

DIM0108.0 - 2011.1 - Turma 01 - Prova 1

Prof: David Déharbe (DIMAp/UFRN)

24 de Outubro de 2011

Exercício 1 *Qual a saída produzida para o programa seguinte?*

```
#include <stdio.h>
int main ()
{
    int n = 2;
    int m = 3;
    int p = 4;
    if (n + m * p == 20) printf("S"); else printf("N");
    if (n + m * p == 14) printf("S"); else printf("N");
    if (n - m - p == -5) printf("S"); else printf("N");
    if (n - m - p == 3) printf("S"); else printf("N");
    if (- n - m == -5) printf("S"); else printf("N");
    if (- n - m == 1) printf("S"); else printf("N");
    printf("\n");

    return 0;
}
```

Considerando que:

- a precedência do operador * é maior que a do operador +;
- o operador binário - é associativo a esquerda;
- a precedência do operador unário - é maior que a do operador binário -;

a saída produzida pelo programa é:

NSSNSN

Exercício 2 *Escrever um programa que lê três números inteiros, e imprime S se um desses três números é igual à soma dos outros dois e imprime N caso contrário.*

```

#include <stdio.h>

int main ()
{
    int a;
    int b;
    int c;

    if (a == b + c || b == c + a || c == b + a)
        printf("S\n");
    else
        printf("N\n");

    return 0;
}

```

Exercício 3 *Qual é a saída produzida pela execução do seguinte programa?*

```

#include <stdio.h>

void mist(int * x, int * y, int z)
{
    * x = * x + *y;
    * y = z;
}

int main ()
{
    int a = 1;
    int b = 2;
    int c = 3;
    int* p = &c;

    mist(&a, &b, c);
    printf("%i %i %i\n", a, b, c);
    mist(&b, &a, *p);
    printf("%i %i %i\n", a, b, c);
    mist(p, p, 0);
    printf("%i %i %i\n", a, b, c);
    mist(p, &a, b);
    printf("%i %i %i\n", a, b, c);

    return 0;
}

```

A sub-rotina `mist` tem como parâmetros `x` e `y`, ambos guardam o endereço de uma variável de tipo `int`, e `z`, que guarda um valor do tipo `int`.

O resultado da execução de `mist` é de atribuir à variável apontada por `x` a soma das variáveis apontadas por `x` e por `y`, e então à variável apontada por `y` o valor guardado em `z`.

Note que se `x` e `y` são iguais, então a variável apontada por eles é atribuída o valor guardado em `z`.

Obs. Há um erro na quarta linha: não deveria aparecer `*p` e sim `p`.

Caso tivesse sido `p`, a execução resultaria na seguinte saída:

3 3 3
3 6 3
3 6 0
6 6 3

Exercício 4 Existe na linguagem C um operador de pós-decremento que permite decrementar (ou seja, diminuir em um) o valor de uma variável inteira. Seja x do tipo `int`, a avaliação da expressão $x--$ resulte na diminuição em um (1) do valor de x , e retorna o valor inicial de x .

Por exemplo, se x for igual a 0, a execução do comando $y = x--$; resulta na atribuição a y do valor 0 e a x do valor -1.

Defina uma sub-rotina com interface `int pos_dec(int * p)`, tal que `pos_dec(&x)` seja equivalente a $x--$.

```
int pos_dec(int * p)
{
    int inicial = *p;
    *p = *p - 1;
    return inicial;
}
```

Exercício 5 Assinale a resposta correta.

Seja a uma variável do tipo `unsigned int` com um valor qualquer. As expressões $0 \wedge a$, $0 \& a$, e $0 | a$ são respectivamente iguais a:

1. $a, a, e a$;
2. $0, a, e a$;
3. $a, 0, e a$;
4. $a, a, e 0$;
5. $0, 0, e 0$.

Observe que:

- $0 \wedge 0 = 0$ e $0 \wedge 1 = 1$, logo $0 \wedge b = b$, para qualquer bit b ;
- $0 \& b = 0$, para qualquer bit b ; e
- $0 | b = b$, para qualquer bit b .

Logo, a terceira alternativa é a correta.

Exercício 6 Para estimular a inovação tecnológica dos produtores de automóveis, o Governo Federal resolveu instituir um imposto extra (IE) que incide sobre os veículos que tem menos de dois terços dos seus componentes manufaturados no Mercosul. O montante deste imposto é de 33,5% do valor de produção de veículo. Este imposto acrescenta-se ao imposto sobre produtos industriais (IPI) que é de 27,5% do valor de produção do produto.

Escreva uma sub-rotina que, dado o valor de produção de um veículo, e a proporção de componentes fabricados no Mercosul, retorna o valor do veículo incluído o IE e o IPI.

```
/**
 \brief Calcula o valor de um veiculo com impostos
 \param valor_inicial o valor do veiculo sem impostos
 \param percentual_mercosul percentual de componentes do veiculo
        manufaturados no mercosul
 \return valor final do veiculo.
 */
float valor_final(float valor_inicial, float percentual_mercosul)
{
    float valor_ipi;
    float valor_ie;
    float ipi = 0.275;
    float ie = 0.335;
    valor_ipi = valor_inicial * ipi;
    if (percentual_mercosul < 2./3.)
        valor_ie = valor_inicial * ie;
    else
        valor_ie = 0.;
    return valor_inicial + valor_ie + valor_ipi;
}
```