - 5 _____ / ____ / ____ / _____ /

Máquina de troco

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, CSS e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

 $\bf O$ problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50\}$. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo, embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
  ordenar as moedas em ordem decrescente
  enquanto valor > 0
    i = 1
        enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
        escreva (moeda[i])
        valor = valor - moeda[i]
        fim{enquanto}
        i + i + 1
        fim{enquanto}
        fim{enquanto}
        fim{enquanto}
        fimfenquanto}
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort() ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente
console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

O uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);

Exemplo com strings:
const frase = "01å, mundo!";
// Extrair a palavra "mundo"
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- · Forneça a série de moedas existentes no país
- · Depois, forneça o valor do troco a ser produzido

```
• O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá

Preencha a data a seguir:

Moedas: 1,35,50,2,5 (Moedas, separadas por vírgulas)

Valor a pagar: 59 (nnn)

Calcular

Moedas a liberar: 50,5,2,2 (Moedas, separadas por vírgulas)

Para testar

1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
}
```

Avaliação:

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

9 15 20 10 1

e com o valor do troco igual a

84

Responda aqui:

Moedas a devolver



405-76032 - /

Prof Dr P Kantek (pkantek@gmail.com) VIVXh91a V: 1.02 FELIPE JESMIEL LEITE

Analise a tela do seu aplicativo

A encomenda

Máquina de troco

24FRO405 - 6

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, $\ensuremath{\mathrm{CSS}}$ e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

Apoio à folha h91

volve a lista de moedas a dispensar.

Geração de troco automático em moedas

- · Forneça a série de moedas existentes no país
- Depois, forneça o valor do troco a ser produzido

```
    O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5
                                                          (Moedas, separadas por vírgulas)
Valor a pagar: 59
 Calcular
Moedas a liberar: 50,5,2,2
                                                                     (Moedas, separadas por vírgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que de-

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
}
```

Avaliação:

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

45 9 21 18 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76049 - /

CEP-UFPR-UTFPR-PUC/Pr-UP Sistemas de Informação 30/08/2024 -11:25:00.2 Matemática aplicada Prof Dr P Kantek (pkantek@gmail.com) Máquina de troco VIVXh91a V: 1.02 GABRIELA BERNARDO PASQUALOTT 76056 24FRO405 - 7

Máquina de troco

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, $\ensuremath{\mathrm{CSS}}$ e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo, embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- Forneca a série de moedas existentes no país

 Depois, forneça o valor do troco a ser pr O seu programa indicará a sequência de 	
Vamos lá	
Preencha a data a seguir:	
Moedas: 1,35,50,2,5	(Moedas, separadas por vírgulas)
Valor a pagar: 59 (nnn)	
Calcular	
Moedas a liberar: 50,5,2,2	(Moedas, separadas por virgulas)
Para testar	
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10, 1, 5, 10, 25, 50, 100 troco para 51 é: 50,1 1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1 1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2 1, 7, 11, 20, 66 troco para 78 é: 66,11,1	10,1,1,1

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
```

A	Avaliação:				

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

21 35 40 13 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76056 - /

Prof Dr P Kantek (pkantek@gmail.com) VIVXh91a V: 1.02 GUILHERME VITOR DA SILVA

volve a lista de moedas a dispensar. Analise a tela do seu aplicativo

Máquina de troco

24FRO405 - 8

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, CSS e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo, embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

Apoio à folha h91

A encomenda

Geração de troco automático em moedas

- · Forneça a série de moedas existentes no país
- · Depois, forneca o valor do troco a ser produzido

O seu programa indicará a sequência de moedas a serem fornecidas
Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5 (Moedas, separadas por vírgulas)
Valor a pagar: 59 (nnn)
Calcular
Moedas a liberar: 50,5,2,2 (Moedas, separadas por virgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1 1, 5, 10, 25, 50, 100 troco para 51 é: 50,1 1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1 1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2 1, 7, 11, 20, 66 troco para 78 é: 66,11,1

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que de-

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
7
```

Avaliação:

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

71 21 6 40 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76063 - /

Prof Dr P Kantek (pkantek@gmail.com) VIVXh91a V: 1.02 LAURA PACHECO DELATTRE

Máquina de troco

24FRO405 - 9

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, $\ensuremath{\mathrm{CSS}}$ e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- · Forneça a série de moedas existentes no país
- Depois, forneça o valor do troco a ser produzido

```
    O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5
                                                          (Moedas, separadas por vírgulas)
Valor a pagar: 59
 Calcular
Moedas a liberar: 50,5,2,2
                                                                     (Moedas, separadas por vírgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
```

A	Avaliação:				

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

18 35 50 5 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76087 - /

Prof Dr P Kantek (pkantek@gmail.com) VIVXh91a V: 1.02 LAURO HENRIK RUTZ

Máquina de troco

24FRO405 - 10

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, $\overline{\text{CSS}}$ e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- · Forneça a série de moedas existentes no país
- Depois, forneça o valor do troco a ser produzido

```
    O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5
                                                          (Moedas, separadas por vírgulas)
Valor a pagar: 59
 Calcular
Moedas a liberar: 50,5,2,2
                                                                     (Moedas, separadas por vírgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
}
```

Avaliação:

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

66 5 35 11 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76687 - /

Prof Dr P Kantek (pkantek@gmail.com)

VIVXh91a V: 1.02 LUIZ MIGUEL OLINEK

Máquina de troco

24FRO405 - 11

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, $\overline{\text{CSS}}$ e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- Forneça a série de moedas existentes no país
- Depois, forneça o valor do troco a ser produzido

```
    O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5
                                                          (Moedas, separadas por vírgulas)
Valor a pagar: 59
 Calcular
Moedas a liberar: 50,5,2,2
                                                                     (Moedas, separadas por vírgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
}
```

Avaliação:

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

10 2 15 9 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76106 - /

Prof Dr P Kantek (pkantek@gmail.com)

VIVXh91a V: 1.02 NICOLE DA SILVA DE AZEVEDO

Máquina de troco

24FRO405 - 12

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, $\overline{\text{CSS}}$ e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- Forneça a série de moedas existentes no país
- Depois, forneça o valor do troco a ser produzido

```
    O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5
                                                          (Moedas, separadas por vírgulas)
Valor a pagar: 59
 Calcular
Moedas a liberar: 50,5,2,2
                                                                     (Moedas, separadas por vírgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
```

Avaliação:					
II .					
III					

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

21 71 6 3 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76113 - /

Prof Dr P Kantek (pkantek@gmail.com) VIVXh91a V: 1.02 PAULO JOSE FREIRE CORREA

volve a lista de moedas a dispensar. Analise a tela do seu aplicativo

Máquina de troco

24FRO405 - 13

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, CSS e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

Apoio à folha h91

A encomenda

Geração de troco automático em moedas

- Forneça a série de moedas existentes no país
- Depois, forneça o valor do troco a ser produzido

```
    O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5
                                                          (Moedas, separadas por vírgulas)
Valor a pagar: 59
 Calcular
Moedas a liberar: 50,5,2,2
                                                                     (Moedas, separadas por vírgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que de-

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
}
```

Avaliação:

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

20 21 40 13 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76120 - /

CEP-UFPR-UTFPR-PUC/Pr-UP 11:25:00.2 Matemática aplicada Máquina de troco 76137

Prof Dr P Kantek (pkantek@gmail.com) VIVXh91a V: 1.02 PEDRO HENRIQUE PIEKARSKI DE

Sistemas de Informação 30/08/2024 -

Máquina de troco

24FRO405 - 14

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, $\overline{\text{CSS}}$ e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo, embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- · Forneça a série de moedas existentes no país

Depois, forneça o valor do troco a ser pro O seu programa indicará a sequência de n	
Vamos lá	
Preencha a data a seguir:	
Moedas: 1,35,50,2,5	(Moedas, separadas por vírgulas)
Valor a pagar: 59 (nnn)	
Calcular	
Moedas a liberar: 50,5,2,2	(Moedas, separadas por vírgulas)
Para testar	
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10 1, 5, 10, 25, 50, 100 troco para 51 é: 50,1 1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1 1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2 1, 7, 11, 20, 66 troco para 78 é: 66,11,1),1,1,1

