

Bacharelado em Ciência da Computação – CEFET/RJ
GCC1734 – Inteligência Artificial
Informações administrativas do curso

Prof. Eduardo Bezerra
`ebezerra@cefet-rj.br`

1 Canais oficiais e acesso a materiais

Este documento resume os aspectos administrativos do curso **GCC1734 – Inteligência Artificial**. Ao longo do período, utilizaremos três canais principais, cada um com um propósito distinto e complementar.

A página do curso é o ponto central para consulta rápida de informações. Nela você encontrará o plano de curso (conteúdo programático, cronograma e organização geral) e o material das aulas (slides e notas). O endereço é:

<https://eic.cefet-rj.br/~ebezerra/gcc1734>

O repositório GitHub do curso será utilizado para disponibilizar exemplos de código, conjuntos de dados e código-base utilizado nos trabalhos práticos. O endereço é:

<https://github.com/AILAB-CEFET-RJ/gcc1734>

A plataforma Microsoft Teams será o ambiente operacional do curso. Por ela serão publicados os enunciados das tarefas (trabalhos práticos e listas), mensagens e comunicados para a turma (dicas, avisos e resultados), bem como a submissão das entregas. Sempre que houver material complementar, ele também poderá ser disponibilizado por esse canal. Assim, considere o Teams como o local onde “as coisas acontecem” (prazos, entregas e recados), enquanto a página e o GitHub concentram materiais e código.

2 Material para estudo

O livro-texto do curso é:

Russell & Norvig, *Artificial Intelligence: A Modern Approach*, 4ª edição.

O material didático do curso será organizado principalmente na forma de notas de aula. As notas constituem o texto central do curso. Elas contêm:

- Definições formais;
- Exemplos comentados;
- Demonstrações e modelagens completas;
- Perguntas conceituais distribuídas ao longo do texto.

O estudante deve ler as notas **antes** da aula correspondente. A aula presencial não será uma repetição do texto, mas um espaço para:

- Discussão de dúvidas;
- Análise de formulações incorretas;
- Resolução de problemas;
- Simulação de execução de algoritmos.

Perguntas e Entry Tickets

Ao longo das notas, há blocos identificados como **Pergunta**. Essas perguntas não são decorativas. Elas fazem parte do processo de estudo.

Antes de cada aula, será indicado um subconjunto dessas perguntas como **entry ticket**. O estudante deverá:

- Responder às perguntas por escrito;
- Entregar as respostas;
- Estar preparado para defender seu raciocínio oralmente.

O objetivo do entry ticket é simples: garantir que a leitura foi feita com atenção e que o estudante já tentou formular respostas antes do encontro presencial.

A aula parte do pressuposto de que o material foi lido. Sem essa preparação prévia, o aproveitamento da aula é comprometido.

Responsabilidade do Estudante

Ler não é o mesmo que entender. Responder às perguntas força o estudante a:

- Traduzir definições formais em suas próprias palavras;
- Aplicar conceitos a exemplos concretos;
- Identificar lacunas no próprio entendimento.

A dinâmica do curso depende dessa preparação. O modelo adotado é o de *sala de aula invertida*: o primeiro contato com o conteúdo ocorre individualmente; a aula é o espaço de consolidação.

3 Dinâmica das aulas

A maioria das aulas seguirá o modelo de *sala de aula invertida*: o contato com o conteúdo novo ocorre *antes* da aula presencial, por meio da leitura do material disponibilizado previamente. O tempo em sala será reservado para resolução de problemas, discussão de estratégias, análise de erros frequentes e aprofundamento dos conceitos, e não para exposição inicial do conteúdo.

- *Leitura prévia.* Antes de cada aula, o estudante deve realizar a leitura do material indicado. Essa leitura não é opcional: ela é condição para o aproveitamento da aula presencial e integra formalmente o processo de avaliação.
- *Entry-tickets.*¹ Como comprovação de leitura e condição de participação na aula, cada estudante deverá enviar, via Microsoft Teams, o *entry-ticket*, composto por questões baseadas no material previamente indicado. O entry-ticket será disponibilizado com antecedência, juntamente com o material de leitura.

As respostas deverão ser enviadas até as 12h (meio-dia) do dia da aula correspondente. Não haverá entrega em atraso. A correção será objetiva (realizado ou não realizado). Em cada bimestre, as duas menores notas desse componente poderão ser desconsideradas para fins de cálculo da média.

