Teoria da Computação

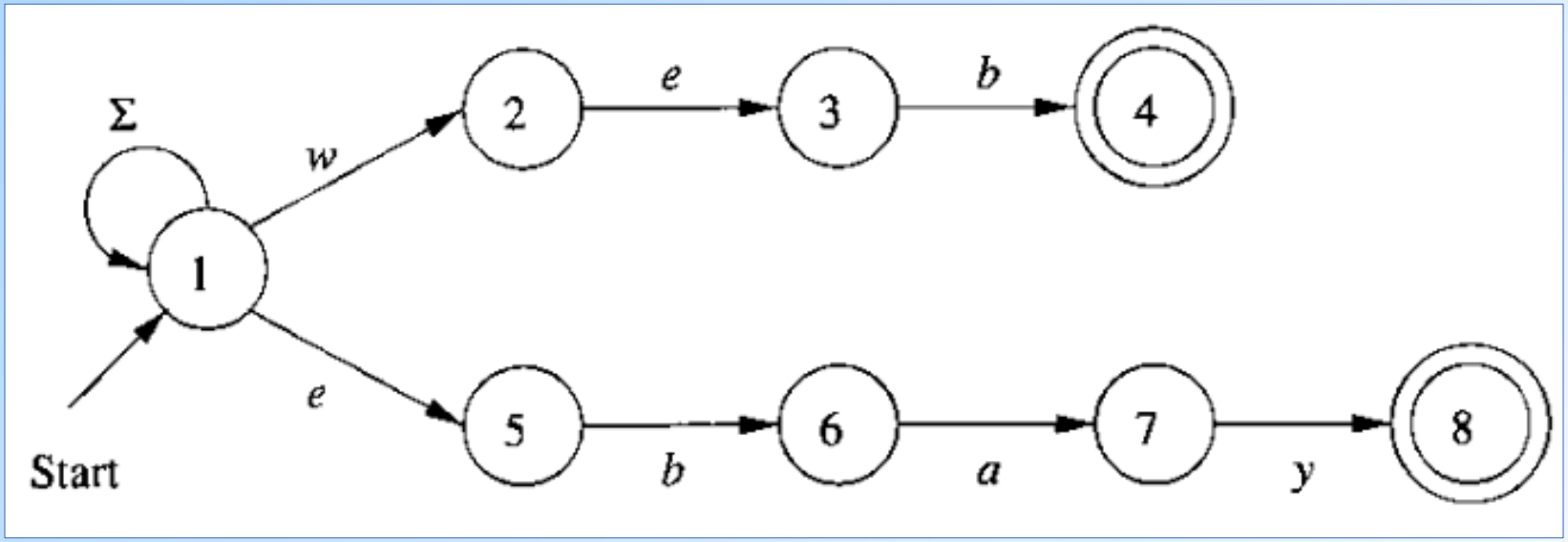
Apresentação e Motivação

Prof. Thiago Alves

Ementa

- Parte 1
  - Autômatos Finitos Determinísticos
  - Autômatos Finitos Não-determinísticos
  - Expressões Regulares

Autômatos Finitos



Expressões Regulares

- (0 + 1)*1 + 0(0 + 1)*
- [a-z][a-z]*@[a-z][a-z]*.(com|edu)

Importância

- Processamento de texto
- Análise Léxica de um Compilador
- Autômatos finitos modelam protocolos, circuitos eletrônicos, sistemas distribuídos e concorrentes, etc.

Importância

Donate to Wikipedia
Wikimedia Shop

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools
What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page

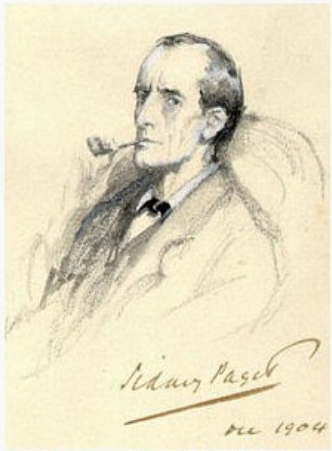
Sherlock Holmes (/ˈʃɜrlɒk ˈhoʊmz/) is a fictional character created by Scottish author and physician Sir Arthur Conan Doyle, a graduate of the University of Edinburgh Medical School. A London-based "consulting detective" whose abilities border on the fantastic, Holmes is known for his astute logical reasoning, his ability to adopt almost any disguise and his use of forensic science to solve difficult cases.

