



UNIVERSIDADE  
FEDERAL DE  
SERGIPE



DEPARTAMENTO  
DE COMPUTAÇÃO

# Busca em cadeias (Força bruta e Rabin-Karp)

## Projeto e Análise de Algoritmos

Bruno Prado

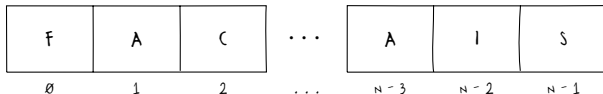
Departamento de Computação / UFS

# Introdução

- ▶ O que é uma cadeia?
  - ▶ É uma sequência de símbolos  $T$  com tamanho  $n$

# Introdução

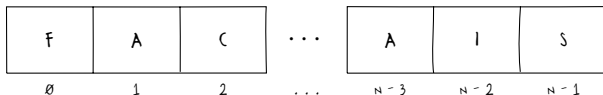
- ▶ O que é uma cadeia?
  - ▶ É uma sequência de símbolos  $T$  com tamanho  $n$
  - ▶ Os símbolos são definidos por um alfabeto finito  $\Sigma$



$$\Sigma = \{a, b, \dots, y, z\}$$

# Introdução

- ▶ O que é uma cadeia?
  - ▶ É uma sequência de símbolos  $T$  com tamanho  $n$
  - ▶ Os símbolos são definidos por um alfabeto finito  $\Sigma$

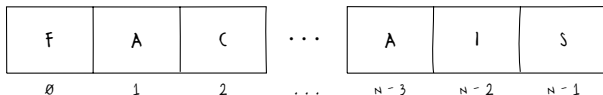


$$\Sigma = \{a, b, \dots, y, z\}$$

- ▶ Aplicações multidisciplinares
  - ▶ Biologia: representação da cadeia de DNA, sendo composta pelos símbolos  $A, C, G, T$

# Introdução

- ▶ O que é uma cadeia?
  - ▶ É uma sequência de símbolos  $T$  com tamanho  $n$
  - ▶ Os símbolos são definidos por um alfabeto finito  $\Sigma$

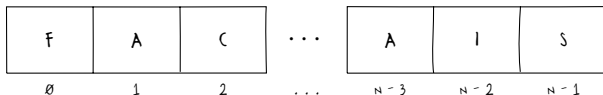


$$\Sigma = \{a, b, \dots, y, z\}$$

- ▶ Aplicações multidisciplinares
  - ▶ Biologia: representação da cadeia de DNA, sendo composta pelos símbolos  $A, C, G, T$
  - ▶ Computação: armazenamento de texto através do tipo *string*, com o padrão de codificação ASCII

# Introdução

- ▶ O que é uma cadeia?
  - ▶ É uma sequência de símbolos  $T$  com tamanho  $n$
  - ▶ Os símbolos são definidos por um alfabeto finito  $\Sigma$



$$\Sigma = \{a, b, \dots, y, z\}$$

- ▶ Aplicações multidisciplinares
  - ▶ Biologia: representação da cadeia de DNA, sendo composta pelos símbolos  $A, C, G, T$
  - ▶ Computação: armazenamento de texto através do tipo *string*, com o padrão de codificação ASCII
  - ▶ ...

# Introdução

- ▶ Notação e terminologia
  - ▶ Todas as cadeias de tamanho finito que podem ser construídas do alfabeto finito  $\Sigma$  é definido por  $\Sigma^*$

# Introdução

- ▶ Notação e terminologia
  - ▶ Todas as cadeias de tamanho finito que podem ser construídas do alfabeto finito  $\Sigma$  é definido por  $\Sigma^*$
  - ▶ Uma cadeia vazia é denotada pelo símbolo  $\varepsilon$

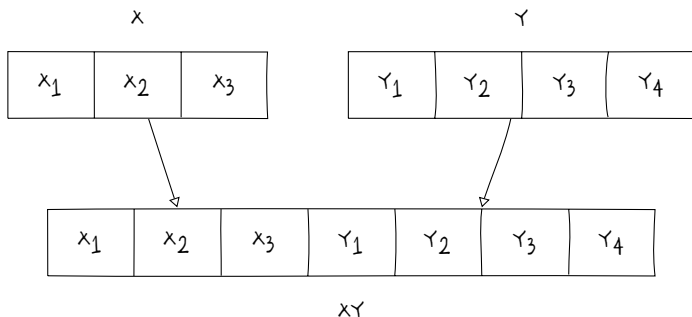


# Introdução

- ▶ Notação e terminologia
  - ▶ Todos as cadeias de tamanho finito que podem ser construídas do alfabeto finito  $\Sigma$  é definido por  $\Sigma^*$
  - ▶ Uma cadeia vazia é denotada pelo símbolo  $\varepsilon$
  - ▶ O tamanho de uma cadeia  $x$  é definida por  $|x|$

# Introdução

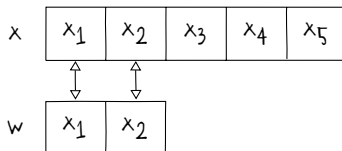
- ▶ Notação e terminologia
  - ▶ Todas as cadeias de tamanho finito que podem ser construídas do alfabeto finito  $\Sigma$  é definido por  $\Sigma^*$
  - ▶ Uma cadeia vazia é denotada pelo símbolo  $\varepsilon$
  - ▶ O tamanho de uma cadeia  $x$  é definida por  $|x|$
  - ▶ A concatenação de duas cadeias  $x$  e  $y$  resulta em uma cadeia  $xy$  com os caracteres de  $x$  seguidos dos caracteres de  $y$ , com tamanho total de  $|x| + |y|$



# Introdução

## ► Notação e terminologia

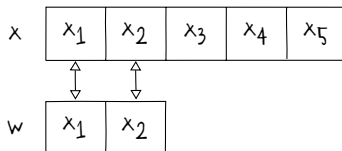
- Prefixo: a cadeia  $w$  é um prefixo da cadeia  $x$  ( $w \sqsubseteq x$ ) se  $x = wy$ , com  $y \in \Sigma^*$  e  $|w| \leq |x|$



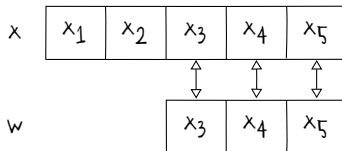
# Introdução

## ► Notação e terminologia

- Prefixo: a cadeia  $w$  é um prefixo da cadeia  $x$  ( $w \sqsubseteq x$ ) se  $x = wy$ , com  $y \in \Sigma^*$  e  $|w| \leq |x|$

