

# PETROBRAS

ENGENHARIA DE EQUIPAMENTOS - ELETRÔNICA

ENGENHEIRO(A) JÚNIOR - ÁREA: AUTOMAÇÃO

## MICROPROCESSADORES, CLPs E ARQUITETURAS COMPUTACIONAIS

QUESTÕES RESOLVIDAS PASSO A PASSO



PRODUZIDO POR EXATAS CONCURSOS

[www.exatas.com.br](http://www.exatas.com.br)

# ÍNDICE DE QUESTÕES

---

## ENGENHARIA DE EQUIPAMENTOS - ELETRÔNICA - CEBRASPE - PETROBRAS 2021

Q107 (pág. 1) Q108 (pág. 1) Q109 (pág. 2) Q110 (pág. 2)

## ENGENHEIRO(A) DE EQUIPAMENTOS JÚNIOR - ELETRÔNICA - PETROBRAS 2018.1

Q64 (pág. 3) Q66 (pág. 3) Q67 (pág. 4) Q68 (pág. 5)

## ENGENHEIRO(A) JÚNIOR - AREA: AUTOMAÇÃO - TRANSPETRO 2018.1

Q26 (pág. 6) Q55 (pág. 7) Q56 (pág. 8) Q58 (pág. 9) Q59 (pág. 10)  
Q60 (pág. 11) Q63 (pág. 11) Q67 (pág. 12)

## ENGENHEIRO(A) DE EQUIPAMENTOS JÚNIOR - ELETRÔNICA - PETROBRAS 2014.2

Q63 (pág. 17) Q65 (pág. 17)

## PROFISSIONAL JÚNIOR - ENGENHARIA ELETRÔNICA - BR DISTRIBUIDORA 2014

Q39 (pág. 13)

## ENGENHEIRO(A) DE EQUIPAMENTOS JÚNIOR - INSTRUMENTAÇÃO - INNOVA 2012

Q45 (pág. 14) Q58 (pág. 15)

## ENGENHEIRO(A) DE TERMELÉTRICA JÚNIOR - ELETRÔNICA - TERMOBAHIA 2012

Q53 (pág. 16)

## ENGENHEIRO(A) DE EQUIPAMENTOS JÚNIOR - ELETRÔNICA - PETROBRAS 2012.1

Q64 (pág. 18)

## ENGENHEIRO(A) DE EQUIPAMENTOS JÚNIOR - ELETRÔNICA - PETROBRAS 2011

Q56 (pág. 19) Q57 (pág. 20) Q58 (pág. 21) Q61 (pág. 22)

## ENGENHEIRO(A) DE EQUIPAMENTOS JÚNIOR - ELETRÔNICA - PETROBRAS 2010.2

Q55 (pág. 23) Q67 (pág. 24) Q68 (pág. 26) Q69 (pág. 27)

## ENGENHEIRO(A) DE EQUIPAMENTOS JÚNIOR - ELETRÔNICA - PETROBRAS 2010.1

Q4 (pág. 28) Q5 (pág. 30) Q25 (pág. 29) Q26 (pág. 31) Q57 (pág. 32)  
Q58 (pág. 36)

ENGENHEIRO(A) JÚNIOR - AREA: AUTOMAÇÃO - TRANSPETRO 2012

Q26 (pág. 37)

ENGENHEIRO(A) JÚNIOR - AREA: AUTOMAÇÃO - TRANSPETRO 2011

Q47 (pág. 38) Q48 (pág. 40) Q49 (pág. 41)

ENGENHEIRO(A) JÚNIOR - AREA: AUTOMAÇÃO - TRANSPETRO 2008

Q39 (pág. 39)

ENGENHEIRO(A) JÚNIOR - AREA: AUTOMAÇÃO - TRANSPETRO 2006

Q34 (pág. 42) Q35 (pág. 43) Q39 (pág. 44) Q40 (pág. 45)