Holmes, who first appeared in print in 1887, was featured in four novels and 56 short stories. The first novel, *A Study in Scarlet*, appeared in *Beeton's Christmas Annual* in 1887 and the second, *The Sign of the Four*, in *Lippincott's Monthly Magazine* in 1890. The character's popularity grew with the first series of short stories in *The Strand Magazine*, beginning with "A Scandal in Bohemia" in 1891; additional short-story series and two novels (published in serial form) appeared from then to 1927. The events in the stories take place from about 1880 to 1914.

All but four stories are narrated by Holmes's friend and biographer, Dr. John H. Watson. Two are narrated by Holmes himself ("The Adventure of the Blanched Soldier" and "The Adventure of the Lion's Mane"), and two others are written in the third person ("The Adventure of the Mazarin Stone" and "His Last Bow"). In two stories ("The Adventure of the Musgrave Ritual" and "The Adventure of the *Gloria Scott*"), Holmes tells Watson the story from memory, with Watson narrating the frame story. The first and fourth novels, *A Study in Scarlet* and

Sherlock Holmes

Sherlock Holmes character



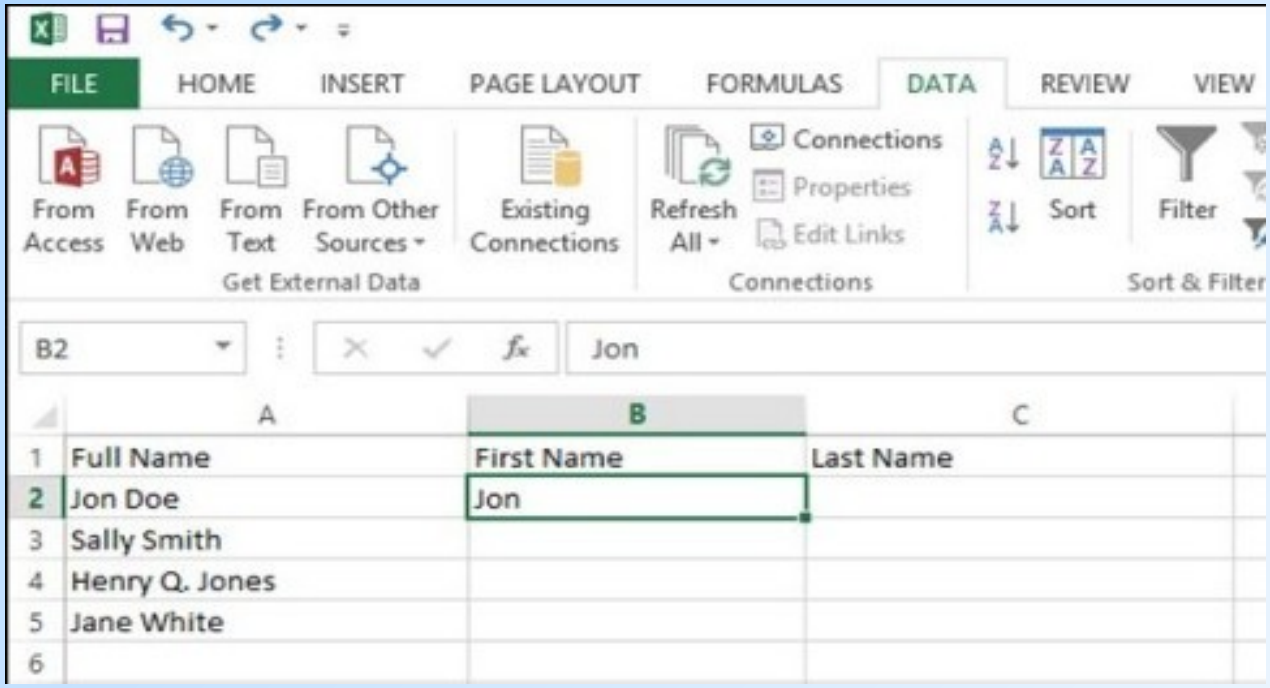
Sherlock Holmes in a 1904 illustration b

