

## Solução exame de PI de 5 de Janeiro de 2006

### Pergunta 1

*Type casting* é a conversão do tipo da informação. Por exemplo, imagine um float  $a$  e um int  $b$ . Não é permitido a atribuição  $a = b$ , pois isso é uma mistura de tipos. Antes de atribuir o valor, temos de converter-o,  $a = (\text{float}) b$ . Nos dois lados do = deve estar o mesmo tipo da informação.

### Pergunta 2

$$56 = 32+16+8 = 0x128 + 0x64 + 1x32 + 1x16 + 1x8 + 0x4 + 0x2 + 0x1 \\ = 00111000$$

$$156 = 128 + 16+8+4 = 1x128 + 0x64 + 0x32 + 1x16 + 1x8 + 1x4 + 0x2 + 0x1 \\ = 10011100$$

$$\begin{array}{r} 56 = 00111000 \\ 156 = 10011100 \\ \hline 56 \& 156 = 00011000 = 16+8 = 24 \end{array}$$

a	b	a & b
1	1	1
1	0	0
0	1	0
0	0	0

24

### Pergunta 3

```
int numesp(char s[])
{
    int cnt = 0, i = 0;
    int len;

    while (s[i] != '\0')
    {
        if (s[i] == ' ')
            cnt++;
        i++;
    }
    return(cnt);
}
```

### Pergunta 4

```
float area(float a, float b, int n)
{
    int i;
    float dx, x;
    float suma=0.0;

    dx = (b-a)/ (float) n; /* veja pergunta 1 */
    for (i=0; i<n; i++)
    {
        x = a + (float) i * dx;
        suma += dx*sin(x);
    }
    return(suma);
}
```

### Pergunta 5

Esta pergunta tem a ver com o trabalho prático deste ano, o problema do sudoku. Há muitas soluções. Por isso também é importante explicar como é que funciona o seu algoritmo.

**a)**

```
int coluna(int a[9][9], int n)
{
    int ipresente;
    int i, j;

    for (i=1; i<=9; i++)
        { /* i: o numero a procurar na coluna */
            ipresente = 0;
            for (j=0; j<9; j++) /* j: indice da linha */
                if (a[j][n]==i)
                    ipresente=1; /* encontrou i: asinala */
            if (ipresente==0)
                return(0)
            /* encontrou um i nao presente -> sair */
        }
    /* quando chega aqui, todos os i estiveram presente */
    return(1);
}

int linha(int a[9][9], int m)
{
    int ipresente;
    int i, j;

    for (i=1; i<=9; i++)
        {
            ipresente = 0;
            for (j=0; j<9; j++)
                if (a[m][j]==i)
                    ipresente=1;
            if (ipresente==0)
                return(0)
        }
    return(1);
}
```

**b)**

```
main()
{
    int i, b[9][9];
    int sudoku=1;

    for (i=0; i<9; i++)
        {
            if (coluna(b, i)==0)
                sudoku=0;
            if (linha(b, i)==0)
                sudoku=0;
        }
}
```

```

    }
    if (sudoku==1)
        printf("A matriz esta certa");
    else
        printf("A matriz esta mal");
}

```

### **Pergunta 6**

**a)**

```

typedef struct {
    long int num;
    char nome[41];
    int curso;
    int ano;
    struct {
        char nome[31];
        char sigla[6];
        int nota;
        float creditos;
    } disciplina[10];
} aluno;

```

**b)**

```
aluno alunos[1000];
```

**c)**

$1000 \times (4 + 41 \times 1 + 2 + 2 + 10 \times (31 \times 1 + 6 \times 1 + 2 + 4)) = 479000$  bytes

**d)**

```

printf("Lista dos alunos a ser tirado do curso:\n");
for (i=0; i<1000; i++)
{
    sumcreditos = 0.0;
    for (j=0; j<10; j++)
        if (alunos[i].disciplina[j].nota >= 10)
            sumcreditos += alunos[i].disciplina[j].creditos;
    if (sumcreditos<30.0)
        printf("%s\n", alunos[i].nome);
}

```