
Lógica Aplicada à Computação

Prof. Vivek Nigam

Aula 1 – Introdução ao Curso

Bem vindos ao curso!

Algumas Informações Práticas

Vivek Nigam

`vivek.nigam@gmail.com`

Sala: LARQSS (Laboratório de Arquitetura e Sistemas de Software)

Página Web

`www.nigam.info/logica/logica.html`

Objetivos do Curso

Objetivos do Curso

- Entender a **importância** da lógica para a computação;

“Lógica esta para a Computação como o Cálculo para a Matemática.” (M. Vardi)

Objetivos do Curso

- Entender a **importância** da lógica para a computação;
“Lógica esta para a Computação como o Cálculo para a Matemática.” (M. Vardi)
- Saber como interpretar **fórmulas lógicas proposicionais** e de **primeira ordem**;

$$(a \wedge b) \Rightarrow c \qquad \forall x \exists y. [p(x) \Rightarrow q(x, y)]$$

Objetivos do Curso

- Entender a **importância** da lógica para a computação;

“Lógica esta para a Computação como o Cálculo para a Matemática.” (M. Vardi)

- Saber como interpretar **fórmulas lógicas proposicionais** e de **primeira ordem**;

$$(a \wedge b) \Rightarrow c \qquad \forall x \exists y. [p(x) \Rightarrow q(x, y)]$$

- Entender a diferença entre **sintaxe** e **semântica** e suas conexões: **completude e corretude**.

\vdash VS \models

Objetivos do Curso

- Como **modelar problemas** usando fórmulas lógicas.

Problema das N-Rainhas, modelar o comportamento de programs como seus invariantes, etc.

Objetivos do Curso

- Como **modelar problemas** usando fórmulas lógicas.

Problema das N-Rainhas, modelar o comportamento de programs como seus invariantes, etc.

- Saber como construir provas formais de teoremas.

$$a, b, a \wedge b \Rightarrow c \vdash c$$

Objetivos do Curso

- Como **modelar problemas** usando fórmulas lógicas.

Problema das N-Rainhas, modelar o comportamento de programs como seus invariantes, etc.

- Saber como construir provas formais de teoremas.

$$a, b, a \wedge b \Rightarrow c \vdash c$$

- Saber como programar na linguagem de programação **Prolog**.

```
irmão(X, Y) :- pai(Z, X), pai(Z, Y).
```

Ementa do Curso

Ementa do Curso

- Revisão de conceitos elementares da matemática: relações, funções, árvores, provas por indução e por recursão.

Ementa do Curso

- Revisão de **conceitos elementares da matemática**: relações, funções, árvores, provas por indução e por recursão.
- **Lógica Proposicional**: Sintaxe e Semântica; Cálculo de Sequentes, teoremas de corretude e completude.

Ementa do Curso

- Revisão de **conceitos elementares da matemática**: relações, funções, árvores, provas por indução e por recursão.
- **Lógica Proposicional**: Sintaxe e Semântica; Cálculo de Sequentes, teoremas de corretude e completude.
- **Lógica de Primeira Ordem**: Sintaxe e Semântica; Cálculo de Sequentes, teoremas de corretude e completude.

Ementa do Curso

- Revisão de **conceitos elementares da matemática**: relações, funções, árvores, provas por indução e por recursão.
- **Lógica Proposicional**: Sintaxe e Semântica; Cálculo de Sequentes, teoremas de corretude e completude.
- **Lógica de Primeira Ordem**: Sintaxe e Semântica; Cálculo de Sequentes, teoremas de corretude e completude.
- **Linguagem de Programação Prolog**: Substituição, Unificação, Resolução e modelagem de sistemas.

Bibliografia do Curso

- Jean Gallier: Logic in Computer Science – **Disponível online.**
- Herbert Enderton. A Mathematical Introduction to Logic, Second Edition.
- Jair Minoro Abe, Alexandre Scalzitti, João Inácio da Silva Filho. Introdução à Lógica Para a Ciência da Computação. Editora Arte e Ciência.

