Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

Exemplo

Problem

Programação Dinâmica

 Propriedade de subestrutura ótima

Obter recurs

 Algoritmo Bottom-UP para valor ótimo

. Obter uma solução

Aula 17 Programação Dinâmica Multiplicação de Sequência de Matrizes

Projeto e Análise de Algoritmos

Professor Eurinardo Rodrigues Costa Universidade Federal do Ceará Campus Russas

2021.1

Exemplo

Programação Dinâm

Propriedade de

2. Obter recursá

Algoritmo Bottom-UP
 para valor ótimo

Obter uma solução

Aulas Passadas

Multiplicação de Sequência de Matrizes

Multiplicações escalares de duas matrizes

Exemplo

Problema

Programação Dinâmica

- 1. Propriedade de subestrutura ótima
- 2. Obter recursão
- 3. Algoritmo Bottom-UP para valor ótimo
- 4. Obter uma solução ótima do problema

Aulas Passadas

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

Multiplicações escalares o duas matrizes

Exemplo Problem

Programação Dinâmica

- Propriedade de subestrutura ótima
- 2. Obter recur
- 3. Algoritmo Bottom-UP
 - l. Obter uma soluçã itima do problema

Multiplicações escalares d duas matrizes

Exemplo Problema

Matrizes

Programação Dinâmica

 Propriedade de subestrutura ótima

Obter recursă

Algoritmo Bottom-UP
para valor ótimo

Obter uma soluçã ima do problema

Definição (Programação Dinâmica)

Aulas Passadas

Multiplicação de Sequência de Matrizes

Multiplicações escalares duas matrizes

Problema

Programação Dinâmica

- Propriedade de subestrutura ótima
- Obter recursă
- Algoritmo Bottom-UP para valor ótimo
- . Obter uma solução tima do problema

Definição (Programação Dinâmica)

Seguir os passos:

(1) Verificar propriedade de subestrutura ótima

Aulas Passadas

Multiplicação de Sequência de Matrizes

Exemplo

Problema

Programação Dinâmica

- subestrutura ótima
- Algoritmo Bottom-UI
- . Obter uma solução

Definição (Programação Dinâmica)

- (1) Verificar propriedade de subestrutura ótima
 - "Um pedaço da solução ótima é solução ótima de um pedaço do problema?"

Matrizes

- Propriedade de subestrutura ótima
- 2. Obter recursão
- Algoritmo Bottom-UP para valor ótimo
- Obter uma solução
 ótimo do problemo

Definição (Programação Dinâmica)

- (1) Verificar propriedade de subestrutura ótima
 - "Um pedaço da solução ótima é solução ótima de um pedaço do problema?"
- (2) Obter uma recursão para o valor ótimo do problema

duas matrizes Exemplo

Problema

- Programação Dinâmio 1. Propriedade de
 - 2 Obtor reguração
- Algoritmo Bottom-UP
 para valor ótimo
- 4. Obter uma solução

Definição (Programação Dinâmica)

- (1) Verificar propriedade de subestrutura ótima
 - "Um pedaço da solução ótima é solução ótima de um pedaço do problema?"
- (2) Obter uma recursão para o valor ótimo do problema
- (3) Obter um algoritmo Bottom-UP para calcular o valor ótimo

Aulas Passadas

Multiplicação de Sequência de Matrizes

Exemplo

Problema

- Programação Dinami
 Propriedade de
 - subestrutura otima
- Algoritmo Bottom-UP
- para valor ótimo 4. Obter uma solução

Definição (Programação Dinâmica)

- (1) Verificar propriedade de subestrutura ótima
 - "Um pedaço da solução ótima é solução ótima de um pedaço do problema?"
- (2) Obter uma recursão para o valor ótimo do problema
- (3) Obter um algoritmo Bottom-UP para calcular o valor ótimo
- (4) Obter uma solução ótima do problema.

