

Verdadeiro ou falso

1. Uma desvantagem da representação de sinal-magnitude é que existem duas representações de 0.

R = V

2. Tanto a representação de sinal-magnitude quanto a representação de dois complementos usam o bit mais significativo como bit de sinal.

R = V

3. Não é necessário que a ALU sinalize quando ocorrer um overflow.

R = F

4. Overflow pode ocorrer somente se houve um carry.

R = V

5. Comparada com a adição e subtração, a multiplicação é uma operação complexa, seja realizada em hardware ou software.

R = V

6. Para cada 1 no multiplicador, é necessária uma operação de adição e shift; mas para cada 0, apenas um shift é necessário.

R = V

7. A adição e a subtração podem ser realizadas em números em notação de complemento de dois, tratando-os como inteiros não assinados.

R = V

8. O algoritmo de Booth executa mais adições e subtrações do que um algoritmo direto.

R = V

9. Com uma notação de ponto fixo, é possível representar uma gama de inteiros positivos e negativos centrados em ou perto de 0.

R = V

10. Uma vantagem da representação polarizada é que os números não negativos de ponto flutuante podem ser tratados como números inteiros para fins comparativos.

R = F

11. Para a representação da base 2, um número normal é aquele em que o bit mais significativo do significand é zero.

R = V

12. As representações reais de ponto flutuante incluem um padrão de bits especial para designar zero.

R = V

13. Os números representados na notação de ponto flutuante não são espaçados uniformemente ao longo da linha numérica, assim como os números de pontos fixos.

R = F

14. O overflow é um problema menos grave porque o resultado em geral pode ser satisfatoriamente aproximado por 0.

R = F

15. Uma das desvantagens da matemática de ponto flutuante é que muitos cálculos produzem resultados que não são exatos e devem ser arredondados para o valor mais próximo que a notação pode representar.

R = V

MÚLTIPLA ESCOLHA

1. O esquema mais comum na implementação da parte inteira da ALU é:

R = C. representação de complemento de dois

2. A representação **COMPLEMENTO DE DOIS** é quase universalmente usada como a representação do processador para números inteiros.

3. O ato de mover o bit de sinal para a nova posição mais à esquerda e preencher com as cópias do bit de sinal é chamado **EXTENSÃO DE AMPLITUDE**.

4. Na representação **DE SINAL MAGNITUDE**, a regra para formar a negação de um número inteiro é inverter o bit de sinal.

5. **OVERFLOW** é quando o resultado pode ser maior do que pode ser mantido no tamanho da palavra que está sendo usada.

6. **MULTIPLICAÇÃO** envolve a geração de produtos parciais, um para cada dígito do multiplicador, que são somados para produzir o produto final.

7. Embora considerado obsoleto, o termo **MANTISSA** às vezes é usado em vez de significando.

8. Números negativos inferiores a $-(2 - 2^{-23}) \times 2^{128}$ são chamados de **OVERFLOW NEGATIVO**.

9. Números negativos maiores que 2^{-127} são chamados de **UNDERFLOW NEGATIVO**.

10. Números positivos inferiores a 2^{-127} são chamados de **UNDERFLOW POSITIVO**.

11. Números positivos maiores que $(2 - 2^{-23}) \times 2^{128}$ são chamados de **OVERFLOW POSITIVO**.

12. Os formatos **DE INTERCÂMBIO** estendem um formato básico suportado, fornecendo bits adicionais no expoente e no significando.

13. **NÚMEROS SUBNORMAIS** estão incluídos no IEEE 754 para lidar com casos de underflow de expoente.

14. **OVERFLOW DE EXPOENTE** é quando um expoente positivo excede o valor máximo possível de expoente.

15. **UNDERFLOW DE EXPOENTE** significa que o número é muito pequeno para ser representado e pode ser relatado como 0.