- *Atividades em sala.* O tempo presencial será dedicado a exercícios desenvolvidos em duplas ou pequenos grupos, que poderão ser recolhidos ao final da aula. A avaliação considerará o envolvimento efetivo na resolução das atividades propostas.
- *Presença.* As aulas são presenciais. A chamada será realizada no início de cada encontro e apenas nesse momento. De acordo com as normas institucionais, o estudante poderá ter até 25% de faltas.
- *Dúvidas e comunicações.* O Microsoft Teams deve ser utilizado como canal preferencial para questões administrativas e dúvidas sobre o conteúdo, de modo que as respostas fiquem registradas e acessíveis a toda a turma.

4 Estrutura de avaliação

A avaliação do curso será composta por participação em aula, trabalhos práticos e dois testes de múltipla escolha. Havendo necessidade e possibilidade, poderá existir avaliação final, conforme as regras institucionais e o contexto do período.

A nota de cada bimestre será calculada usando pesos, conforme descrito a seguir.

¹O entry-ticket (bilhete de entrada) na sala de aula invertida é uma ferramenta de avaliação diagnóstica rápida, aplicada no início da aula, para verificar se os alunos estudaram o material prévio (vídeos, textos) e identificar dúvidas. Ele promove o protagonismo, a autonomia e prepara para atividades ativas.

- No **1º bimestre**, a participação em aula (incluindo aqui entrega dos entry-tickets) compõe **25%** da nota, os trabalhos **T1 e T2** compõem **35%**, e o **Teste 1** compõe os **40%** restantes.
- No **2º bimestre**, a participação em aula (incluindo aqui entrega dos entry-tickets) compõe **25%** da nota, os trabalhos **T3 e T4** compõem **35%**, e o **Teste 2** compõe os **40%** restantes.

O entry-ticket somente será contabilizado para fins de avaliação se o estudante estiver presente na aula correspondente. A ausência à aula implica desconsideração do entry-ticket enviado.

Os prazos de entrega dos trabalhos práticos estarão definidos no plano de curso. É responsabilidade do estudante planejar-se para cumpri-los. Há tolerância para atraso, com penalidade automática: a cada dia corrido de atraso, será descontado 20% da nota máxima do trabalho. Após cinco dias de atraso, a nota atribuída será zero. Será considerado o horário de submissão registrado no Microsoft Teams. Frações de dia serão contabilizadas como um dia completo.

A presença nos dois testes é obrigatória. Será oferecida apenas uma única prova de segunda chamada ao final do período letivo. A prova de segunda chamada substitui, no máximo, um dos testes regulares. Assim, mesmo que o estudante esteja ausente nos dois testes, poderá repor apenas um deles por meio da segunda chamada. A prova de segunda chamada será discursiva e abrangerá todo o conteúdo programático do curso, independentemente de qual teste não tenha sido realizado. Recomenda-se, portanto, que o estudante organize sua agenda para realizar os testes nas datas regulares, evitando a necessidade de avaliação substitutiva cumulativa.

A média semestral será calculada como a média aritmética simples das médias obtidas no primeiro e no segundo bimestres. Ou seja, se M_1 e M_2 representam as médias do 1º e 2º bimestres, respectivamente, então a média semestral será dada por $(M_1 + M_2)/2$.

O exame final deverá ser obrigatoriamente realizado pelos estudantes que não obtiverem média semestral superior a 7,0 após o cálculo das notas dos dois bimestres, desde que tenham alcançado média semestral superior a 3,0. Estudantes com média semestral inferior a 3,0 não terão direito à realização do exame final. O exame final abrangerá todo o conteúdo programático do curso e será aplicado conforme calendário institucional.

5 Trabalhos práticos

Os trabalhos práticos envolvem programação em **Python 3.11**. É responsabilidade do estudante garantir familiaridade com os fundamentos da linguagem. Para revisão de aspectos básicos, recomenda-se o livro a seguir. Os capítulos 1–11 cobrem os fundamentos necessários.

<https://greenteapress.com/wp/learning-with-python/>

Cada trabalho deve ser acompanhado de um vídeo explicativo obrigatório, com duração aproximada entre 20 e 30 minutos. No vídeo, o estudante deve:

- Explicar o código implementado;
- Executar cada parte solicitada;
- Analisar os resultados obtidos.