ENGENHEIRO(A) DE EQUIPAMENTOS JÚNIOR - ELETRÔNICA - TERMOAÇU 2008.1

Q42 (pág. 46) Q43 (pág. 47) Q57 (pág. 49) Q58 (pág. 50) Q59 (pág. 51)  
Q60 (pág. 48)

ENGENHEIRO(A) DE EQUIPAMENTOS JÚNIOR - ELETRÔNICA - REFAP 2007

Q29 (pág. 52) Q30 (pág. 53)

ENGENHEIRO(A) DE TERMELÉTRICA JÚNIOR - ELETRÔNICA - TERMOCEARÁ 2009

Q47 (pág. 53)

ENGENHEIRO(A) - ELETRÔNICA - ELETROBRAS ELETRONUCLEAR 2010

Q45 (pág. 54) Q46 (pág. 56)

PROFISSIONAL JÚNIOR - ENGENHARIA ELETRÔNICA - BR DISTRIBUIDORA 2008

Q46 (pág. 55)

ENGENHEIRO(A) DE EQUIPAMENTOS PLENO - ELETRÔNICA - PETROBRAS 2006

Q33 (pág. 58) Q38 (pág. 59) Q58 (pág. 60)

TÉCNICO DE INSTRUMENTAÇÃO - TRANSPETRO 2006

Q34 (pág. 62) Q37 (pág. 61)

QUESTÕES RESOLVIDAS NESTA APOSTILA: 63

## QUESTÃO 25

ENGENHEIRO(A) DE EQUIPAMENTOS JÚNIOR - ELETRÔNICA - PETROBRAS 2011

A instrução da arquitetura x86 que realiza um acesso direto entre memória e I/O, não passando o dado, portanto, pela CPU, é

- (A) `lea`
- (B) `out`
- (C) `insb`
- (D) `movsx`
- (E) `movsw`

## RESOLUÇÃO

A alternativa correta deve apresentar uma instrução que transfere dados diretamente entre um endereço de memória e uma porta I/O do dispositivo, sem utilizar um registrador de uso geral. Para isso, o candidato deve conhecer detalhadamente cada uma das funções apresentadas e os tipos de operadores que estas aceitam. Apesar da arquitetura x86 ser registrador-memória, busca-se uma instrução com operandos de memória, lembrando que em x86 portas I/O são mapeadas em memória.

- (A) `lea`: Em inglês, *load effective address*. A função desta instrução é escrever o endereço de uma variável fonte em um registrador de destino. Assemelha-se ao operador unário `&` da linguagem C, como uma instrução para lidar com ponteiros.
- (B) `out`: Para a comunicação com dispositivos externos, o comando de saída é usado para o envio de informações a uma porta. `out` copia um byte ou *word* do acumulador para uma porta especificada. Quase atende o solicitado no enunciado, porém um dos operadores, neste caso, tem que ser um acumulador, onde já se encontre informação acessada da memória.
- (C) `insb`: `ins`/`insb`/`insw` são versões mais avançadas das instruções `in` e `out`. `insb` registra uma string presente em uma porta E/S especificada no argumento da função e armazena-o em um endereço de memória também fornecido na chamada da função. Portanto o dado é movido diretamente da porta para a memória (ou vice-versa), **sem passar pela CPU**.
- (D) `movsx`: Em inglês, *move sign-extends*. Junto com a operação `movzx`, é útil quando um valor presente em um registrador de 8 bits for movido para outro de 16 bits. No caso de `movsx`, o bit de sinal deste valor é considerado e os 8 bits superiores que faltam são preenchidos de acordo no registrador de 16 bits.
- (E) `movsw`: A operação `movs` move valores entre endereços de memória. Sua variante `movsw` move uma *word*. Apesar da semelhança com a função solicitada no enunciado, instruções do tipo `mov` não são a resposta correta do exercício, pois a arquitetura demanda operações específicas para acessar as portas E/S.

ALTERNATIVA (C)