Lógica

Equivalência de Negação Lógica

Lógica vs. Programação

- O que é um software de qualidade?
- Para desenvolver um software de qualidade, você considera lógica como característica mais importante?
- Além de lógica, quais características um projeto de software precisa ter ser considerado de qualidade?

Lógica vs. Programação

1. Clareza e Legibilidade

- Nomes significativos: use nomes claros para variáveis, funções e classes (ex: `calcula_media`, `nome_aluno`).
- Evite abreviações excessivas: `notaFinal` é melhor que `nf`.
- Indentação consistente: essencial em Python. Use 4 espaços por nível.
- Comentários úteis: explique por que algo é feito, não o que (que já está no código).

2. Organização do Código

- Funções curtas e coesas: cada função deve ter uma única responsabilidade.
- Modularização: divida o código em arquivos e funções reutilizáveis.
- Evite código duplicado: use funções ou laços quando necessário.

3. Testabilidade e Validação

- Teste o código com dados variados: entrada válida, inválida, vazia etc.
- Use `try-except` para tratamento de erros (em níveis mais intermediários).
- Desconfie de código que "só funciona com os dados certos".

Lógica vs. Programação

4. Lógica e Estrutura

- Use estruturas de controle com clareza: evite ifs aninhados demais.
- Se algo parece complicado demais, talvez possa ser feito de forma mais simples.

5. Padrões e Convenções

- Siga convenções de nomes da linguagem: ex., em Python, nomes de função e variáveis em snake_case.
- Documente funções com docstrings

6. Princípios de Design de Código (ou padrões de projeto)

- Ex: SOLID (orientado a objetos)

Lógica vs. Programação

7. Qualidade e Manutenibilidade:

- Escreva testes automatizados (unitários, de integração).
- Refatore com frequência: melhorar o código sem mudar seu comportamento.
- Use ferramentas de lint e formatação automática (ex: flake8, black, prettier).
- Evite código com muitos efeitos colaterais (ex: funções que alteram o estado de forma inesperada).

8. Estrutura e Arquitetura

- Organize seu projeto em camadas ou módulos separados:
- Exemplo: dados/, negocio/, interface/
- Utilize injeção de dependência onde fizer sentido.
- Evite acoplamento forte entre módulos — prefira composição a herança, quando possível.

Lógica vs. Programação

9. Documentação e Comunicação

- Comente intenções, e linhas não óbvias
- Use README.md e docstrings para orientar outros desenvolvedores.
- Documente interfaces públicas, APIs e contratos.

10. Segurança e Confiabilidade

- Valide toda entrada de dados (input sanitization).
- Evite hardcode de dados sensíveis (ex: senhas no código).
- Use tratamento de exceções com responsabilidade: nunca capture exceções genéricas sem motivo.

11. Desempenho e Escalabilidade

- Escolha algoritmos e estruturas de dados adequadas ao problema.
- Avalie complexidade de cada funcionalidade e refatore se preciso.

Lógica vs. Programação

12. Boas Práticas de Trabalho em Equipe

- Use controle de versão (Git) corretamente: commits pequenos e com mensagens claras.
- Faça code reviews e aceite feedback de forma colaborativa.
- Escreva código pensando no próximo desenvolvedor.

13. Cuidados com equipes distribuídas

- Diferenças na padronização ou na forma de programar
- Horários de trabalhos distintos
- Diferenças culturais

1. Introdução a Negação de Proposições

- Antes de adentrarmos no assunto, é importante esclarecer que não se deve confundir equivalência com negação.
- Ao se construir negação de uma proposição, constrói-se uma nova proposição com valores lógicos sempre opostos aos da proposição original.
- Veremos mais adiante, por exemplo, que a negação de $p \wedge q$ é $\sim p \vee \sim q$. Nesse caso, não podemos dizer que $p \wedge q$ é equivalente a $\sim p \vee \sim q$, podemos dizer que $\sim(p \wedge q)$ é equivalente a $\sim p \vee \sim q$, ou seja, $\sim(p \wedge q) \equiv \sim p \vee \sim q$.
- Feitas estas considerações iniciais, vamos analisar as equivalências provenientes da negação de proposições. Existem muitas maneiras de se expressar uma negação. A seguir serão apresentadas as formas mais comuns.