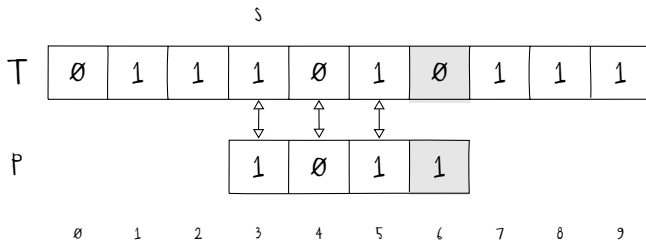


- Sufixo: a cadeia  $w$  é sufixo da cadeia  $x$  ( $w \sqsubseteq x$ ) se  $x = yw$ , com  $y \in \Sigma^*$  e  $|w| \leq |x|$



# Introdução

- ▶ Como pode ser definida a busca em cadeias?
  - ▶ É o processo para encontrar todas as ocorrências de um padrão  $P$  em uma cadeia  $T$  que possuem  $m$  e  $n$  símbolos, respectivamente, onde  $m \leq n$



$$\begin{aligned}\Sigma &= \{0, 1\} \\ |P| = m &= 4, \quad |T| = n = 10 \\ 0 \leq s &\leq n - m\end{aligned}$$

# Busca em cadeias

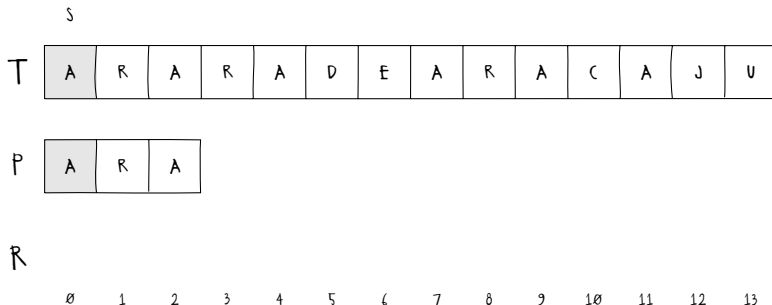
## ► Força bruta

```
1 // Padrão de tipos por tamanho
2 #include <stdint.h>
3 // Busca em cadeias por força bruta
4 void forca_bruta(int32_t* R, char* T, char* P) {
5     // Tamanhos das cadeias T e P
6     int32_t n = strlen(T), m = strlen(P);
7     // Iterando na cadeia T até o índice n - m
8     for(int32_t s = 0; s <= n - m; s++) {
9         // Contador de diferenças
10        uint32_t d = 0;
11        // Iterando na cadeia P enquanto d for zero
12        for(int32_t i = 0; i < m && d == 0; i++) {
13            // Caso os símbolos sejam diferentes
14            if(P[i] != T[s + i]) d++;
15        }
16        // O índice da combinação é armazenado
17        if(d == 0) inserir(R, s);
18    }
19 }
```

# Busca em cadeias

## ► Força bruta

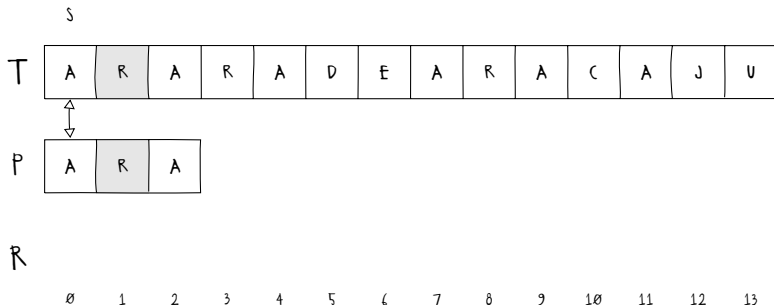
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias

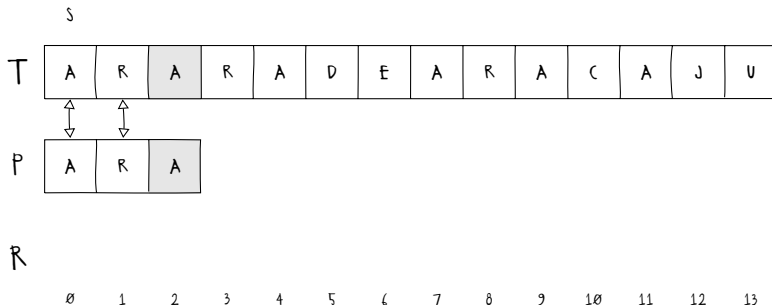




# Busca em cadeias

## ► Força bruta

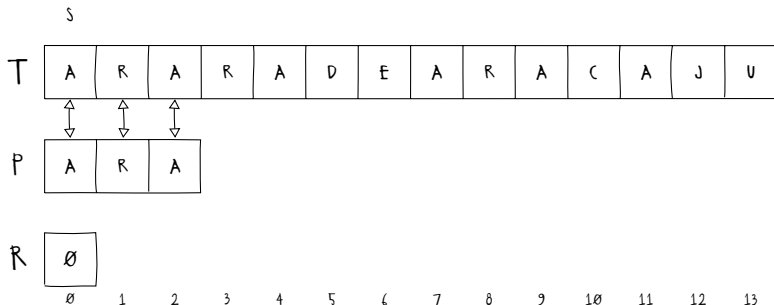
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

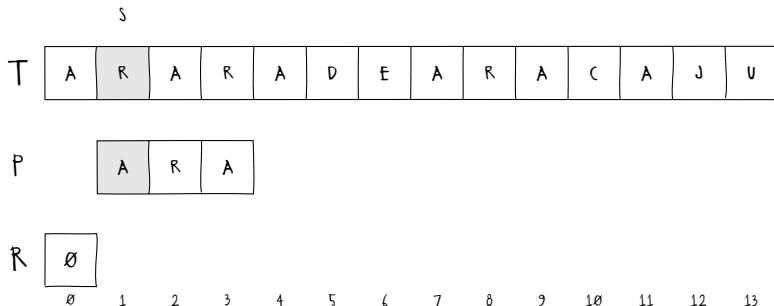
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

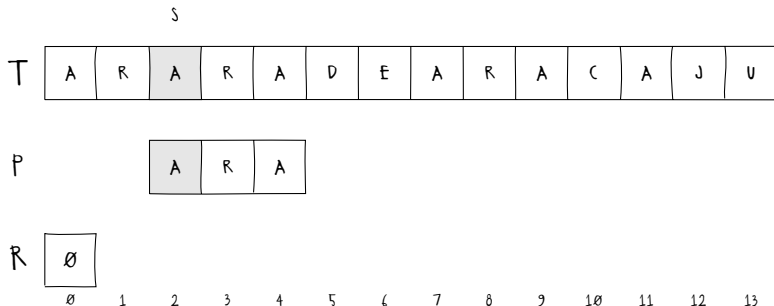
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