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
7
```

Avaliação:

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

2 5 45 13 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76137 - /

Prof Dr P Kantek (pkantek@gmail.com) VIVXh91a V: 1.02 RAUL KNAPKI CUNHA

volve a lista de moedas a dispensar. Analise a tela do seu aplicativo

A encomenda

Máquina de troco

76144

24FRO405 - 15

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, $\ensuremath{\mathrm{CSS}}$ e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

Apoio à folha h91

Geração de troco automático em moedas

- Forneça a série de moedas existentes no país
- Depois, forneça o valor do troco a ser produzido

```
    O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5
                                                          (Moedas, separadas por vírgulas)
Valor a pagar: 59
 Calcular
Moedas a liberar: 50,5,2,2
                                                                     (Moedas, separadas por vírgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que de-

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
}
```

F	Avaliação:					
ī	<u> </u>					
Ш						

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

5 10 9 6 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76144 - /

Prof Dr P Kantek (pkantek@gmail.com) VIVXh91a V: 1.02 VICTOR BRONDANI DOS SANTOS

Máquina de troco

24FRO405 - 16

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, $\overline{\text{CSS}}$ e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- Forneça a série de moedas existentes no país
- Depois, forneça o valor do troco a ser produzido

```
    O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5
                                                          (Moedas, separadas por vírgulas)
Valor a pagar: 59
 Calcular
Moedas a liberar: 50,5,2,2
                                                                     (Moedas, separadas por vírgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
}
```

Avaliação:

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

7 66 21 6 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76168 - /

Prof Dr P Kantek (pkantek@gmail.com)

VIVXh91a V: 1.02 VICTOR HUGO DOS SANTOS DE CA

Máquina de troco

24FRO405 - 17

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, $\ensuremath{\mathrm{CSS}}$ e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

- Forneça a série de moedas existentes no país
- Depois, forneça o valor do troco a ser produzido

```
    O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5
                                                          (Moedas, separadas por vírgulas)
Valor a pagar: 59
 Calcular
Moedas a liberar: 50,5,2,2
                                                                     (Moedas, separadas por vírgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
```

1	Avaliação:					
Ī						

Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

9 18 25 71 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76175 - /

Prof Dr P Kantek (pkantek@gmail.com) VIVXh91a V: 1.02

VITOR EDUARDO DE GOES FONTES

Máquina de troco

76182

24FRO405 - 18

Mais um exercício para desenvolver as habilidades até agora trabalhadas, a saber: HTML, $\overline{\text{CSS}}$ e finalmente a programação Javascript. O aplicativo é para calcular o troco em moedas.

O problema do troco Suponha um país que tem uma certa configuração de moedas. No Brasil: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. Nos EUA: $\{1,\,5,\,10,\,25,\,50$ e $100\}$ centavos. 5, 10, 25, 50}. Em qualquer um desses países, máquinas automáticas de venda de alguma coisa precisam calcular e entregar um eventual troco ao freguês. A combinação de moedas que totaliza o troco não é única. Por exemplo, em uma país cujas moedas são 1, 6, 10, o troco de 19 centavos pode ser 10,6,1,1,1 ou 6,6,6,1 ou outras combinações ainda, por exemplo, 19 moedas de 1.

Usa-se um algoritmo guloso aqui. O resultado quase sempre é o ótimo embora em algumas vezes, ele entregue um resultado pior. O critério de otimização é a menor quantidade de moedas (para preservar a máquina e minimizar as visitas de reposição de numerário).

O algoritmo

```
função troco (moedas[n], valor : inteiro)
 ordenar as moedas em ordem decrescente
 enquanto valor > 0
    i = 1
      enquanto i <= tamanho(moeda)
        enquanto valor >= moeda[i]
          escreva (moeda[i])
          valor = valor - moeda[i]
        fim{enquanto}
i + i + 1
      fim{enquanto}
 fim{enquanto}
fimfuncao
```

Ordenando Arrays com sort() O método sort() é o principal para ordenar arrays em JavaScript. Ele modifica o array original, então, se você precisar manter o array original, faça uma cópia antes de usar sort().

Por padrão, sort () ordena elementos como strings. Para ordenar números corretamente, você precisa fornecer uma função de comparação:

Acompanhe o exemplo:

```
const numbers = [4, 2, 5, 1, 3];
numbers.sort((a, b) => a - b); // Ordem crescente console.log(numbers); // Output: [1, 2, 3, 4, 5]
numbers.sort((a, b) => b - a); // Ordem decrescente
console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Dica para tirar a última vírgula Existem inúmeras maneiras de fazer isso (método substring(), usando expressões regulares, usando join(), mas aqui vai-se sugerir o uso de slice(). Acompanhe

```
ml= ml.slice(0, -1)
```

 ${\cal O}$ uso de slice é o mesmo para arrays ou strings e ele foi copiado do ambiente Python. Ou seja, se você sabe usar lá, sabe aqui.

```
array.slice(begin, end)
string.slice(begin, end)
```

Seus parâmetros são:

begin: O índice onde a extração deve começar. Se for negativo, ele será contado a partir do final da matriz ou string. end: O índice um a menos do que o índice onde a extração deve terminar. Se for omitido, todos os elementos até o final serão extraídos. Se for negativo, ele será contado a partir do final da matriz ou string.

Retorna uma nova matriz ou string contendo a parte extraída. A matriz ou string original não é modificada.

```
const frutas = ['maçã', 'banana', 'laranja', 'uva'];
// Extrair os dois primeiros elementos
const primeirasFrutas = frutas.slice(0, 2);
// Extrair do segundo elemento até o final
const ultimasFrutas = frutas.slice(1);
// Extrair os dois últimos elementos
const duasUltimasFrutas = frutas.slice(-2);
Exemplo com strings:
const frase = "Olá, mundo!";
// Extrair a palavra "mundo
const palavra = frase.slice(5);
// Extrair as primeiras 5 letras
const primeirasLetras = frase.slice(0, 5);
```

A encomenda

O aplicativo pede a sequência de moedas, separadas por vírgulas. Depois, pede-se o valor a devolver. O botão de calcular chama o programa que devolve a lista de moedas a dispensar.

Analise a tela do seu aplicativo

Apoio à folha h91

Geração de troco automático em moedas

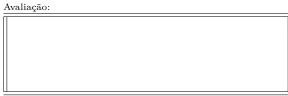
- Forneça a série de moedas existentes no país
- Depois, forneça o valor do troco a ser produzido

```
    O seu programa indicará a sequência de moedas a serem fornecidas

Vamos lá
Preencha a data a seguir:
Moedas: 1,35,50,2,5
                                                          (Moedas, separadas por vírgulas)
Valor a pagar: 59
 Calcular
Moedas a liberar: 50,5,2,2
                                                                     (Moedas, separadas por vírgulas)
Para testar
1, 5, 10, 25, 50, 100 troco para 98 é: 50,25,10,10,1,1,1
1, 5, 10, 25, 50, 100 troco para 51 é: 50,1
1, 5, 10, 25, 50, 100 troco para 77 é: 50,25,1,1
1, 2, 5, 35, 50 troco para 59 é: 50,5,2,2
1, 7, 11, 20, 66 troco para 78 é: 66,11,1
```

o JS

```
function execu(){
   sm=document.getElementById('sm').value;
   val=parseInt(document.getElementById('val').value);
   sm=sm.split('.')
   sm=sm.map(Number)
   sm.sort((a,b) => b-a);
   while (val>0) {
      i=0
   while (i<=sm.length){
      while (val>=sm[i]){
      ml=ml+sm[i]+',
   val=val-sm[i]
        i=i+1
     }
   ml= ml.slice(0, -1)
   document.getElementById('ml').value=ml
}
```



Depois que programar o aplicativo calcule a sequência de moedas no país cujas moedas são

9 98 3 33 1

e com o valor do troco igual a

Responda aqui: Moedas a devolver



405-76182 - /