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Seguência de

Multiplicações escalares de duas matrizes

Exemplo

Problema

Matrizes

- Programação Diná
- Propriedade de subestrutura ótima
- 2. Obter recurs
- Algoritmo Bottom-UP
 para valor ótimo

◆ロト 4 雨 ト 4 豆 ト 4 豆 ・ 夕 Q ○

Exemplo

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

Multiplicações escalares de duas matrizes

Exemplo Problema

Programação Dinâmica

- Propriedade de subestrutura ótima
- 2. Obter recurs
- 3. Algoritmo Bottom-UP
 - . Obter uma solução tima do problema

Multiplicações escalares duas matrizes

Exemplo Problema

Matrizes

Programação Dinâmica 1. Propriedade de

subestrutura ótima

3. Algoritmo Bottom-UP

Obter uma solução ima do problema

Exemplo

 $A_{10\times100}\times B_{100\times5}\times C_{5\times50}$

Exemplo

 $A_{10\times100} \times B_{100\times5} \times C_{5\times50}$ Temos as opções: PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

Multiplicações escalares de duas matrizes Exemplo

Problema

Programação Dinâmica

- Propriedade de subestrutura ótima
- 2. Obter recurs
 - 3. Algoritmo Bottom-UP
 - . Obter uma solução tima do problema

duas matrizes Exemplo

Problema

Matrizes

Programação Dinâmica

- Propriedade de subestrutura ótima
- Obter recurs
 - 3. Algoritmo Bottom-UP
 - . Obter uma solução tima do problema

Exemplo

 $A_{10\times100} \times B_{100\times5} \times C_{5\times50}$ Temos as opções:

(I) $(A \times B)_{10 \times 5} \times C_{5 \times 50}$

duas matrizes Exemplo

Problema

Matrizes

Programação Dinâmica

- Propriedade de subestrutura ótima
- Obter recurs
 - 3. Algoritmo Bottom-UP
 - . Obter uma solução tima do problema

Exemplo

(I)
$$(A \times B)_{10 \times 5} \times C_{5 \times 50}$$

 $\rightarrow 10 \times 100 \times 5$

Problema

Matrizes

Programação Dinâmica

- Propriedade de subestrutura ótima
- Obter recursă
- Algoritmo Bottom-UP
 para valor ótimo
 - 4. Obter uma solução ótima do problema

Exemplo

(I)
$$(A \times B)_{10 \times 5} \times C_{5 \times 50}$$

 $\rightarrow 10 \times 100 \times 5 + 10 \times 5 \times 50$

Matrizes

Problema

- Propriedade de
- Obter recursão
- 3. Algoritmo Bottom-UP
 - Obter uma solução
 Stima do problema

Exemplo

(I)
$$(A \times B)_{10 \times 5} \times C_{5 \times 50}$$

 $\rightarrow 10 \times 100 \times 5 + 10 \times 5 \times 50 = 7500$

- Propriedade de
- subestrutura otima
- 3. Algoritmo Bottom-UP
- Obter uma solução

Exemplo

- (I) $(A \times B)_{10 \times 5} \times C_{5 \times 50}$ $\rightarrow 10 \times 100 \times 5 + 10 \times 5 \times 50 = 7500$
- (II) $A_{10\times100} \times (B \times C)_{100\times50}$

Matrizes

Problema

- Propriedade de
- 2. Obter recursão
- Algoritmo Bottom-UP
 para valor ótimo
 - para valor otimo 4. Obter uma solução

Exemplo

- (I) $(A \times B)_{10 \times 5} \times C_{5 \times 50}$ $\rightarrow 10 \times 100 \times 5 + 10 \times 5 \times 50 = 7500$
- (II) $A_{10\times100} \times (B \times C)_{100\times50}$ $\rightarrow 100 \times 5 \times 50$

Problema

Matrizes

- Programação Dinamio
 Propriedade de
- Propriedade de subestrutura ótima
- Obter recursão
- Algoritmo Bottom-UP para valor ótimo
- Obter uma solução
 ótima do problema

Exemplo

- (I) $(A \times B)_{10 \times 5} \times C_{5 \times 50}$ $\rightarrow 10 \times 100 \times 5 + 10 \times 5 \times 50 = 7500$
- (II) $A_{10\times100} \times (B \times C)_{100\times50}$ $\rightarrow 100 \times 5 \times 50 + 10 \times 100 \times 50$

Problema
Programação Dinâmica

- Propriedade de subestrutura ótima
- 2. Obter recursão
- Algoritmo Bottom-UP para valor ótimo
- para valor otimo 4. Obter uma solução

