



# Transcrição de aula

Disciplina	Programação Imperativa - 1º ano - LEI		
Secretário	Número: 64333	Nome:	Decho Miguel de Melo Ferreira
Data:	2012-03-19	Nº Página	
Turno:	TP5	Nº Alunos	

## SUMÁRIO

```
a) int x [15] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15};
    int *y, int *z;
    y = x; z = x + 3;
int *i = 0;
    while (i < 5)
    { printf ("%d %d %d \n", x[i], *y, *z);
      y++;
      z += 2;
      i++;
    }
```

i	x [i]	*y	*z
0	1	1	4
1	2	2	6
2	3	3	8
3	4	4	10
4	5	5	12



```
b) void r ( int a [], int m)
{
    int i, tmp;
    tmp = a [0];
    for (i = 1; (i < m) i++)
        a [i-1] = a [i];
    a [m-1] = tmp;
}

int main ()
{
    int v [10] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };
    int i;
    for (i = 0; (i < 5); i++) r (v, 10);
    for (i = 0; (i < 5); i++) printf ("%d", v [i]);
    return 0;
}
```

Para cada uma das alíneas que se seguem, defina um programa que lê (usando a função scanf) uma sequência de (no máximo 100) números inteiros terminada com o número 0 e imprime no ecrã:

- quantos números são superiores à média;
- a sequência sem os elementos repetidos (se os houver).

```
(a) float media (int lista [], int melems)
{
    int i = 0;
    int b = 0;
    while (i < melems)
        { b = b + a [i];
          i++;
        }
    return b / melems;
}
```



```
int lerLista (int Lista [], int mElem)
{
    int i = 0;
    int x = 1;
    while (i < mElem && x != 0)
    {
        printf ("Insira número ");
        scanf ("%d", &x);
        lista [i] = x;
        i++;
    }
    return i - 1;
}

int supmedia (int Lista [], int mElem, float media)
{
    int i = 0, res = 0;
    while (i < mElem)
    {
        if (lista [i] > media)
            res++;
        i++;
    }
    return res;
}

# define MAXELEM 100
int main ()
{
    float m;
    int SeqInt [MAXELEM], cont;
    cont = lerLista (SeqInt, MAXELEM);
    m = media (SeqInt, cont);
    printf ("! %d não superiores a %.f \n", supmedia (SeqInt, cont, m), m);
    return 0;
}
```



Transcrição (folha de continuação)

NºPágina: 4

```
int atoi (char #digitos)
{
    int = 0, res = 0, a = 0;
    while (digitos[i] != '\0')
    {
        a = digitos[i] - '0';
        res = 10 * res + a;
        i++;
    }
    return res;
}
```

```
int bin2dec (char #b)
{
    int = 0, res = 0, m = strlen(b) - 1;
    if (digitos[1] == '\0') res = 0;
    else {
        if (digitos[1] == '1')
            res = 2^m + bin2dec(digitos + 1);
        else
            res = bin2dec(digitos + 1);
    }
    return res;
}
```