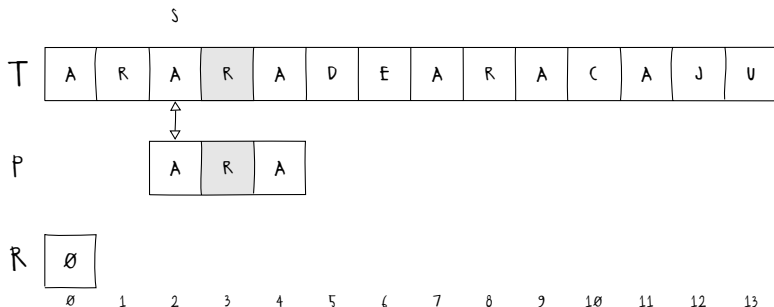
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

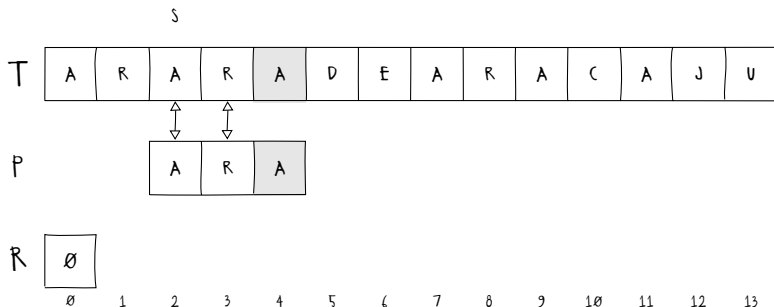
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

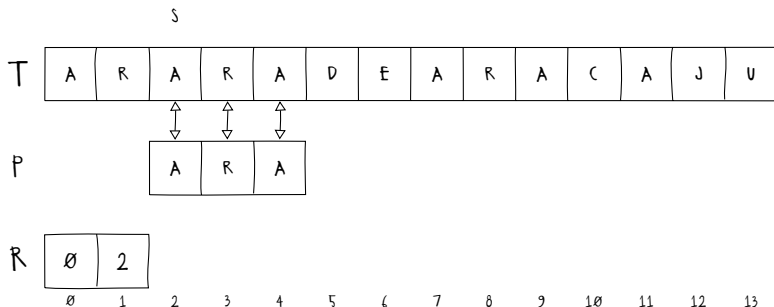
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

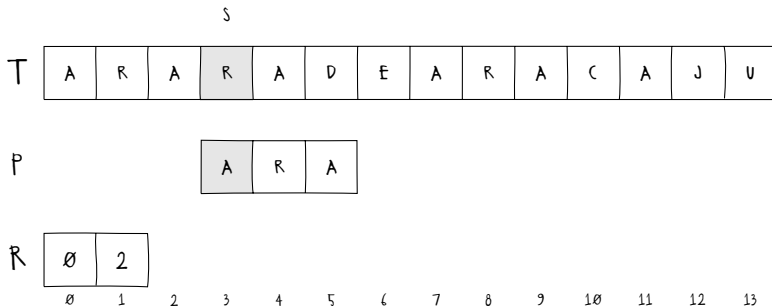
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias

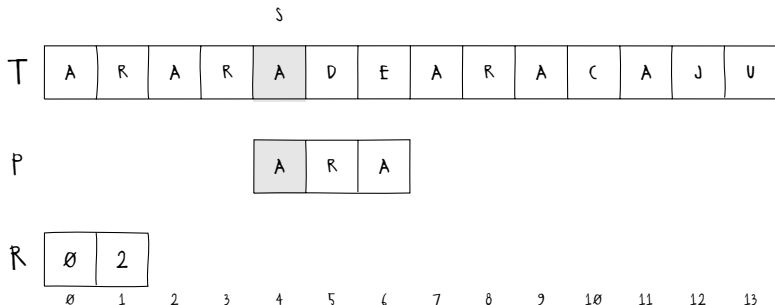




# Busca em cadeias

## ► Força bruta

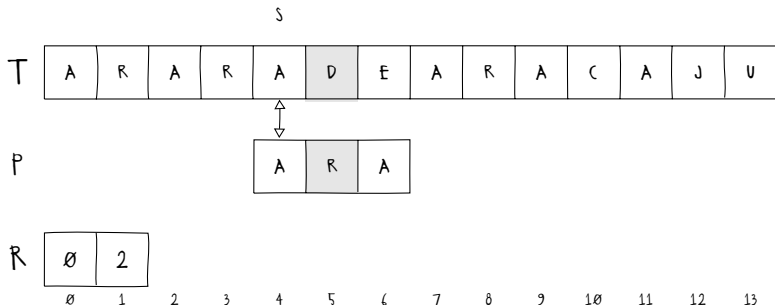
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

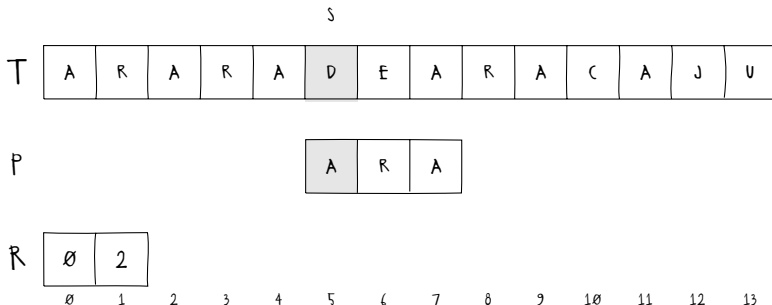
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

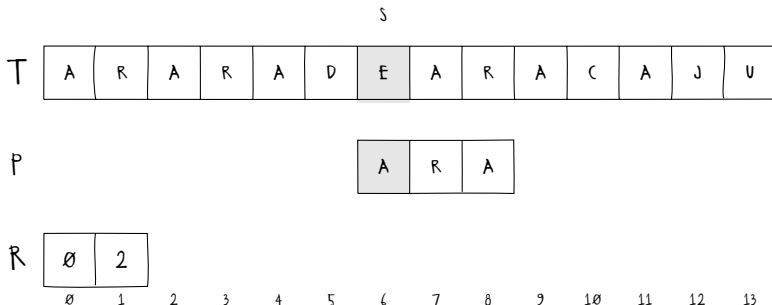
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

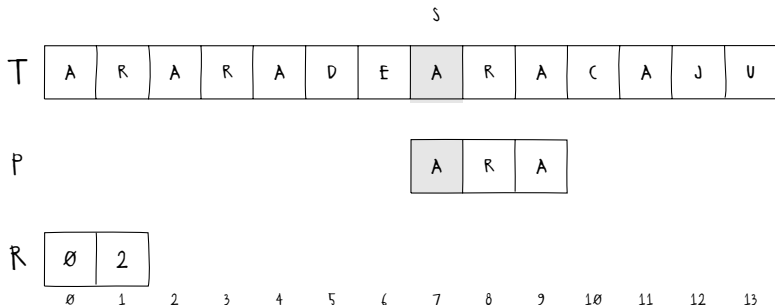
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