Exemplo

- (I) $(A \times B)_{10 \times 5} \times C_{5 \times 50}$ $\rightarrow 10 \times 100 \times 5 + 10 \times 5 \times 50 = 7500$
- (II) $A_{10\times100} \times (B \times C)_{100\times50}$ $\rightarrow 100 \times 5 \times 50 + 10 \times 100 \times 50 = 75000$

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Seguência de

Multiplicações escalares d duas matrizes

Problema

Matrizes

Programação Dinâmic

- Propriedade de subsetrutura ótima
- 2. Obter recurs
- 3. Algoritmo Bottom-UP
 - Obter uma soluçã tima do problema

Problema

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Seguência de

Multiplicações escalares de duas matrizes

Problema

Matrizes

- Programação Dinâmica
- Propriedade de subestrutura ótima
- 2. Obter recursă
- Algoritmo Bottom-UP
 Para valor átimo
 - 4. Obter uma soluçã ótima do problema

Problema

Entrada

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Seguência de

Multiplicações escalares de duas matrizes

Problema

Matrizes

Toblema

Propriedade de

2. Obter recurs

Algoritmo Bottom-UP
 para valor ótimo

4. Obter uma soluçã

Problema

Entrada

 \blacktriangleright Matrizes A_1, A_2, \cdots, A_n

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de

Problema

Matrizes

- ▶ Matrizes A_1, A_2, \cdots, A_n
- ▶ Inteiros p_0, p_1, \dots, p_n

Aulas Passadas

Multiplicação de Seguência de

Matrizes

Multiplicações escalares de

Exemplo

- Programação Dinâmica 1. Propriedade de
- subestrutura otima
- 3. Algoritmo Bottom-UP
- . Obter uma solução tima do problema

- \blacktriangleright Matrizes A_1, A_2, \cdots, A_n
- ▶ Inteiros p_0, p_1, \dots, p_n

tais que

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

- \blacktriangleright Matrizes A_1, A_2, \cdots, A_n
- ightharpoonup Inteiros p_0, p_1, \dots, p_n

tais que

 A_1 possui dimensão $p_0 \times p_1$

PAA - Aula 17

Prof Furinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

- \blacktriangleright Matrizes A_1, A_2, \cdots, A_n
- ► Inteiros p_0, p_1, \dots, p_n

tais que

 A_1 possui dimensão $p_0 \times p_1$ A_2 possui dimensão $p_1 \times p_2$ PAA - Aula 17

Prof Furinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

- \blacktriangleright Matrizes A_1, A_2, \cdots, A_n
- ► Inteiros p_0, p_1, \dots, p_n

tais que

 A_1 possui dimensão $p_0 \times p_1$ A_2 possui dimensão $p_1 \times p_2$ A_3 possui dimensão $p_2 \times p_3$

Aulas Passadas

Multiplicação de Sequência de Matrizes

- \blacktriangleright Matrizes A_1, A_2, \cdots, A_n
- ► Inteiros p_0, p_1, \dots, p_n

tais que

```
A_1 possui dimensão p_0 \times p_1
 A_2 possui dimensão p_1 \times p_2
 A_3 possui dimensão p_2 \times p_3
A_n possui dimensão p_{n-1} \times p_n
```

Aulas Passadas

Multiplicação de Sequência de Matrizes

- \blacktriangleright Matrizes A_1, A_2, \cdots, A_n
- lnteiros p_0, p_1, \dots, p_n

tais que

```
A_1 possui dimensão p_0 \times p_1
 A_2 possui dimensão p_1 \times p_2
 A_3 possui dimensão p_2 \times p_3
A_n possui dimensão p_{n-1} \times p_n
```

Objetivo

Aulas Passadas

Multiplicação de Sequência de Matrizes

- ► Matrizes A_1, A_2, \cdots, A_n
- ► Inteiros p_0, p_1, \dots, p_n

tais que

```
A_1 possui dimensão p_0 \times p_1

A_2 possui dimensão p_1 \times p_2

A_3 possui dimensão p_2 \times p_3

\vdots \vdots \vdots

A_n possui dimensão p_{n-1} \times p_n
```

Objetivo

Colocar parênteses em $A_1 \times A_2 \times A_3 \times \cdots \times A_n$ de modo que resultará no menor número de multiplicações escalares possível.