Método de Avaliação

- Duas listas de exercícios – (20%)
- Uma prova no fim do semestre – (40 %)
- Projetos/Seminários: grupos de até 5 alunos – (40 %)

Média Ponderada

$$\text{Nota-final} = (2 * \text{Surpresa} + 4 * \text{Final} + 4 * \text{Seminários}) / 10$$

Datas (passíveis de mudança)

Datas (passíveis de mudança)

Duas listas de exercícios – (20%)

- Entrega da Primeira Lista de Exercícios: **7 de fevereiro**
- Entrega da Segunda Lista de Exercícios: **19 de março**

Datas (passíveis de mudança)

Duas listas de exercícios – (20%)

- Entrega da Primeira Lista de Exercícios: **7 de fevereiro**
- Entrega da Segunda Lista de Exercícios: **19 de março**

Uma Prova no Final do Semestre – (40 %)

Datas (passíveis de mudança)

Duas listas de exercícios – (20%)

- Entrega da Primeira Lista de Exercícios: **7 de fevereiro**
- Entrega da Segunda Lista de Exercícios: **19 de março**

Uma Prova no Final do Semestre – (40 %)

Projetos/Seminários: **grupos de até 6 alunos** – (40 %)

- Os seminários serão realizados durante **as duas ou três semanas** do semestre.

Possíveis Temas para o Seminário

- SAT Solvers
- Verificação de Circuitos Lógicos
- Protocolos de Segurança
- Datalog
- Network Datalog
- Programação em Lógica Linear
- Constraint Logic Programming
- Answer-Set Programming
- Lambda-Prolog
- Lógicas Temporais
- Lógicas Módaes
- Isomorfismo Curry-Howard

Conflitos

Dia **20 de dezembro**, eu estarei fora para uma colaboração. Portanto, **não haverá aula de Lógica Aplicada à Computação neste dia.**

Por favor discutam qual o melhor horário para **repor esta aula.**

Conflitos

Dia **20 de dezembro**, eu estarei fora para uma colaboração. Portanto, **não haverá aula de Lógica Aplicada à Computação neste dia.**

Por favor discutam qual o melhor horário para **repor esta aula.**

Eu gostaria que você elegessem um **representante da turma.** Para discutir horários de provas, etc.

Por que estudar lógica?

Por que estudar lógica?

No seu Cotidiano

Por que estudar lógica?

No seu Cotidiano

Melhorar a sua argumentação: qual a lógica do seu argumento? (Vejam os cursos de Metodologia de Pesquisa e Pesquisa Aplicada à Computação.)

Por que estudar lógica?

No seu Cotidiano

Melhorar a sua argumentação: qual a lógica do seu argumento? (Vejam os cursos de Metodologia de Pesquisa e Pesquisa Aplicada à Computação.)

Melhorar a sua capacidade de expor idéias: por que devo te acreditar?

Por que estudar lógica?

No seu Cotidiano

Melhorar a sua argumentação: qual a lógica do seu argumento? (Vejam os cursos de Metodologia de Pesquisa e Pesquisa Aplicada à Computação.)

Melhorar a sua capacidade de expor idéias: por que devo te acreditar?

Identificar argumentos falsos: argumentos circulares usado por muitos políticos.

Por que estudar lógica aplicada a computação?