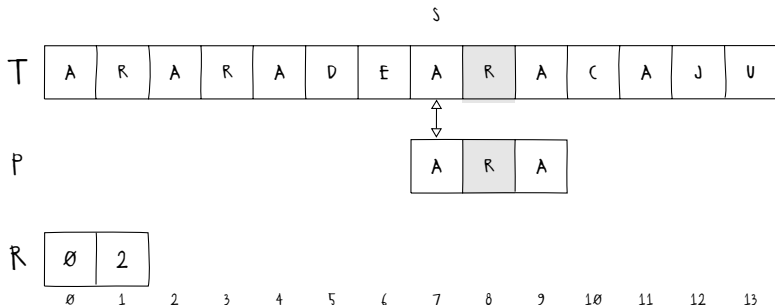
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

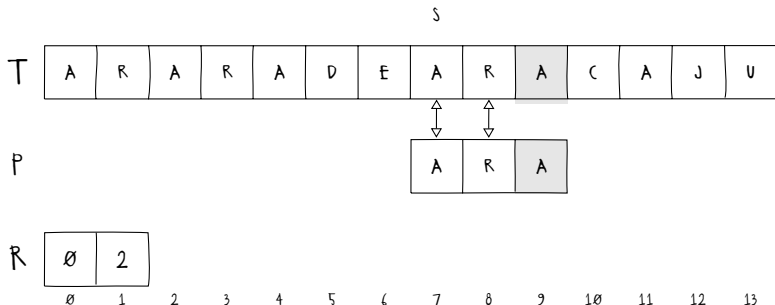
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

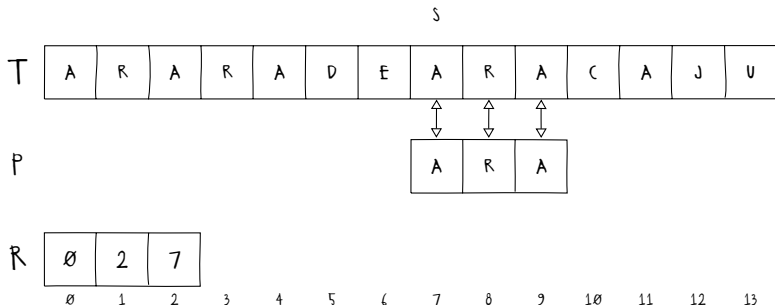
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias

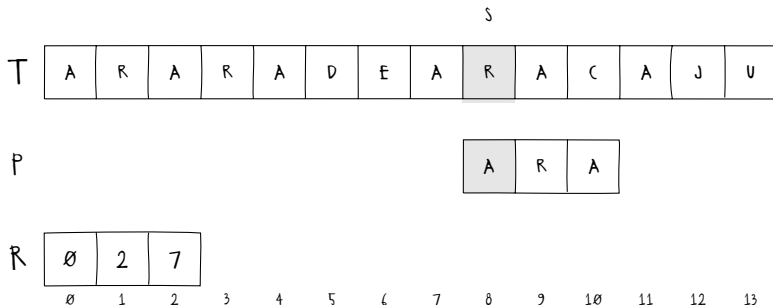




# Busca em cadeias

## ► Força bruta

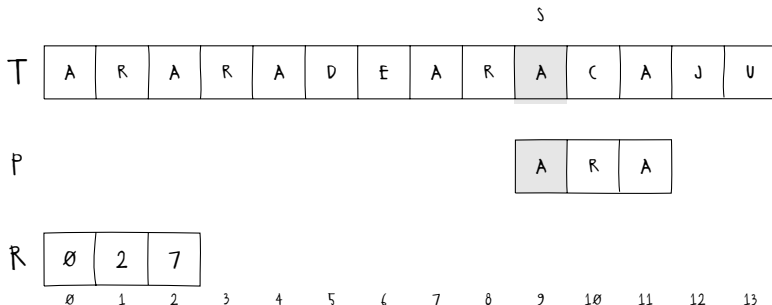
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

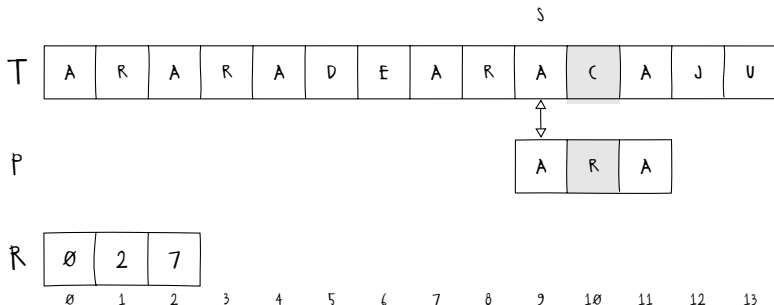
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

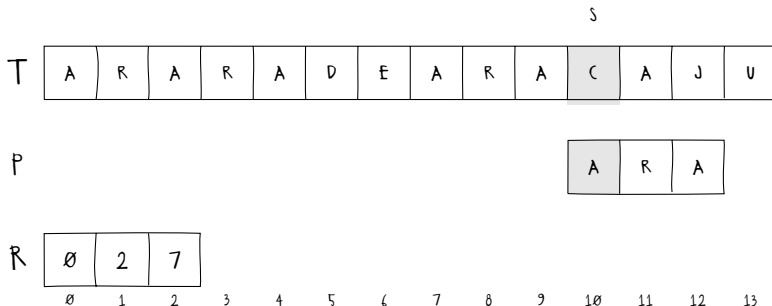
- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

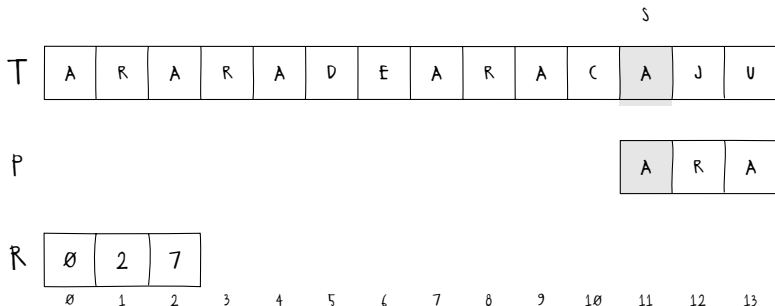
## ► Força bruta

- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

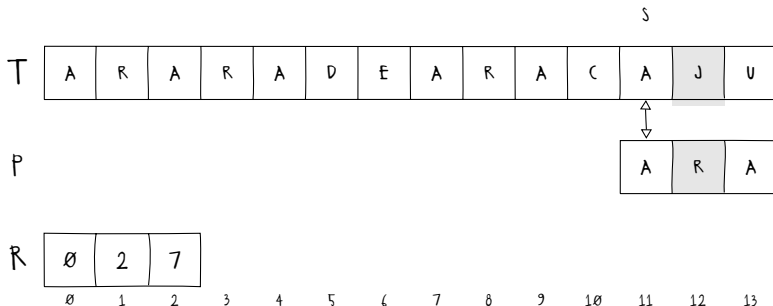
- ▶ Força bruta
  - ▶ Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