Aulas Passadas

Multiplicação de Sequência de Matrizes

Exemplo

Problema

- Propriedade de
- 2. Obter recursão
- Algoritmo Bottom-UP para valor ótimo

◆ロト ◆問 ◆ ▼ → ▼ ■ り ○ ○

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Seguência de

Multiplicações escalares d duas matrizes

Problem

Matrizes

Programação Dinâmio

- Propriedade de subestrutura ótima
- 2. Obter recurs
- 3. Algoritmo Bottom-UP
 - 4. Obter uma solução

1. Propriedade de subestrutura ótima

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Seguência de

Matrizes

Multiplicações escalares de duas matrizes

Problema

Programação Dinâmio 1. Propriedade de

subestrutura ótima

2. Obter recursão

 Algoritmo Bottom-UP para valor ótimo

> . Obter uma solução tima do problema

1. Propriedade de subestrutura ótima

"Um pedaço da **solução ótima** é **solução ótima** de um pedaço do problema?"

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

Multiplicações escalares de duas matrizes

Problema

Programação Dinâmica

- Propriedade de subestrutura ótima
- 2. Obter recursă
- 3. Algoritmo Bottom-UP
 - Obter uma solução ma do problema

Multiplicações escalares de duas matrizes

Problema

Matrizes

Programação Dinâmica 1. Propriedade de

subestrutura ótima

Algoritmo Bottom-UP

Obter uma solução

1. Propriedade de subestrutura ótima

"Um pedaço da **solução ótima** é **solução ótima** de um pedaço do problema?" Sim.

Exemplo

- Programação Dinâmica
 Propriedade de
- subestrutura ótima
- 3. Algoritmo Bottom-UP
 - 4. Obter uma solução Stima do problema

1. Propriedade de subestrutura ótima

"Um pedaço da **solução ótima** é **solução ótima** de um pedaço do problema?" Sim.

Seja $S = (A_1 \times \cdots \times A_k) \times (A_{k+1} \times \cdots \times A_n)$ uma **solução ótima** onde a última multiplicação realizada está entre A_k e A_{k+1} .

Exemplo

- Programação Dinâmica 1. Propriedade de
 - subestrutura ótima
- 3. Algoritmo Bottom-UP
 - l. Obter uma solução

1. Propriedade de subestrutura ótima

"Um pedaço da **solução ótima** é **solução ótima** de um pedaço do problema?" Sim.

Seja $S = (A_1 \times \cdots \times A_k) \times (A_{k+1} \times \cdots \times A_n)$ uma **solução ótima** onde a última multiplicação realizada está entre A_k e A_{k+1} . Os parênteses em S colocados em $(A_1 \times \cdots \times A_k)$ formam uma **solução ótima** para o "**pedaço**" $A_1 \times \cdots \times A_k$

1. Propriedade de

eubaetrutura ótima

Sequência de Matrizes

1. Propriedade de subestrutura ótima

"Um pedaço da solução ótima é solução ótima de um pedaco do problema?" Sim.

Seja $S = (A_1 \times \cdots \times A_k) \times (A_{k+1} \times \cdots \times A_n)$ uma solução ótima onde a última multiplicação realizada está entre A_k e A_{k+1} . Os parênteses em S colocados em $(A_1 \times \cdots \times A_k)$ formam uma **solução ótima** para o "**pedaco**" $A_1 \times \cdots \times A_k$, pois, em caso contrário, existe solução ótima s' melhor de modo que podemos ter $S' = s' \times (A_{k+1} \times \cdots \times A_n)$ melhor que S.

1. Propriedade de

eubaetrutura ótima

1. Propriedade de subestrutura ótima

"Um pedaço da solução ótima é solução ótima de um pedaco do problema?" Sim.

Seja $S = (A_1 \times \cdots \times A_k) \times (A_{k+1} \times \cdots \times A_n)$ uma solução ótima onde a última multiplicação realizada está entre A_k e A_{k+1} . Os parênteses em S colocados em $(A_1 \times \cdots \times A_k)$ formam uma **solução ótima** para o "**pedaco**" $A_1 \times \cdots \times A_k$, pois, em caso contrário, existe solução ótima s' melhor de modo que podemos ter $S' = s' \times (A_{k+1} \times \cdots \times A_n)$ melhor que S. Absurdo! Pois S é solução ótima.