watson Realçar tudo Diferenciar maiúsculas/minúsculas Mais de 100 ocorrências

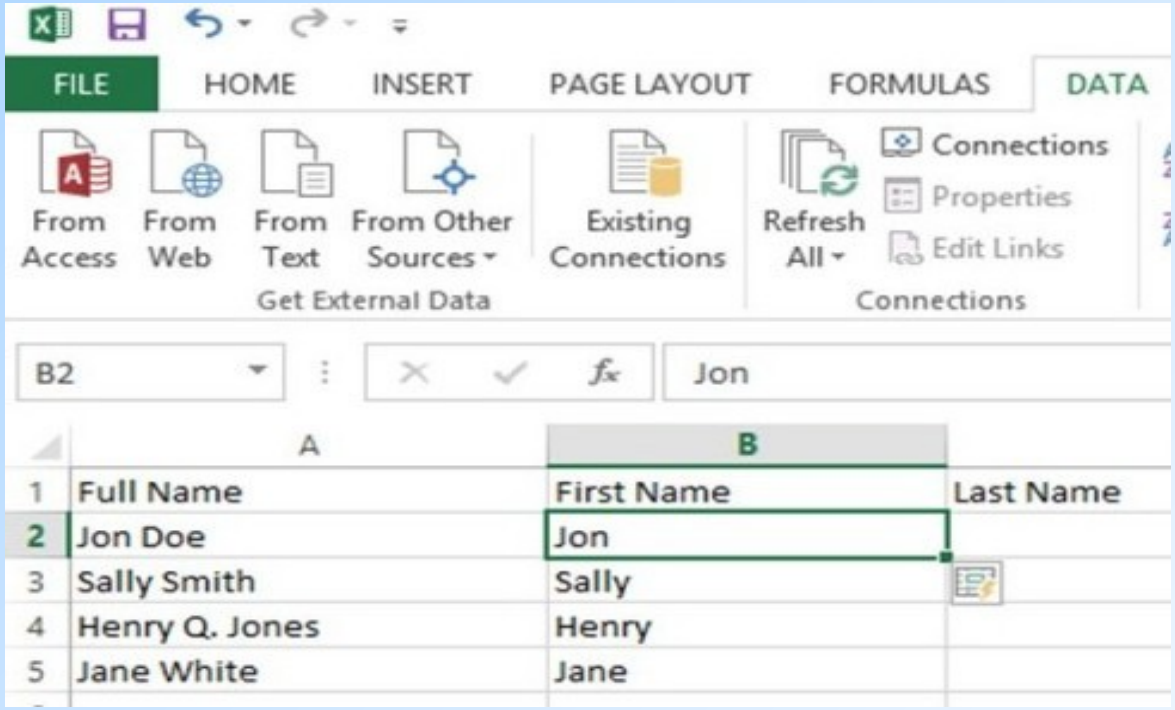
Importância

```
lista = []
#Colocando os elementos na lista
for i in range(5):
        print 'Digite o elemento da lista'
        lista.append(3.5*i)

#Somando os elementos da lista
soma = 0
for i in range(5):
        soma = soma + lista[i]

print 'A soma da lista e:', soma
```