## ► Força bruta

- Realizando a busca pelo padrão  $P = ara$  na cadeia  $T = araradearacaju$  através do seu deslocamento e comparação um para um dos símbolos das cadeias



# Busca em cadeias

- ▶ Força bruta
  - ▶ Análise de complexidade
    - ▶ Espaço  $O(n + m)$
    - ▶ Tempo  $\Omega(n - m + 1)$  e  $O((n - m + 1) \times m)$

# Busca em cadeias

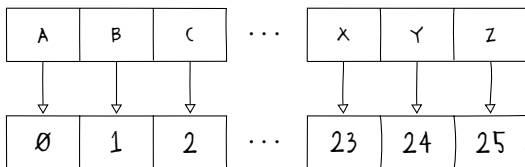
- ▶ Rabin-Karp
  - ▶ A busca na cadeia de símbolos é feita com o pré-processamento do padrão de texto



# Busca em cadeias

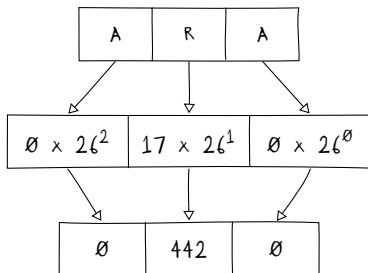
## ► Rabin-Karp

- A busca na cadeia de símbolos é feita com o pré-processamento do padrão de texto
- Considerando um alfabeto  $\Sigma = \{a, b, \dots, y, z\}$  com 26 símbolos, as cadeias podem ser representadas como números em base  $b = |\Sigma| = 26$



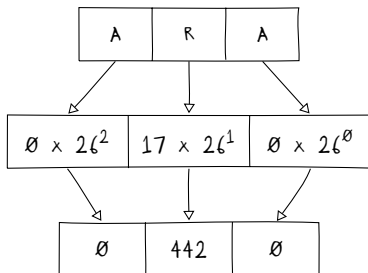
# Busca em cadeias

- ▶ Rabin-Karp
  - ▶ A busca na cadeia de símbolos é feita com o pré-processamento do padrão de texto
  - ▶ Considerando um alfabeto  $\Sigma = \{a, b, \dots, y, z\}$  com 26 símbolos, as cadeias podem ser representadas como números em base  $b = |\Sigma| = 26$



# Busca em cadeias

- ▶ Rabin-Karp
  - ▶ A busca na cadeia de símbolos é feita com o pré-processamento do padrão de texto
  - ▶ Considerando um alfabeto  $\Sigma = \{a, b, \dots, y, z\}$  com 26 símbolos, as cadeias podem ser representadas como números em base  $b = |\Sigma| = 26$



$$P = ara \rightarrow p = 0 + 442 + 0 = 442$$

# Busca em cadeias

- ▶ Rabin-Karp
  - ▶ A comparação das cadeias é feita comparando as representações numéricas do texto  $t$  e do padrão  $p$

# Busca em cadeias

- ▶ Rabin-Karp
  - ▶ A comparação das cadeias é feita comparando as representações numéricas do texto  $t$  e do padrão  $p$
  - ▶ Como o alfabeto e o padrão de busca podem conter muitos símbolos, o valor numérico  $b^m - 1$  pode não ser representável na plataforma ( $2^{64} - 1 \leq b^m - 1$ )

# Busca em cadeias

- ▶ Rabin-Karp
  - ▶ A comparação das cadeias é feita comparando as representações numéricas do texto  $t$  e do padrão  $p$
  - ▶ Como o alfabeto e o padrão de busca podem conter muitos símbolos, o valor numérico  $b^m - 1$  pode não ser representável na plataforma ( $2^{64} - 1 \leq b^m - 1$ )
  - ▶ Para contornar esta limitação, é aplicada a operação de módulo com um número primo  $q$  para restringir o tamanho do número gerado  $p \bmod q$  e reduzir as chances de colisões

# Busca em cadeias

## ► Rabin-Karp

- A comparação das cadeias é feita comparando as representações numéricas do texto  $t$  e do padrão  $p$
- Como o alfabeto e o padrão de busca podem conter muitos símbolos, o valor numérico  $b^m - 1$  pode não ser representável na plataforma ( $2^{64} - 1 \leq b^m - 1$ )
- Para contornar esta limitação, é aplicada a operação de módulo com um número primo  $q$  para restringir o tamanho do número gerado  $p \bmod q$  e reduzir as chances de colisões

O valor de  $b \times q$  precisa ser representável ( $b \times q \leq 2^{64} - 1$ )

# Busca em cadeias

## ► Rabin-Karp

```
1 // Padrão de tipos por tamanho
2 #include <stdint.h>
3 // Busca em cadeias com Rabin-Karp
4 void rabin_karp(int32_t* R, char* T, char* P, uint32_t
    b, uint32_t q) {
5     // Variáveis auxiliares
6     int32_t n = strlen(T), m = strlen(P), h = pow(b, m
        - 1) % q, t = 0, p = 0;
7     // Pré-processamento numérico das cadeias p e t
8     for(int32_t i = 0; i < m; i++) {
9         p = (b * p + v(P[i])) % q;
10        t = (b * t + v(T[i])) % q;
11    }
12    ...
21 }
```



# Busca em cadeias

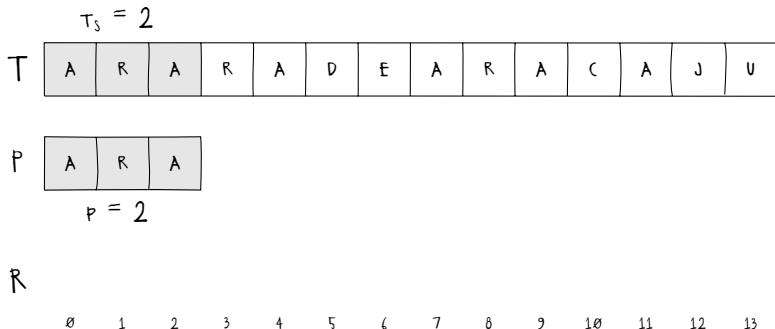
## ► Rabin-Karp

```
1 // Padrão de tipos por tamanho
2 #include <stdint.h>
3 // Busca em cadeias com Rabin-Karp
4 void rabin_karp(int32_t* R, char* T, char* P, uint32_t
   b, uint32_t q) {
5     // Variáveis auxiliares
6     int32_t n = strlen(T), m = strlen(P), h = pow(b, m
       - 1) % q, t = 0, p = 0;
7     ...
8     // Iterando na cadeia T até o índice n - m
9     for(int32_t s = 0; s <= n - m; s++) {
10         // Comparando o padrão com a cadeia
11         if((p == t) && igual(P, m, T, s))
12             // Armazenando índice de combinação
13             inserir(R, s);
14         // Deslocamento numérico
15         t = (b * (t - v(T[s]) * h) + v(T[s + m])) % q;
16     }
17 }
```