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

Multiplicações escalares d duas matrizes

Problema

Programação Dinâmica

- subestrutura ótima
- 2. Obter recursão
- 3. Algoritmo Bottom-UP
 - l. Obter uma soluçã itima do problema

2. Obter recursão

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

Multiplicações escalares de duas matrizes Exemplo

- Programação Dinâmica
- Propriedade de subestrutura ótima
- 2. Obter recursão
- 3. Algoritmo Bottom-UP

duas matrizes

Exemplo

Problems

Problema Programação I

Matrizes

 Propriedade de subestrutura ótima

Obter recursão

Algoritmo Bottom-UP para valor ótimo

2. Obter recursão

M[i,j]= menor número de multiplicações escalares na multiplicação das matrizes $A_i \times A_{i+1} \times \cdots \times A_j$

Matrizes

duas matrizes

Exemplo

Problems

Programação Dinâm

subestrutura ótima

2 Obter recursão

Algoritmo Bottom-UP

ara valor ótimo . Obter uma solução

2. Obter recursão

M[i,j]= menor número de multiplicações escalares na multiplicação das matrizes $A_i \times A_{i+1} \times \cdots \times A_j$ Objetivo \rightarrow calcular M[1,n]

Problema

- Propriedade de subsetrutura ótima
- 2 Obter recursão
- 3. Algoritmo Bottom-UP
- Obter uma solução ótima do problema

2. Obter recursão

M[i,j] = menor número de multiplicações escalares na multiplicação das matrizes $A_i \times A_{i+1} \times \cdots \times A_j$ Objetivo \rightarrow calcular M[1,n]

$$M[i,j] = \begin{cases} \min\{M[i,k] + M[k+1,j] + p_{i-1} \times p_k \times p_j\} & \text{, se } i < j \\ 0 & \text{, se } i = j \end{cases}$$

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Seguência de

Multiplicações escalares d duas matrizes

Problema

Matrizes

Programação Dinâm

- Propriedade de subestrutura ótima
- 2. Obter recurs
- Algoritmo Bottom-UP para valor ótimo
 - Obter uma soluçã ótima do problema

3. Algoritmo Bottom-UP para valor ótimo

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

Multiplicações escalares d

Exemplo

Programação Dinâm

Propriedade de

2. Obter recursă

 Algoritmo Bottom-UP para valor ótimo

. Obter uma soluçã tima do problema

3. Algoritmo Bottom-UP para valor ótimo

Algoritmo 3: Mult-Matrizes(*P*)

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Seguência de

Multiplicações escalares de

Exemplo Problema

Matrizes

Programação Dinâmica

Propriedade de subestrutura ótima

2. Obter recurs

Algoritmo Bottom-UP para valor ótimo

. Obter uma solução

Multiplicações escalares de duas matrizes

Exemplo

Matrizes

Programação Dinâmica

 Propriedade de subestrutura ótima

Obter recursão

 Algoritmo Bottom-UP para valor ótimo

Obter uma soluçã ima do problema

3. Algoritmo Bottom-UP para valor ótimo

Algoritmo 4: Mult-Matrizes(P)

1 $n \leftarrow (comprimento de P) - 1$

Multiplicações escalares o duas matrizes

Exemplo

Matrizes

Programação Dinâmic

 Propriedade de subestrutura ótima

2. Obter recursão

 Algoritmo Bottom-UP para valor ótimo

Obter uma soluçã

3. Algoritmo Bottom-UP para valor ótimo

Algoritmo 5: Mult-Matrizes(P)

- 1 $n \leftarrow (comprimento de P) 1$
- 2 criar matriz $M_{n \times n}$

Problema

Programação Dinâm 1. Propriedade de

Obter recursão

 Algoritmo Bottom-UP para valor ótimo

4 D > 4 B > 4 B > 4 B > 4 D > 4 D >

3. Algoritmo Bottom-UP para valor ótimo

Algoritmo 6: Mult-Matrizes(P)

- 1 n ← (comprimento de P) 1
- 2 criar matriz $M_{n \times n}$
- 3 para $i \leftarrow 1$ até n faça

Multiplicações escalares o duas matrizes

- Exemplo
- Programação Dinâmi
- Propriedade de subestrutura ótima
- Algoritmo Bottom-UP
- para valor ótimo
 4. Obter uma solução

3. Algoritmo Bottom-UP para valor ótimo

Algoritmo 7: Mult-Matrizes(P)