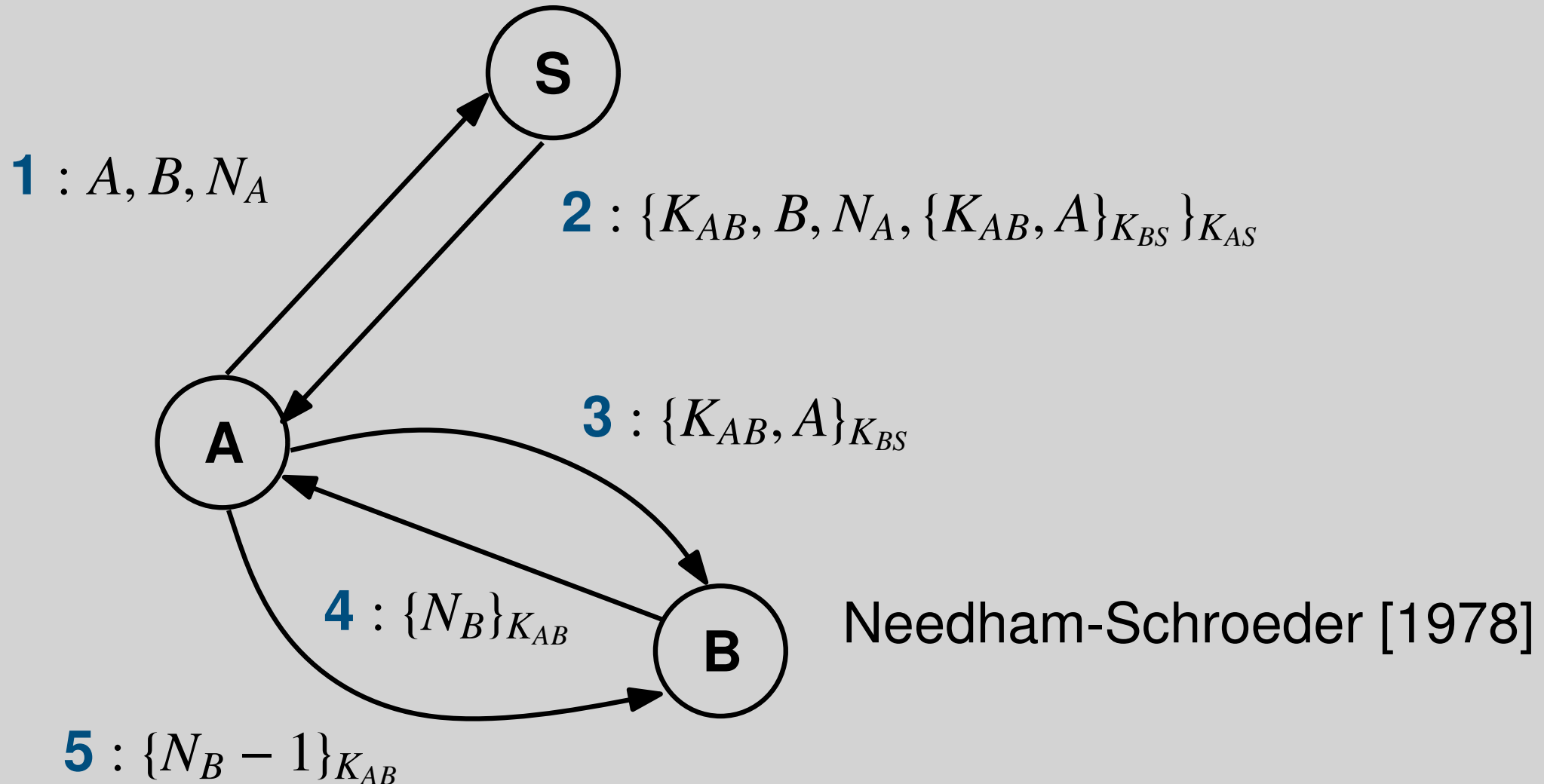


Caso do Foguete Ariane 5

On June 4, 1996 an unmanned **Ariane 5 rocket** launched by the European Space Agency exploded just forty seconds after its lift-off from Kourou, French Guiana. The rocket was on its first voyage, **after a decade of development costing USD 7 billion**. The destroyed rocket and **its cargo were valued at USD 500 million**. A board of inquiry investigated the causes of the explosion and in two weeks issued a report. It turned out that **the cause of the failure was a software error in the inertial reference system**. Specifically a 64 bit floating point **number** relating to the horizontal velocity of the rocket with respect to the platform **was converted to a 16 bit signed integer**. The number was larger than 32,767, the largest integer storeable in a 16 bit signed integer, and thus the conversion failed.