# Busca em cadeias

## ► Rabin-Karp

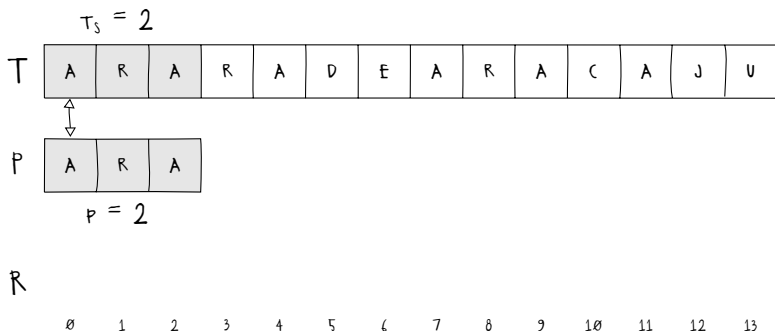
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

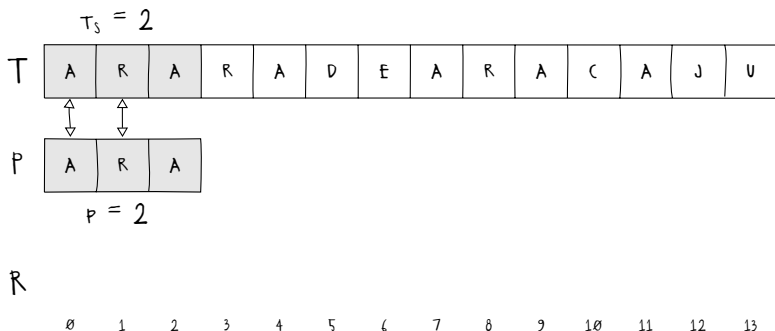
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

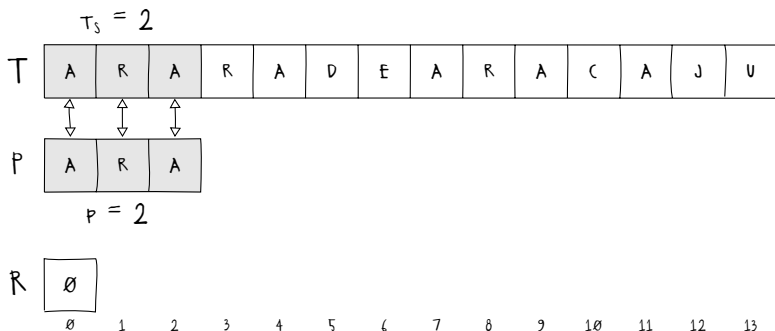
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

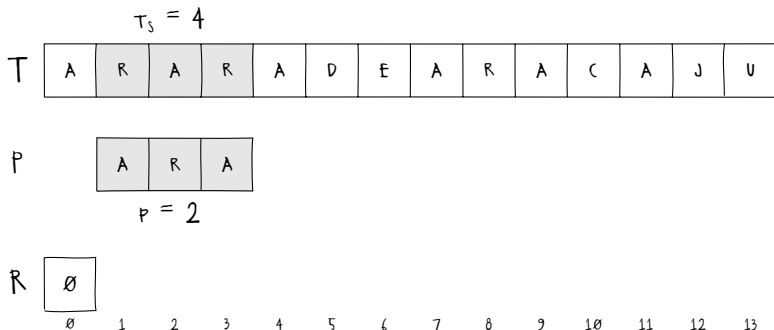
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

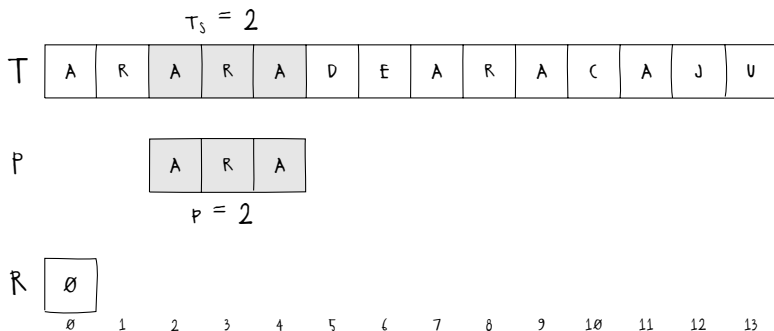
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

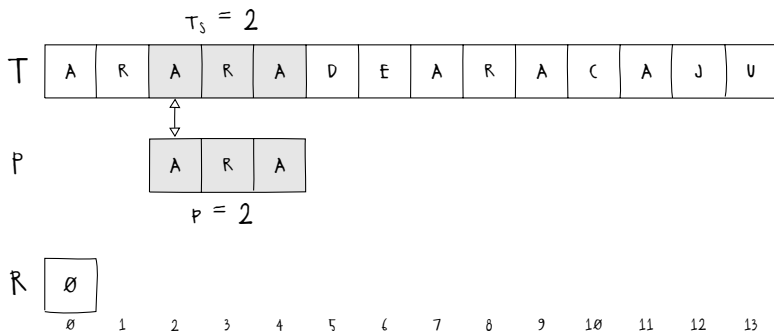
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$

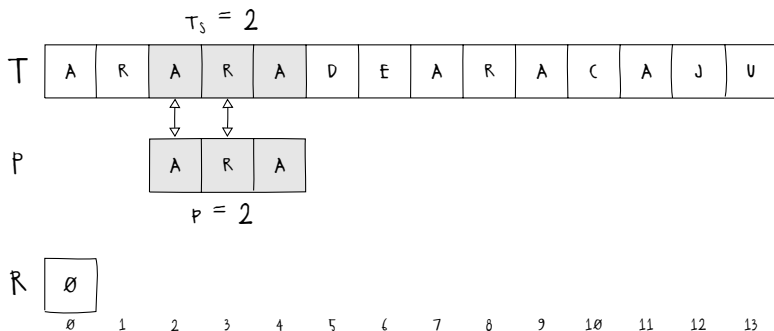




# Busca em cadeias

## ► Rabin-Karp

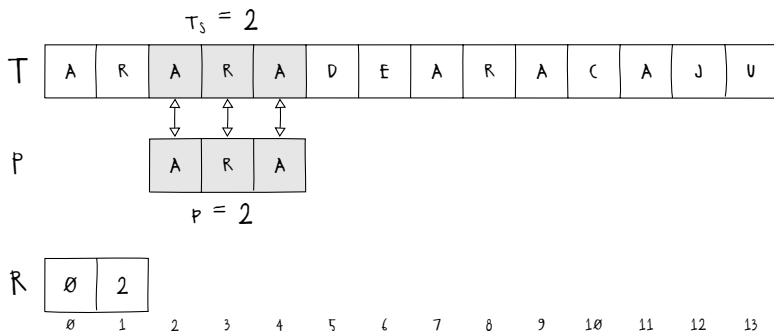
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