2. Equivalências da Negação de Proposições

1. Dupla negação da proposição simples

- Um resultado importante que pode ser obtido da tabela verdade é que a negação da negação de p sempre tem valor lógico igual a proposição p , ou seja, é equivalente a p :
 $\sim(\sim p) \equiv p$
- Como exemplo, a dupla negação "Não é verdade que [o Flamengo não é um time fraco]" é equivalente a "Flamengo é um time fraco".
- A prova dessa equivalência corresponde à tabela-verdade abaixo.

p	$\sim p$	$\sim(\sim p)$
V	F	V
F	V	F

2. Equivalências da Negação de Proposições

2. Negação da conjunção

- Para realizar a negação conjunção $p \wedge q$, deve-se seguir o seguinte procedimento: Nega-se ambas as parcelas da conjunção e troca-se a conjunção ($\wedge$) pela disjunção inclusiva ($\vee$).
- Como resultado, podemos escrever que a negação de $p \wedge q$, também conhecida por $\sim(p \wedge q)$, é equivalente a $\sim p \vee \sim q$: **$\sim(p \wedge q) \equiv \sim p \vee \sim q$**
- Como exemplo, sejam as proposições:

p : "Comi lasanha."

q : "Bebi refrigerante."

- A conjunção dessas duas proposições pode ser escrita por: $p \wedge q$: "[Comi lasanha] e [bebi refrigerante]."
- A negação dessa frase é: $\sim(p \wedge q) \equiv \sim p \vee \sim q$: "[Não comi lasanha] ou [não bebi refrigerante]."

2. Equivalências da Negação de Proposições

A tabela verdade que comprova a equivalência:

p	q	$\sim p$	$\sim q$	$p \wedge q$	$\sim(p \wedge q)$	$\sim p \vee \sim q$
V	V	F	F	V	F	F
V	F	F	V	F	V	V
F	V	V	F	F	V	V
F	F	V	V	F	V	V

2. Equivalências da Negação de Proposições

3. Negação da disjunção inclusiva

- De modo semelhante à negação da conjunção, para negarmos a disjunção inclusiva $p \vee q$, devemos seguir o seguinte procedimento: Negam-se ambas as parcelas da disjunção inclusiva e troca-se a disjunção inclusiva ($\vee$) pela conjunção ($\wedge$).
- Como resultado disso, podemos escrever que a negação de $p \vee q$, também conhecida por $\sim(p \vee q)$, é equivalente a $\sim p \wedge \sim q$: **$\sim(p \vee q) \equiv \sim p \wedge \sim q$**

2. Equivalências da Negação de Proposições

- Vejamos um exemplo $p \vee q$: "[Comi lasanha] ou [bebi refrigerante]."
- A negação dessa frase seria: $\sim (p \vee q) \equiv \sim p \wedge \sim q$: "[Não comi lasanha] e [não bebi refrigerante]."
- Essa equivalência pode ser facilmente constatada na tabela-verdade:

p	q	$\sim p$	$\sim q$	$p \vee q$	$\sim(p \vee q)$	$\sim p \wedge \sim q$
V	V	F	F	V	F	F
V	F	F	V	V	F	F
F	V	V	F	V	F	F
F	F	V	V	F	V	V

2. Equivalências da Negação de Proposições

- Exemplo:

(SSP AM/2022) Considere a afirmação: “Hoje é sexta-feira e amanhã não trabalharei”.

A negação lógica dessa sentença é

- a) Hoje não é sexta-feira ou amanhã trabalharei.
- b) Hoje não é sexta-feira, então amanhã trabalharei.
- c) Hoje é sexta-feira e amanhã trabalharei.
- d) Hoje é sexta-feira ou amanhã não trabalharei.

2. Equivalências da Negação de Proposições

4. Negação da condicional

- A negação de $p \rightarrow q$ é realizada por meio da seguinte equivalência: $\sim (p \rightarrow q) \equiv p \wedge \sim q$
- A negação da condicional é realizada do seguinte modo: Mantém-se o primeiro termo, troca-se a condicional ($\rightarrow$) pela conjunção ($\wedge$) e nega-se o segundo termo.
- Como exemplo, considere a condicional: $p \rightarrow q$: "Se [eu beber], então [dou gargalhadas]."
- A negação dessa expressão pode ser escrita como: $\sim (p \rightarrow q) \equiv p \wedge \sim q$: "[Eu bebo] e [não dou gargalhadas]."