Importância



Importância



Ementa

- Parte 2
  - Gramáticas Livres de Contexto
  - Autômatos de Pilha

GLC

⟨SENTENCE⟩ → ⟨NOUN-PHRASE⟩⟨VERB-PHRASE⟩

⟨NOUN-PHRASE⟩ → ⟨CMPLX-NOUN⟩ | ⟨CMPLX-NOUN⟩⟨PREP-PHRASE⟩

⟨VERB-PHRASE⟩ → ⟨CMPLX-VERB⟩ | ⟨CMPLX-VERB⟩⟨PREP-PHRASE⟩

⟨PREP-PHRASE⟩ → ⟨PREP⟩⟨CMPLX-NOUN⟩

⟨CMPLX-NOUN⟩ → ⟨ARTICLE⟩⟨NOUN⟩

⟨CMPLX-VERB⟩ → ⟨VERB⟩ | ⟨VERB⟩⟨NOUN-PHRASE⟩

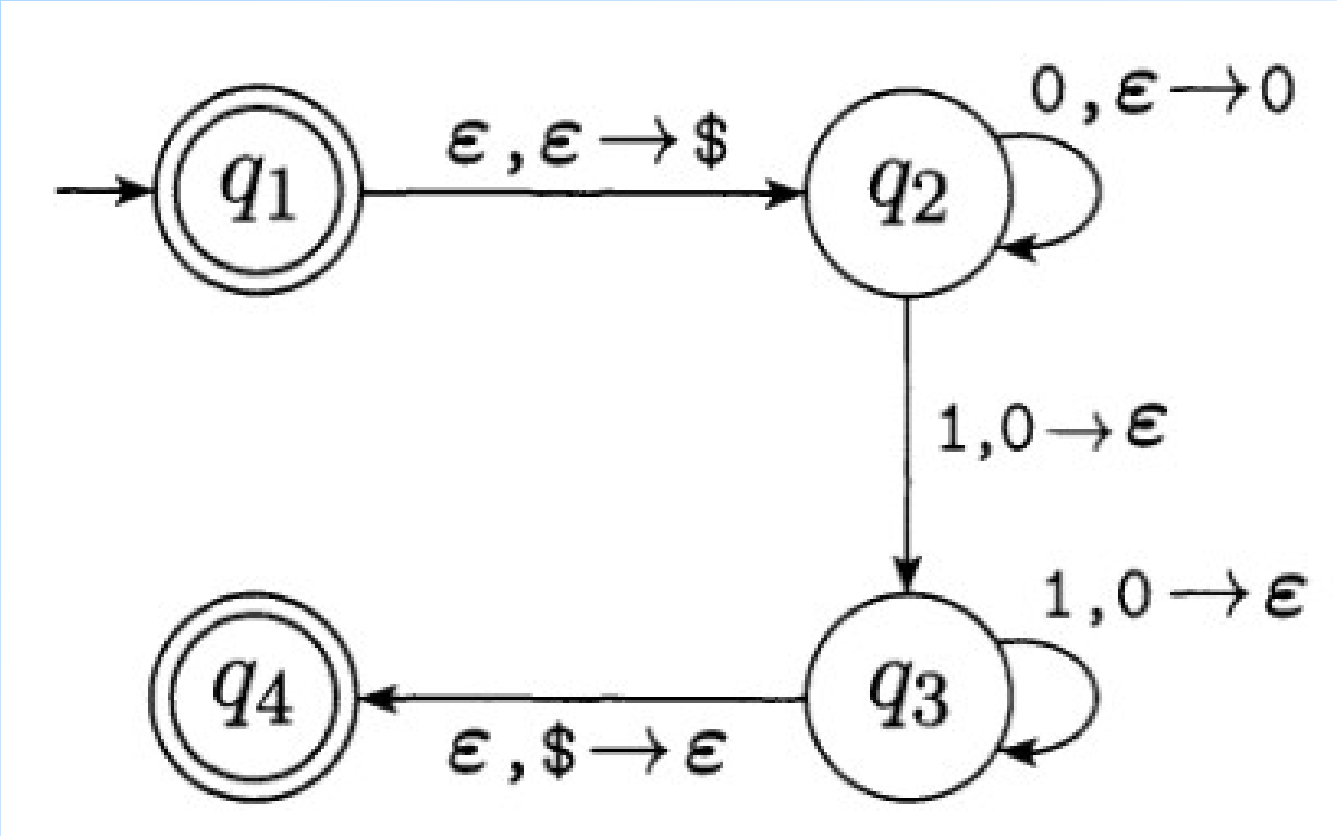
⟨ARTICLE⟩ → `a` | `the`

⟨NOUN⟩ → `boy` | `girl` | `flower`

⟨VERB⟩ → `touches` | `likes` | `sees`

⟨PREP⟩ → `with`

Autômato de Pilha



Importância

- Descrever a sintaxe de linguagens de programação
- Processamento de linguagem natural
- Processamento de documentos XML
- Análise sintática de um compilador

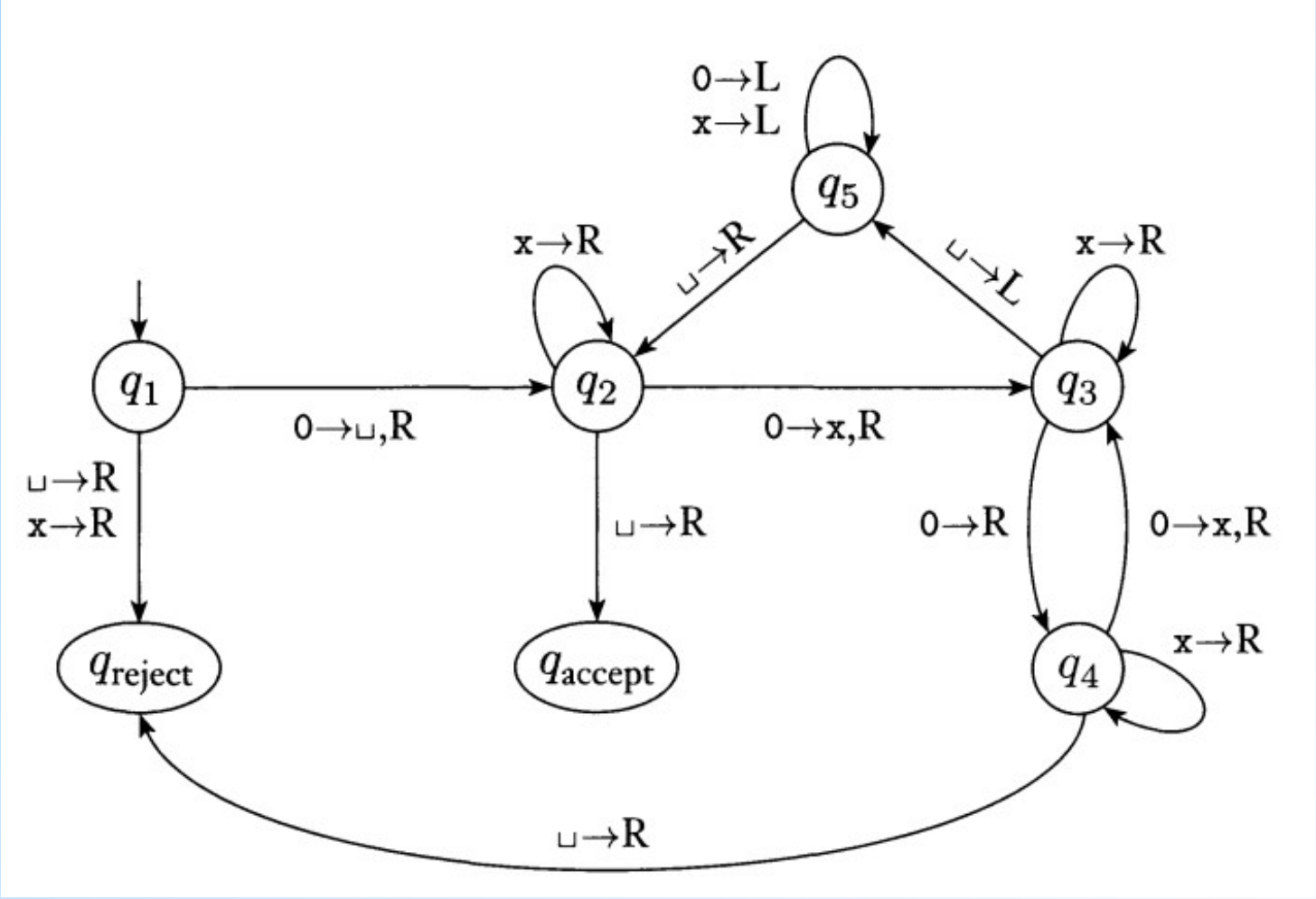
Importância

```
                Goal ::= MainClass ( ClassDeclaration )* <EOF>
           MainClass ::= "class" Identifier "{" "public" "static" "void" "main" "(" "String" "[" "]" Identifier ")" "{" Statement "}" "}"
  ClassDeclaration ::= "class" Identifier ( "extends" Identifier )? "{" ( VarDeclaration )* ( MethodDeclaration )* "}"
     VarDeclaration ::= Type Identifier ";"
MethodDeclaration ::= "public" Type Identifier "(" ( Type Identifier ( "," Type Identifier )* )? ")" "{" ( VarDeclaration )* ( Statement )* "return" Expression ";" "}"
              Type ::= "int" "[" "]"
                     | "boolean"
                     | "int"
                     | Identifier
         Statement ::= "{" ( Statement )* "}"
                     | "if" "(" Expression ")" Statement "else" Statement
                     | "while" "(" Expression ")" Statement
                     | "System.out.println" "(" Expression ")" ";"
                     | Identifier "=" Expression ";"
                     | Identifier "[" Expression "]" "=" Expression ";"
        Expression ::= Expression ( "&&" | "<" | "+" | "-" | "*" ) Expression
                     | Expression "[" Expression "]"
                     | Expression "." "length"
                     | Expression "." Identifier "(" ( Expression ( "," Expression )* )? ")"
                     | <INTEGER_LITERAL>
                     | "true"
                     | "false"
                     | Identifier
                     | "this"
                     | "new" "int" "[" Expression "]"
                     | "new" Identifier "(" ")"
                     | "!" Expression
                     | "(" Expression ")"
        Identifier ::= <IDENTIFIER>
```

