

Atividade Unidade 1

Tabelas-Verdade em Lógica Proposicional

Nesta atividade prática, você aplicará os conceitos de lógica proposicional para analisar sentenças lógicas e construir tabelas-verdade. Esses exercícios simulam situações em que a lógica formal é utilizada para verificar a consistência e validade de diferentes proposições, algo essencial em áreas como programação, matemática e ciências da computação.

Você deve resolver as três questões a seguir, construindo as tabelas-verdade para cada proposição apresentada. Certifique-se de representar todos os possíveis valores de verdade (Verdadeiro e Falso) para as proposições envolvidas.

Situação 1:

Em um sistema de verificação de segurança, a combinação de dois eventos PPP e QQQ deve ser analisada por uma regra lógica que determina se o acesso é permitido. A regra é descrita pela seguinte proposição: $(P \vee Q) \wedge (\sim P \vee \sim Q)$

Construa a tabela-verdade para essa proposição, representando todas as combinações possíveis.

P	Q	$P \vee Q$	$\sim P$	$\sim Q$	$\sim P \vee \sim Q$	$(P \vee Q) \wedge (\sim P \vee \sim Q)$
V	V	V	F	F	F	F
V	F	V	F	V	V	V
F	V	V	V	F	V	V
F	F	F	V	V	V	F

Situação 2:

Um software de monitoramento de processos utiliza uma lógica de controle baseada na regra lógica usada pelo sistema descrita pela proposição: $\sim P \vee Q \rightarrow S$

Monte a tabela-verdade para essa proposição, levando em conta todas as combinações possíveis.

P	Q	S	$\sim P$	$\sim P \vee Q$	$\sim P \vee Q \rightarrow S$
V	V	V	F	V	V
V	V	F	F	V	F
V	F	V	F	F	V
V	F	F	F	F	V
F	V	V	V	V	V
F	V	F	V	V	F
F	F	V	V	V	V
F	F	F	V	V	F

Situação 3:

Em uma simulação de processos paralelos interagem de acordo com a seguinte proposição composta: $(P \wedge Q \rightarrow P \vee Q) \wedge (S \vee \sim P)$

Construa a tabela-verdade para essa proposição, considerando todas as possíveis combinações.

P	Q	S	$P \wedge Q$	$P \vee Q$	$P \wedge Q \rightarrow P \vee Q$	$\sim P$	$S \vee \sim P$	$(P \wedge Q \rightarrow P \vee Q) \wedge (S \vee \sim P)$
V	V	V	V	V	V	F	V	V
V	V	F	V	V	V	F	F	F
V	F	V	F	V	V	F	V	V
V	F	F	F	V	V	F	F	F
F	V	V	F	V	V	V	V	V
F	V	F	F	V	V	V	V	V
F	F	V	F	F	V	V	V	V
F	F	F	F	F	V	V	V	V