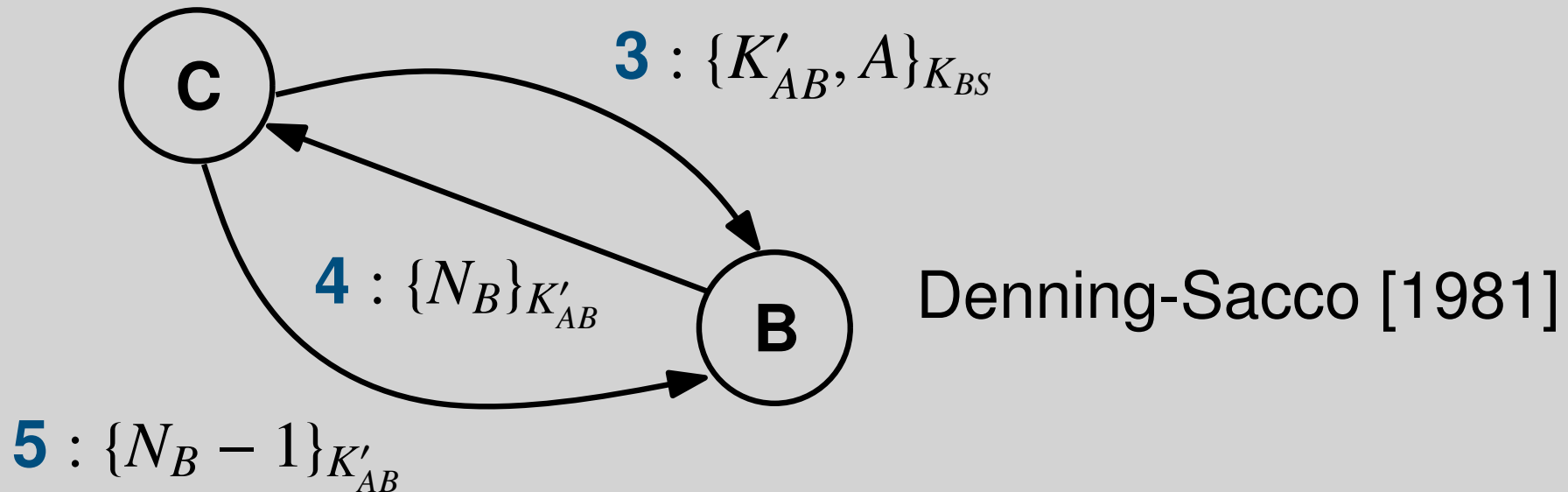
Por que estudar lógica aplicada a computação?

Protocolos de Segurança



Por que estudar lógica aplicada a computação?

Protocolos de Segurança



Um invasor **C** pode lembrar de uma mensagem antiga contendo a chave K'_{AB}

Recentemente, um ataque contra os cartões PIN & Chip foi descoberto!

Por que estudar lógica aplicada a computação?

Caso do Chip da IBM



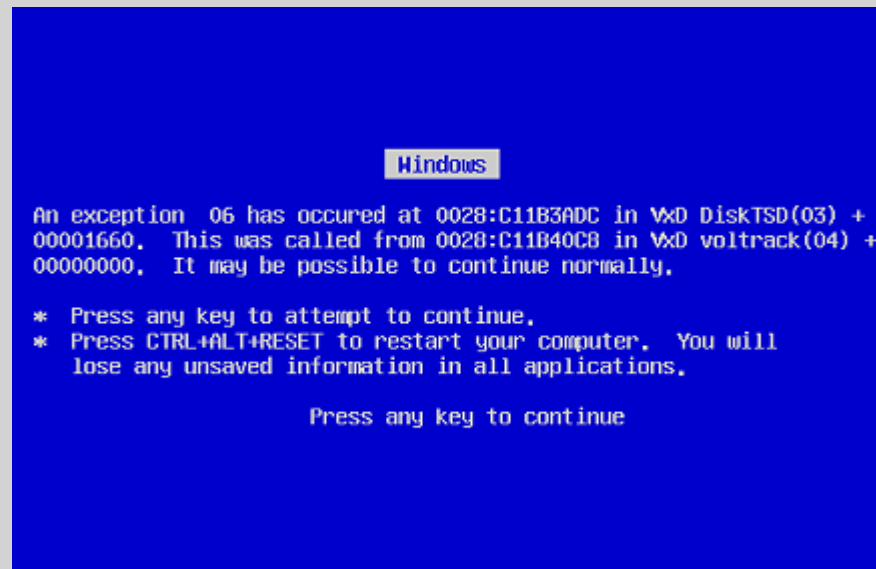
The Pentium FDIV bug was a bug in the Intel P5 Pentium floating point unit (FPU). Certain floating point division operations performed with these processors produced **incorrect results**.

...

Intel ultimately recalled the defective processors.

Por que estudar lógica aplicada a computação?

Tela Azul do Windows (Blue-Screen-of-Death)



Static Driver Verifier

Por que estudar lógica aplicada a computação?