- 1 $n \leftarrow (comprimento de P) 1$
- 2 criar matriz $M_{n \times n}$
- 3 para $i \leftarrow 1$ até n faça $M[i, i] \leftarrow 0$;

Exemplo Problema

Matrizes

- Propiema
- Propriedade de
- 2 Obter recursão
- Algoritmo Bottom-UP para valor ótimo
 - 4. Obter uma soluçã ótima do problema

3. Algoritmo Bottom-UP para valor ótimo

Algoritmo 8: Mult-Matrizes(*P*)

- 1 $n \leftarrow (comprimento de P) 1$
- 2 criar matriz $M_{n \times n}$
- 3 para i ← 1 até n faça M[i, i] ← 0;
- 4 para $i \leftarrow n-1$ até 1 faça

Multiplicações escalares o duas matrizes

Exemplo

Programação Dinâmic

Propriedade de subsetrutura étima

2. Obter recursã

 Algoritmo Bottom-UP para valor ótimo

4. Obter uma soluçã otima do problema

3. Algoritmo Bottom-UP para valor ótimo

Algoritmo 9: Mult-Matrizes(*P*)

- 1 $n \leftarrow (comprimento de P) 1$
- 2 criar matriz $M_{n \times n}$
- 3 para i ← 1 até n faça M[i, i] ← 0;
- 4 para $i \leftarrow n-1$ até 1 faça
- 5 para $j \leftarrow i + 1$ até n faça

Multiplicações escalares o duas matrizes

Exemplo

Matrizes

Programação Dinâmic

Propriedade de

2. Obter recursão

 Algoritmo Bottom-UP para valor ótimo

. Obter uma soluçã

```
3. Algoritmo Bottom-UP para valor ótimo
```

```
Algoritmo 10: Mult-Matrizes(P)
```

```
1 n \leftarrow (comprimento de P) - 1
```

2 criar matriz $M_{n \times n}$

6

3 para i ← 1 até n faça M[i, i] ← 0;

4 para $i \leftarrow n-1$ até 1 faça

5 para $j \leftarrow i + 1$ até n faça

 $M[i,j] \leftarrow \infty$

Matrizes

Algoritmo Bottom-UP

para valor ótimo

```
3. Algoritmo Bottom-UP para valor ótimo
```

```
Algoritmo 11: Mult-Matrizes(P)
```

```
1 n \leftarrow \text{(comprimento de } P\text{)} - 1
2 criar matriz M_{n \times n}
```

3 para $i \leftarrow 1$ até n faça $M[i, i] \leftarrow 0$;

4 para $i \leftarrow n-1$ até 1 faça

para $j \leftarrow i + 1$ até n faça 5 $M[i,j] \leftarrow \infty$ 6

para $k \leftarrow i$ até j-1 faça

Multiplicações escalares d

Exemplo

Matrizes

Problema

Programação Dinâmica

- Propriedade de subestrutura ótima
- 2. Obter recursão
- Algoritmo Bottom-UP para valor ótimo
 - ira valor otimo
 Obter uma solução

3. Algoritmo Bottom-UP para valor ótimo

```
Algoritmo 12: Mult-Matrizes(P)

1 n \leftarrow (comprimento de P) - 1

2 criar matriz M_{n \times n}

3 para i \leftarrow 1 até n faça M[i, i] \leftarrow 0;

4 para i \leftarrow n - 1 até 1 faça

5 para j \leftarrow i + 1 até n faça

6 M[i, j] \leftarrow \infty

7 para k \leftarrow i até j - 1 faça

8 q \leftarrow M[i, k] + M[k + 1, j] + p_{i-1} \times p_k \times p_j
```

Matrizes

Multiplicações escalares d

Exemplo

Programação Dinâmic

- Propriedade de subestrutura ótima
 - 2. Obter recursão
- Algoritmo Bottom-UP para valor ótimo
 - Obter uma solução

```
3. Algoritmo Bottom-UP para valor ótimo
```

```
Algoritmo 13: Mult-Matrizes(P)

1 n \leftarrow (\text{comprimento de }P) - 1

2 criar matriz M_{n \times n}

3 para i \leftarrow 1 até n faça M[i,i] \leftarrow 0;

4 para i \leftarrow n-1 até 1 faça

5 para j \leftarrow i+1 até n faça

6 M[i,j] \leftarrow \infty

7 para k \leftarrow i até j-1 faça

8 q \leftarrow M[i,k] + M[k+1,j] + p_{i-1} \times p_k \times p_j

9 se q < M[i,j] então
```