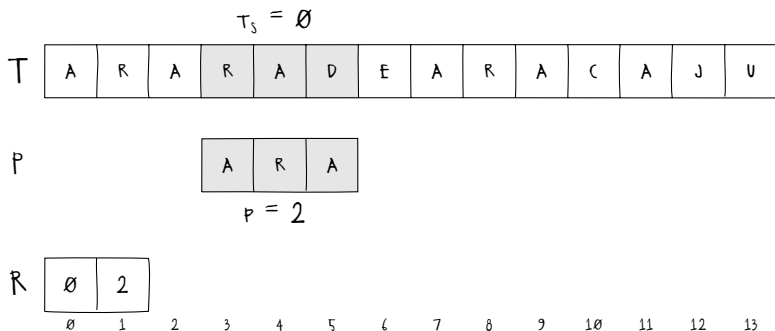
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

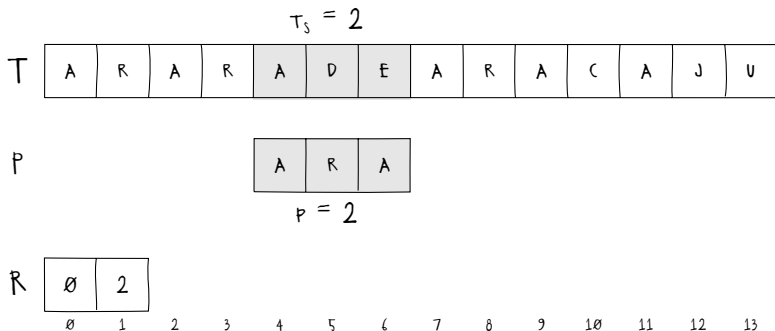
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

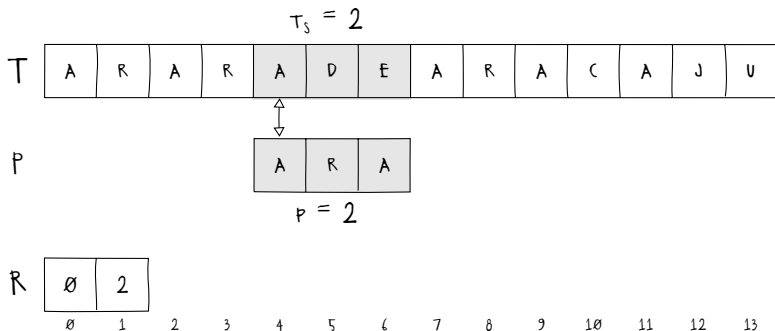
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

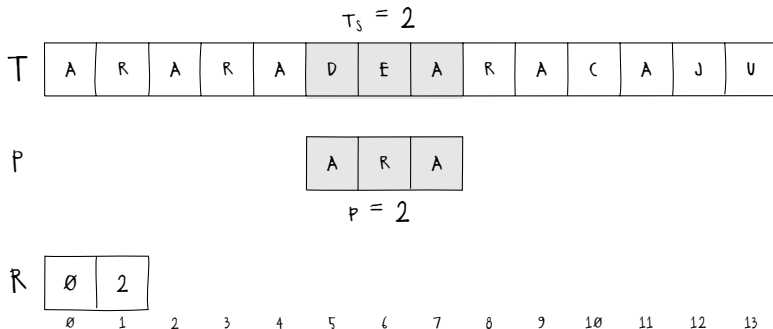
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

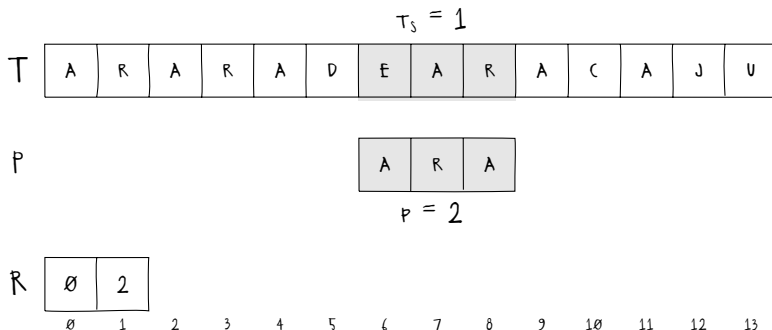
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

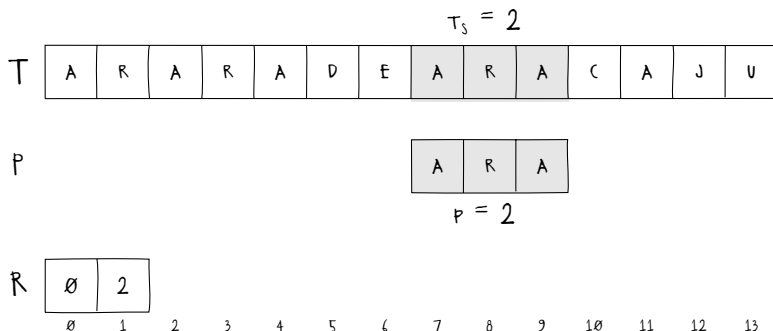
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$

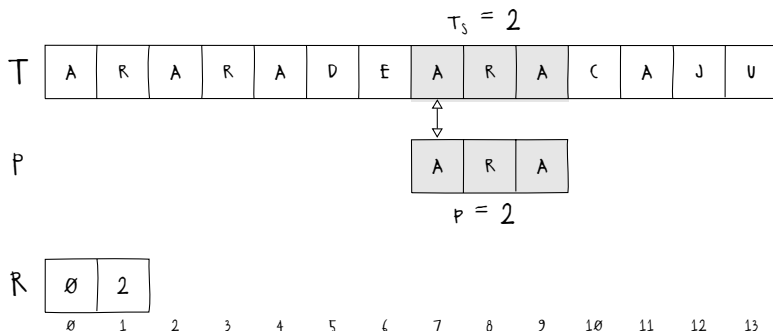




# Busca em cadeias

## ► Rabin-Karp

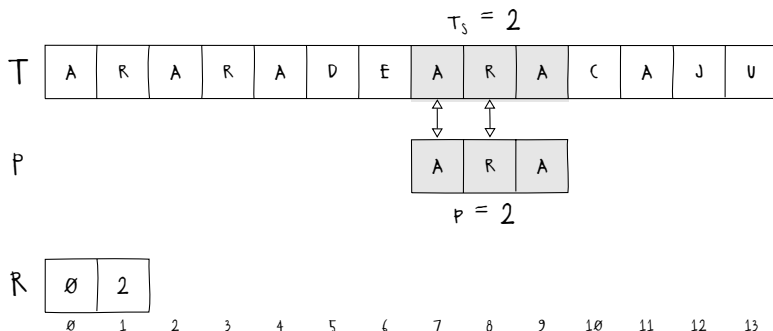
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