2. Equivalências da Negação de Proposições

- A tabela verdade abaixo comprova a negação:

p	q	$\sim q$	$p \wedge \sim q$	$p \rightarrow q$	$\sim(p \rightarrow q)$
V	V	F	F	V	F
V	F	V	V	F	V
F	V	F	F	V	F
F	F	V	F	V	F

2. Equivalências da Negação de Proposições

(EPE/2022) A negação da afirmativa “Se João vai ao jogo, então o Flamengo perde” é

- a) João vai ao jogo e o Flamengo não perde.
- b) João não vai ao jogo e o Flamengo perde.
- c) João não vai ao jogo e o Flamengo não perde.
- d) Se João não vai ao jogo, então o Flamengo perde.
- e) Se João não vai ao jogo, então o Flamengo não perde.

2. Equivalências da Negação de Proposições

- Sejam as proposições simples:

j: "João vai ao jogo."

f: "O Flamengo perde."

- A sentença original pode ser descrita por $j \rightarrow f$: "Se [João vai ao jogo], então [o Flamengo perde]".
- Para realizar a negação de uma condicional, usa-se a equivalência $\sim(p \rightarrow q) \equiv p \wedge \sim q$.
- Logo, a negação pode ser descrita por: $j \wedge \sim f$: "[João vai ao jogo] e [o Flamengo não perde]."
- Letra A

2. Equivalências da Negação de Proposições

5. Negação da disjunção exclusiva

- A negação da disjunção exclusiva mais comum é equivalente a: $\sim(p \vee q) \equiv p \leftrightarrow q$
- Como exemplo, considere a disjunção exclusiva: $p \vee q$: "Ou jogo bola, ou jogo sinuca."
- A negação dessa expressão é dada pelo bicondicional abaixo:

$\sim(p \vee q) \equiv p \leftrightarrow q$: "Jogo bola se e somente se jogo sinuca."

2. Equivalências da Negação de Proposições

- Com a tabela verdade vamos demonstrar que são equivalentes $\sim(p \vee q)$ e $p \leftrightarrow q$:

p	q	$p \vee q$	$p \leftrightarrow q$
V	V	F	V
V	F	V	F
F	V	V	F
F	F	F	V

2. Equivalências da Negação de Proposições

6. Negação da bicondicional

- São quatro as maneiras mais comuns de se negar a bicondicional. A primeira forma que vamos apresentar é compatível com a disjunção exclusiva: $\sim(p \leftrightarrow q) \equiv p \vee q$

p	q	$p \leftrightarrow q$	$\sim(p \leftrightarrow q)$	$p \vee q$
V	V	V	F	F
V	F	F	V	V
F	V	F	V	V
F	F	V	F	F

2. Equivalências da Negação de Proposições

- A imagem abaixo resume as quatro principais formas de negar uma bicondicional:

$$\sim (p \leftrightarrow q) \equiv p \vee \sim q$$

$$\sim (p \leftrightarrow q) \equiv (\sim p) \leftrightarrow q$$

$$\sim (p \leftrightarrow q) \equiv p \leftrightarrow (\sim q)$$

$$\sim (p \leftrightarrow q) \equiv (p \wedge \sim q) \vee (q \wedge \sim p)$$

2. Equivalências da Negação de Proposições

(Pref. Vila Lângaro/2019) A negação da proposição “João passa no concurso público se e somente se João estuda” é:

- a) João não passa no concurso público se e somente se João não estudou.
- b) João não passa no concurso público e João não estudou.
- c) João passa no concurso público e João estuda.
- d) Ou João passa no concurso público ou João estuda.
- e) Se João passa no concurso público, então João estuda.

