



Prática 03 - Operadores Bit a Bit

1. Escreva um programa que inverta a ordem dos bits em um valor inteiro. O programa deve inserir o valor do usuário e o resultado deve imprimir os bits em ordem inversa. Imprima o valor em bits antes e depois de os bits serem invertidos para confirmar que os bits foram corretamente invertidos.
2. Quando lemos um byte da porta paralela (usada antigamente pela impressora), não temos acesso aos três primeiros bits do byte e além disso, o bit mais significativo vem invertido. Faça um programa que leia um byte do teclado e a seguir inverta o bit mais significativo e zere os três bits menos significativos.
3. Faça um programa que leia um byte do teclado e a seguir zere os bits 3 e 4, e inverta os bits 0 e 7. O resultado deverá ser mostrado em hexadecimal na tela.
4. Como exemplo de um possível algoritmo de criptografia, bem simples, para cada byte lido, trocar os dois primeiros bits pelos dois últimos de cada byte. Exemplo o byte lido 01000000 ficará 00000001.
5. Seja a representação de cor de um pixel de uma imagem feita através de um valor inteiro de 32 bits, cujos bits obedecem o seguinte padrão hexadecimal 0xAARRGGBB, onde o byte 0xAA é o valor da opacidade [0-255], e 0xRR, 0xGG e 0xBB [0-255] três bytes das componentes de cor vermelha, verde e azul respectivamente.
 - a. faça um programa que receba como entrada o valor hexadecimal da cor de um pixel e forneça na saída os valores da opacidade e das três componentes de cor em hexadecimal.
 - b. faça um programa que receba como entrada os quatro valores separados da opacidade e das três componentes de cor na base decimal, e forneça na saída o valor de 32 bits da cor em hexadecimal.
 - c. faça um programa que receba como entrada o valor hexadecimal da cor de um pixel e:
 - c1. receber também um valor da opacidade que deve substituir o valor da opacidade da cor lida anteriormente;
 - c2. receber também um valor da componente vermelha que deve substituir o valor da componente vermelha da cor lida anteriormente;
 - c3. receber também um valor da componente verde que deve substituir o valor da componente verde da cor lida anteriormente;
 - c4. receber também um valor da componente azul que deve substituir o valor da componente azul da cor lida anteriormente;OBS.: Em cada um dos quatro casos acima, exibir o valor resultante da substituição em hexadecimal.
 - d. sabe-se que um dos possíveis 256 tons de cinza (porque 256???) é gerado quando cada um dos componentes R, G e B são exatamente o mesmo valor (teste esta característica em algum editor gráfico como o Paint da Microsoft). faça um programa que receba como entrada um valor entre 0-255 na base decimal e forneça na saída o valor de 32 bits em hexadecimal do tom de cinza gerado a partir do valor lido anteriormente. exemplo1: se o valor lido foi 128,

então, considerando a opacidade igual a 0x00, o resultado deve ser 0x00808080. exemplo2: se o valor lido foi 32, então, considerando a opacidade igual a 0x00, o resultado deve ser 0x00202020.

6. Escreva um aplicativo que desloca para a direita uma variável inteira por quatro bits para a direita com o deslocamento para a direita com sinal e, então, desloca essa mesma variável inteira por quatro bits para a direita com o deslocamento para a direita sem sinal. O programa deve imprimir o inteiro em bits antes e depois de cada operação de deslocamento. Execute seu programa uma vez com um inteiro positivo e outra com um inteiro negativo.

7. Mostre como o deslocamento de um inteiro para a esquerda por um pode ser utilizado para realizar a multiplicação por dois e como o deslocamento de um inteiro para a direita por um pode ser utilizado para realizar a divisão por dois. Tenha cuidado em considerar as questões relacionadas ao sinal de um inteiro.

Disponível a partir de: sexta, 24 maio 2013, 13:55

Data de entrega: sexta, 31 maio 2013, 13:55

Você acessou como [Jefferson Uchôa Ponte](#) (Sair)

[TPE1N2013](#)