Java Modelling Language (JML)

```
/*@ requires a != null
   @ && (forall int i; 0 < i && i < a.length;
   @     a[i-1] < a[i] ;
   @*/;
```

```
int binarySearch(int[] a, int x) {
    ...
}
```

Por que estudar lógica aplicada a computação?

Compiladores (Projeto CompCert)

We created a tool that generates **random C programs**, and then spent **two and a half years using it to find compiler bugs. So far, we have reported more than 325 previously unknown bugs** to compiler developers. Moreover, **every compiler that we tested has been found to crash** and also to silently generate wrong code when presented with valid inputs. (PLDI 2011)

Por que estudar lógica aplicada a computação?

Muitas outras aplicações:

Por que estudar lógica aplicada a computação?

Muitas outras aplicações:

- **Bases de dados** – a linguagem SQL, apesar de usar uma outra sintaxe, é basicamente **lógica de primeira ordem**.

Por que estudar lógica aplicada a computação?

Muitas outras aplicações:

- **Bases de dados** – a linguagem SQL, apesar de usar uma outra sintaxe, é basicamente **lógica de primeira ordem**.
- **Inteligência Artificial** – O conhecimento **adquirido durante anos** de experiência pode ser expresso como **fórmulas lógicas**.

Por que estudar lógica aplicada a computação?

Muitas outras aplicações:

- **Bases de dados** – a linguagem SQL, apesar de usar uma outra sintaxe, é basicamente **lógica de primeira ordem**.
- **Inteligência Artificial** – O conhecimento **adquirido durante anos** de experiência pode ser expresso como **fórmulas lógicas**.
- **Complexidade** – Alguns problemas, chamados de NP-completos, **são difíceis para resolver, mesmo tendo a máquina mais potente do mundo**. O estudo de tais problemas envolve normalmente o estudo de lógica.

Um pequeno teste de lógica

Um **escravo**, por desobediência foi aprisionado e condenado à morte. O **senhor** resolveu ser gentil com escravo e disse-lhe: Deixo-te escolher a maneira como vai morrer. Se disseres uma frase **verdadeira**, serás decapitado. Se disseres uma frase **falsa**, serás enforcado.

O **escravo** disse – “**Vou morrer enforcado.**” Com esta afirmação o escravo escapou da morte. Por que?

Um pouco sobre Vivek Nigam

- **1999 – 2003:** Graduação em Engenharia Eletrônica no ITA;
- **2004** (8 meses): Engenheiro na SMAR Equipamentos Eletrônicos;
- **2004 – 2006:** Mestrado duplo em **Lógica Computacional** na Alemanha e em Portugal;
- **2006 – 2009:** Doutorado em **Lógica Computacional** na École Polytechnique na França ;
- **2009 – 2010:** Pós-Doutorado em **Sistemas Collaborativos** na UPENN nos EUA;
- **2010 – 2012:** Pós-Doutorado em **Linguagens de Programação** na LMU na Alemanha;

Um pouco sobre Vivek Nigam

- **Teoria da Computação:** Sistemas de provas, Frameworks Lógicos
- **Segurança de Protocolos:** Modelos Formais para a especificação e verificação de protocolos de segurança;
- **Informática na Saúde:** Aplicação de Métodos Formais em Investigações Clínicas.
- **Semântica de Linguagem de Programação:** Modelos que justifiquem a equivalência de dois programas.

Alunos de Iniciação Científica são muito bem vindos!

Muitos Projetos envolvendo uma equipe de renome internacional

Pesquisadores

- Prof. Andre Scedrov da Universidade da Pennsylvania (Filadélfia): **Mais de 60 publicações A1 – B1.**
- Prof. Carolyn Talcott da SRI International (São Francisco): **Mais de 50 publicações A1 – B1.**
- Prof. Max Kanovich da Queen Mary University (Londres): **Mais de 20 publicações A1 – B1.**

Grupo Internacional e Produção de Alta Qualidade

Publicações envolvendo Vivek Nigam – Últimos dois anos:

- Revista Information and Computation – A2 (Aceito).
- Revista Transactions on Networking – A1 (Aceito).
- Conferência LICS – A2 (2012).
- Conferência RTA – B1 (2012).
- Conferência IHI – B4 (2012).
- Conferência LSFA (ENTCS) – B1 (2011).
- Conferência PPDP – B1 (2011).
- Revista COMLAN – B3 (2011).