Matrizes

Multiplicações escalares o

duas matrizes

Exemplo Problem

Programação Dinâmica

- Propriedade de subestrutura ótima
 - 2. Obter recursão
- Algoritmo Bottom-UP para valor ótimo
 - Obter uma solução ima do problema

3. Algoritmo Bottom-UP para valor ótimo

```
Algoritmo 14: Mult-Matrizes(P)
1 n \leftarrow \text{(comprimento de } P\text{)} - 1
2 criar matriz M_{n \times n}
3 para i \leftarrow 1 até n faça M[i, i] \leftarrow 0;
4 para i \leftarrow n-1 até 1 faça
          para j \leftarrow i + 1 até n faça
5
                M[i,j] \leftarrow \infty
6
                para k \leftarrow i até j-1 faça
7
                      q \leftarrow M[i,k] + M[k+1,j] + p_{i-1} \times p_k \times p_i
8
                      se q < M[i, j] então
                            M[i,j] \leftarrow a
10
```

Matrizes

Multiplicações escalares d

duas matrizes

Problema

Programação Dinâmica

- Propriedade de subestrutura ótima
 - 2. Obter recursão
- Algoritmo Bottom-UP para valor ótimo
 - Obter uma soluçã ima do problema

3. Algoritmo Bottom-UP para valor ótimo

```
Algoritmo 15: Mult-Matrizes(P)
1 n \leftarrow \text{(comprimento de } P\text{)} - 1
2 criar matriz M_{n \times n}
3 para i \leftarrow 1 até n faça M[i, i] \leftarrow 0;
4 para i \leftarrow n-1 até 1 faça
          para j \leftarrow i + 1 até n faça
5
                M[i,j] \leftarrow \infty
6
                para k \leftarrow i até j-1 faça
7
                      q \leftarrow M[i,k] + M[k+1,j] + p_{i-1} \times p_k \times p_i
8
                      se q < M[i, j] então
                        M[i,j] \leftarrow q
10
```

11 **retorne** *M*[1, *n*]

3. Algoritmo Bottom-UP para valor ótimo

11 retorne M[1, n], S

```
Algoritmo 16: Mult-Matrizes(P)
1 n \leftarrow \text{(comprimento de } P\text{)} - 1
2 criar matriz M_{n\times n}, S_{n\times n};
3 para i \leftarrow 1 até n faça M[i, i] \leftarrow 0;
4 para i \leftarrow n-1 até 1 faça
          para j \leftarrow i + 1 até n faça
5
                M[i,j] \leftarrow \infty
6
                para k \leftarrow i até j-1 faça
7
                       q \leftarrow M[i,k] + M[k+1,j] + p_{i-1} \times p_k \times p_i
8
                       se q < M[i, j] então
                         M[i,j] \leftarrow q; S[i,j] \leftarrow k;
10
```

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

Multiplicações escalares de

Exemplo

Problema

- Propriedade de subestrutura ótima
 - 2. Obter recursão
- Algoritmo Bottom-UP para valor ótimo
 - Obter uma solução

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Seguência de

Multiplicações escalares o

Exemplo Problema

Matrizes

Programação Dinân

 Propriedade de subestrutura ótima

2. Obter recurs

Algoritmo Bottom-UP
 para valor ótimo

 Obter uma solução ótima do problema

4. Obter uma solução ótima do problema

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de

Multiplicações escalares de

Exemplo Problema

Matrizes

Programação Dinâmica

- Propriedade de subestrutura ótima
- 2. Obter recursão
- Algoritmo Bottom-UP
- Obter uma solução ótima do problema

4. Obter uma solução ótima do problema

Algoritmo 19: Imprimir(S, i, j)

PAA - Aula 17

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

Multiplicações escalares de duas matrizes

Problema

Programação Dinâmica

- Propriedade de
- 2. Obter recursã
- 3. Algoritmo Bottom-UP
- Obter uma solução ótima do problema

duas matrizes

Problema

Matrizes

Programação Dinâmica

- Propriedade de subestrutura ótima
- 2. Obter recursă
- Algoritmo Bottom-UP
 para valor ótimo
- Obter uma solução ótima do problema