2. Equivalências da Negação de Proposições

- A proposição composta original é uma bicondicional $p \leftrightarrow q$ cujos termos são:
p: " João passa no concurso público."
q: " João estuda."
- É necessário testar as 4 formas de negação para verificar qual é a compatível com as alternativas. Neste caso, a forma $\sim (p \leftrightarrow q) \equiv p \vee q$ está na letra D:
- $p \vee q$: " Ou [João passa no concurso público] ou [João estuda]."
- **CUIDADO** especial com a letra A, que apresenta uma equivalência do bicondicional, não uma negação: $p \leftrightarrow q \equiv (\sim p) \leftrightarrow (\sim q)$

2. Equivalências da Negação de Proposições

7. Negações da conjunção para a forma condicional

- Existem duas maneiras de se negar a conjunção de modo que ela adquira a forma condicional, são elas:
- $\sim(p \wedge q) \equiv p \rightarrow \sim q$
- $\sim(p \wedge q) \equiv q \rightarrow \sim p$

2. Equivalências da Negação de Proposições

(MRE/2016) Considere a sentença "Corro e não fico cansado". Uma sentença logicamente equivalente à negação da sentença dada é:

- a) Se corro então fico cansado.
- b) Se não corro então não fico cansado.
- c) Não corro e fico cansado.
- d) Corro e fico cansado.
- e) Não corro ou não fico cansado.

2. Equivalências da Negação de Proposições

- Sejam as proposições simples:

c: "Corro."

f: "Fico cansado."

- O enunciado apresenta a sentença $c \wedge \sim f$ e pede a negação $\sim(c \wedge \sim f)$.
- Observe que o enunciado requer a negação de uma conjunção e as alternativas apresentam condicionais ("se...então"), conjunções ("e") e disjunção inclusiva ("ou").
Conhecemos três maneiras de se negar uma conjunção:
  - (i). $\sim(p \wedge q) \equiv \sim p \vee \sim q$
  - (ii). $\sim(p \wedge q) \equiv p \rightarrow \sim q$
  - (iii). $\sim(p \wedge q) \equiv q \rightarrow \sim p$

2. Equivalências da Negação de Proposições

- Aplicando essas equivalências para o caso em questão, ficamos com:
- (i). $\sim(c \wedge \sim f) \equiv \sim c \vee \sim(\sim f)$
- (ii). $\sim(c \wedge \sim f) \equiv c \rightarrow \sim(\sim f)$
- (iii). $\sim(c \wedge \sim f) \equiv \sim f \rightarrow \sim c$

- As expressões ficam:
- (i). $\sim(c \wedge \sim f) \equiv \sim c \vee f$
- (ii). $\sim(c \wedge \sim f) \equiv c \rightarrow f$
- (iii). $\sim(c \wedge \sim f) \equiv \sim f \rightarrow \sim c$

2. Equivalências da Negação de Proposições

- Observe que a equivalência (i). $\sim c \vee f$: "[Não corro] ou [fico cansado]" não corresponde a nenhuma alternativa.
- Já a equivalência (ii) fica $c \rightarrow f$: "Se [corro], então [fico cansado]." Correspondente a letra A.
- O gabarito, portanto, é a alternativa A.

3. Conjunção de Condicionais

- Existem algumas equivalências menos comuns, que também podem ser úteis em algumas situações. Elas relacionam disjunção inclusiva com condicional:
- $(p \rightarrow r) \wedge (q \rightarrow r) \equiv (p \vee q) \rightarrow r$
- $(p \rightarrow q) \wedge (p \rightarrow r) \equiv p \rightarrow (q \wedge r)$

P	Q	R	$P \rightarrow R$	$Q \rightarrow R$	$P \vee Q$	$(P \rightarrow R) \wedge (Q \rightarrow R)$	$(P \vee Q) \rightarrow R$
V	V	V	V	V	V	V	V
V	F	V	V	V	V	V	V
V	V	F	F	F	V	F	F
V	F	F	F	V	V	F	F
F	V	V	V	V	V	V	V
F	F	V	V	V	F	V	V
F	V	F	V	F	V	F	F
F	F	F	V	V	F	V	V

P	Q	R	$P \rightarrow Q$	$P \rightarrow R$	$Q \wedge R$	$(P \rightarrow Q) \wedge (P \rightarrow R)$	$P \rightarrow (Q \wedge R)$
V	V	V	V	V	V	V	V
V	F	V	F	V	F	F	F
V	V	F	V	F	F	F	F
V	F	F	F	F	F	F	F
F	V	V	V	V	V	V	V
F	F	V	V	V	F	V	V
F	V	F	V	V	F	V	V
F	F	F	V	V	F	V	V