O vídeo deve ser disponibilizado por meio de link privado (por exemplo, canal privado no YouTube, ou gravado no MS Teams institucional). O link deve ser incluído na submissão. O vídeo é parte integrante da avaliação. Trabalhos sem vídeo não serão considerados completos.

Inconsistências entre o código entregue e a explicação apresentada irão impactar negativamente a nota. O foco da avaliação será sempre a compreensão demonstrada pelo estudante.

6 Conteúdo programático em alto nível

O curso está organizado em três eixos principais.

1. *Fundamentos clássicos de IA (IA simbólica e busca)*. Agentes inteligentes e ambientes; Formulação de problemas de busca; Busca clássica (cega e heurística); Busca adversarial (minimax e poda alfa-beta); Busca com restrições; Agentes lógicos e representação de conhecimento.
2. *Raciocínio sob incerteza e aprendizado*. Redes Bayesianas; Inferência probabilística; Aprendizado por reforço; Fundamentos de aprendizado de máquina.
3. *Agentes baseados em IA generativa*. Modelos de linguagem e geração de texto; Arquiteturas de agentes baseados em LLMs; Planejamento iterativo (ReAct, Plan-and-Act, Reflexion); Uso de ferramentas (tool calling); Integração com bases de conhecimento (RAG); Limitações, avaliação e aspectos éticos.

Para acompanhar bem o curso, espera-se domínio de:

- Estruturas de dados (listas, pilhas, filas, árvores);
- Técnicas de programação (especialmente recursividade);
- Conceitos básicos de matemática (probabilidade, relações de recorrência);
- Noções elementares de complexidade de algoritmos.

7 Como estudar e como usar IA de forma responsável

Uma estratégia recomendada para estudo neste curso é aplicar a **Técnica de Feynman**: explicar um conceito com suas próprias palavras, identificar lacunas na explicação, revisar o material e reconstruir a explicação de forma mais simples e precisa.

O uso de modelos generativos pode ser produtivo quando empregado exatamente para operacionalizar esse processo. Em vez de solicitar respostas prontas, o estudante deve utilizar a IA como tutor que questiona, identifica lacunas e estimula explicações próprias.

O prompt abaixo foi elaborado com esse objetivo. Para leitura introdutória sobre a técnica, veja:

<https://collegeinfo geek.com/feynman-technique/>

8 Uso de Modelos Generativos (IA)

Ferramentas baseadas em modelos generativos (ChatGPT, Claude, Gemini, Copilot e similares) podem ser utilizadas como apoio. Contudo, este é um curso de **implementação de algoritmos**. Logo, o uso inadequado dessas ferramentas compromete diretamente o aprendizado.

Princípio central

Você é integralmente responsável por tudo o que submete.

Se não consegue explicar linha por linha do seu código, você não domina o conteúdo.

Usos aceitáveis

É aceitável utilizar IA para:

- Esclarecer dúvidas conceituais;
- Revisar sintaxe Python;
- Discutir complexidade assintótica;
- Explorar variações de algoritmos.

Usos inadequados

Não é aceitável:

- Gerar soluções completas e apenas adaptá-las superficialmente;
- Utilizar IA para redigir integralmente as soluções das entry-tickets;
- Submeter código que você não compreende integralmente;
- Utilizar IA para contornar a implementação de partes centrais do algoritmo.

Transparência

Se houver uso substantivo de IA, inclua ao final do notebook:

- Qual ferramenta foi utilizada;
- Para qual finalidade;
- Quais partes foram auxiliadas.

Prompt recomendado

Recomenda-se utilizar IA como tutor socrático, não como resolvidor automático.

Prompt recomendado (IA como tutor de IA)

Você é um tutor de Inteligência Artificial em nível de graduação. Não resolva integralmente os exercícios para mim.

Quando eu apresentar um algoritmo ou código:

1. Peça para eu explicar a ideia do algoritmo com minhas palavras.
2. Pergunte qual é o estado, qual é a função sucessora e qual é o critério de parada (se for busca).
3. Pergunte qual é a função de avaliação (se for jogo).
4. Pergunte sobre complexidade temporal e espacial.
5. Aponte lacunas conceituais.
6. Só depois sugira melhorias.

Se eu pedir código, forneça apenas esqueleto parcial e peça que eu complete partes críticas. Priorize raciocínio. Evite respostas finais prontas.