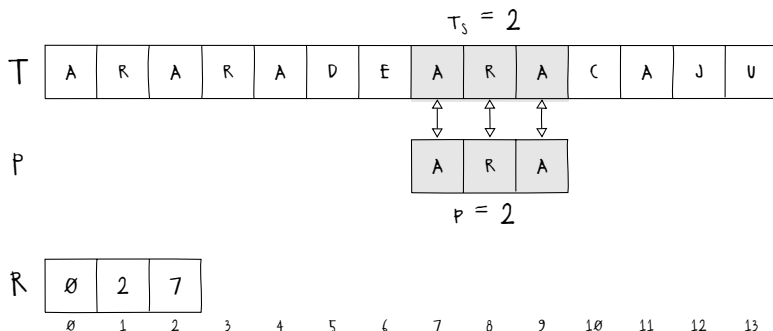
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

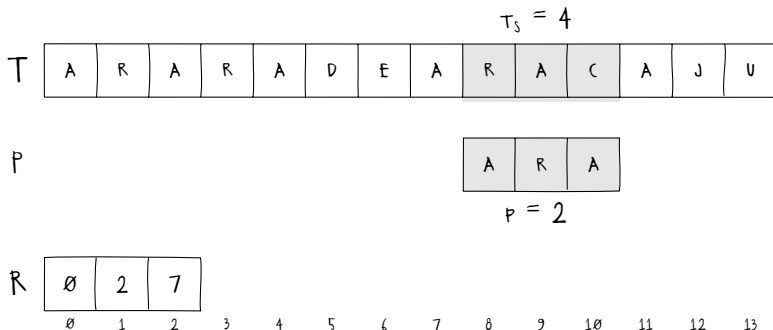
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

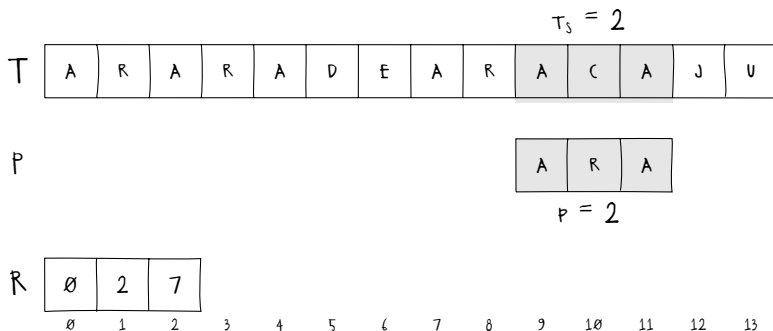
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

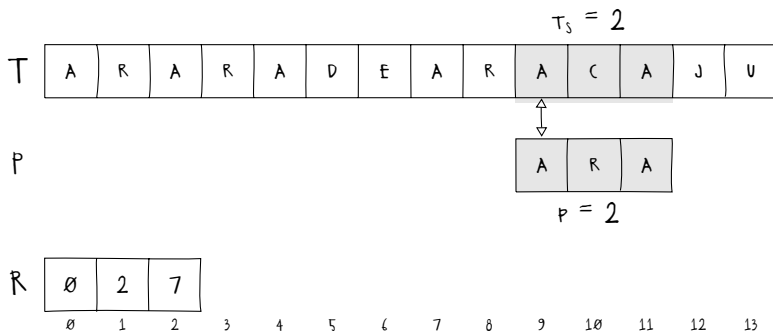
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

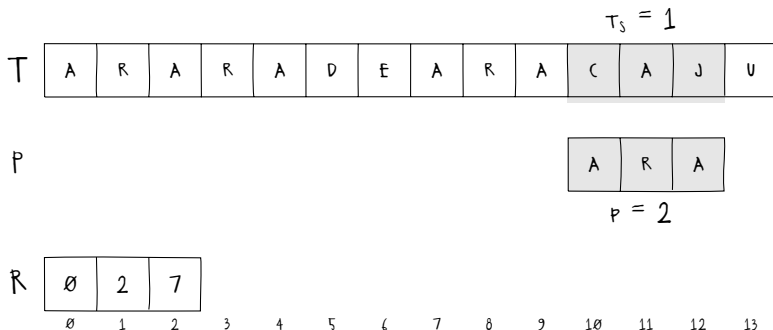
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

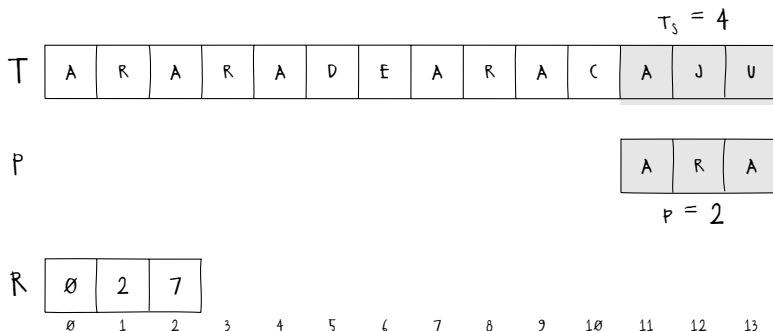
- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$



# Busca em cadeias

## ► Rabin-Karp

- Cada subcadeia de  $T$  de tamanho  $m$  tem seu valor numérico  $t_s$  calculado e comparado com o valor de  $p$ , considerando  $b = 26$ ,  $q = 5$  e  $p = 442 \bmod 5 = 2$





# Busca em cadeias

- ▶ Rabin-Karp
  - ▶ Análise de complexidade
    - ▶ Espaço  $O(n + m)$
    - ▶ Tempo  $\Theta(m) + \Omega(n - m + 1)$  e  $\Theta(m) + O((n - m + 1) \times m)$

# Exemplo

- ▶ Aplique os algoritmos de busca em cadeias para encontrar o padrão 111000 na sequência binária  
10111000110111100010101100011100001101101111
  - ▶ Execute passo a passo a busca na cadeia
  - ▶ Faça uma análise comparativa dos algoritmos, destacando seus princípios de funcionamento, além de suas respectivas vantagens e desvantagens