Ementa

- Parte 3
  - Máquinas de Turing
  - Computabilidade de Problemas

Máquina de Turing



Importância

- Programas podem ser representados por máquinas de Turing
- Limitações do que os programas podem fazer
- Existem problemas para os quais não é possível fazer um programa para resolvê-lo?

Avaliações

- Primeira Etapa
  - $Provas_1$ e $Extras_1$
  - $N_1 = Provas_1 + Extras_1$
- Segunda Etapa
  - $Provas_2$ e $Extras_2$
  - $N_1 = Provas_2 + Extras_2$
- Média = $(2N_1 + 3N_2)/5$

Aprovação

- Média = $(2N_1 + 3N_2)/5$
- Se Média ≥ 7: Aprovado A
- Se Média < 3: Reprovado Direto
- Se Média < 7 e Média ≥ 3:
  - Se Média ≥ 5 e $Prova_1$ ≥ 5 e $Prova_2$ ≥ 5: Aprovado B
  - Caso contrário: Prova Final

Prova Final

- Média Final = (Média + Prova Final)/2
- Se Média Final ≥ 5: Aprovado B
- Se Média Final < 5: Reprovado

Bibliografia

- Sipser, Introduction to the Theory of Computation 3rd Edition.
- Hopcroft, Motwani e Ullman, *Automata Theory, Languages, and Computation* 3rd Edition.
- Lewis e Papadimitriou, Elements of the Theory of Computation 2nd Edition

Faltas e 2ª Chamada

- Cuidado com reprovação por falta!
- Pedido de 2ª chamada deve ser feito na recepção com a devida justificativa.
- Não faço 2ª chamada sem o deferimento do pedido!