4. Obter uma solução ótima do problema

Algoritmo 20: Imprimir(S, i, j)

1 se i=j então

Problema

Matrizes

Programação Dinâmica

- Propriedade de subsetrutura étima
- 2. Obter recursă
- Algoritmo Bottom-UP
 para valor ótimo
- Obter uma solução ótima do problema

4. Obter uma solução ótima do problema

Algoritmo 21: Imprimir(S, i, j)

- se i = j então
- escreva "A_i"

Matrizes

Multiplicações escalares de duas matrizes

Problema

- Propriedade de
- subestrutura otima
- 3. Algoritmo Bottom-UP
- Obter uma solução ótima do problema

4. Obter uma solução ótima do problema

Algoritmo 22: Imprimir(S, i, j)

- 1 se i = j então
- 2 escreva "A_i"
- 3 senão

Multiplicações escalares d duas matrizes

Problema

Matrizes

Programação Dinâmica

- Propriedade de subsetrutura étima
- 2. Obter recursão
- 3. Algoritmo Bottom-UP
- Obter uma solução ótima do problema

4. Obter uma solução ótima do problema

Algoritmo 23: Imprimir(S, i, j)

```
1 se i = j então
```

- 2 escreva "A_i"
- 3 senão
- 4 escreva "("

```
duas matrizes

Exemplo
```

Problema

Matrizes

Programação Dinâmica

- Propriedade de subestrutura ótima
- 2. Obter recursă
- Algoritmo Bottom-UP
 Para valor ótimo
- Obter uma solução ótima do problema

4. Obter uma solução ótima do problema

```
Algoritmo 24: Imprimir(S, i, j)
```

```
1 se i = j então
2 | escreva "A_i"
```

- 3 senão
- escreva "("
 Imprimir(S, i, S[i, j])

Matrizes

```
Multiplicações escalares de 
duas matrizes
```

Problema

Propriedade de

subestrutura ótima

3. Algoritmo Bottom-UP

 Obter uma solução ótima do problema

4. Obter uma solução ótima do problema

```
Algoritmo 25: Imprimir(S, i, j)

se i = j então
```

```
    2 | escreva "A<sub>i</sub>"
    3 senão
    4 | escreva "("
    5 | Imprimir(S, i, S[i, j])
    6 | escreva ")×("
```

```
duas matrizes

Exemplo
```

Problema

Matrizes

Programação Dinâmica

- Propriedade de subestrutura ótima
- 2. Obter recurs
- Algoritmo Bottom-UP
 Para valor étimo
- Obter uma solução ótima do problema

4. Obter uma solução ótima do problema

```
Algoritmo 26: Imprimir(S, i, j)

se i = j então

escreva "A_i"
```

```
3 senão

4 escreva "("

5 Imprimir(S, i, S[i, j])

6 escreva ")×("
```

Imprimir(S, S[i, j] + 1, j)

```
Multiplicações escalares d
duas matrizes
Exemplo
```

Problema

Matrizes

Programação Dinamica
 Propriedade de

- Propriedade de subestrutura ótima
- Obter recursă
- Algoritmo Bottom-UP
 para valor ótimo
- Obter uma solução ótima do problema

4. Obter uma solução ótima do problema

```
Algoritmo 27: Imprimir(S, i, j)

se i = j então

legonaries escreva "A_i"

senão

legonaries escreva "("

Imprimir(S, i, S[i,j])

escreva ")×("

Imprimir(S, S[i,j]) + 1, j)

escreva ")"
```

duas matrizes

Exemplo

Problema

Programação Dinâmica

 Propriedade de subestrutura ótima

2. Obter recurs

Algoritmo Bottom-UP
 para valor ótimo

 Obter uma solução ótima do problema

LEISERSON, C.E., STEIN, C., RIVEST, R.L., CORMEN T.H.

Algoritmos: teoria e prática, 3ed. Editora Campus, ano 2012.

Prof. Eurinardo

Aulas Passadas

Multiplicação de Sequência de Matrizes

Multiplicações escalares o duas matrizes

Problema

Programação Dinâmica

Propriedade de

2. Obter recurs

Algoritmo Bottom-UP
 Para valor étimo

4. Obter uma solução ótima do problema

Obrigado!