

1. Olhe lá! O formato dos slides é melhor para ilustrar estas questões.

2.

```
#include <stdio.h>

int main()
{
    int x;
    int* ptr;

    ptr = &x;
    x = 42;

    printf ("Valor de x: %d\n", x);
    printf ("Endereco de x: %p\n", &x);
    printf ("Endereco apontado por ptr: %p\n", ptr);
    printf ("Valor apontado por ptr: %d\n", *ptr);
    printf ("Endereco d ptr: %p\n", &ptr);

    return 0;
}
```

3.

/* Esta é bem simples, mas tem uma pegadinha: se você usar algo como *a++, sem parêntesis, não vai incrementar o conteúdo apontado por a, e sim O PRÓPRIO A. E aí a apontará para o endereço seguinte, mas sem modificar o conteúdo apontado!!! Vou colocar também um main que chama a função para ilustrar. */

```
void incDec (int *a, int *b) {
    (*a)++;
    (*b)--;
}

int main ()
{
    int foo = 10, bar = 20;

    printf ("Era %d e %d\n", foo, bar);
    incDec (&foo, &bar);
    printf ("Ficou %d e %d\n", foo, bar);

    return 0;
}
```

4.

```
float min_max_media (int n, float* menor, float* maior)
{
    int i;
    float num, soma;

    // Lê o primeiro número fora do loop.
    scanf ("%f", &num);

    *maior = num;
    *menor = num;
    soma = num;

    // Começa em 1 porque o 1o valor já foi lido.
    for (i=1; i < n; i++)
    {
        scanf("%f", &num);
        if (num > *maior)
            *maior = num;
        if(num < *menor)
            *menor = num;
        soma += num;
    }

    return (soma / n);
}

int main ()
{
    float min, max, media;

    media = min_max_media (10, &min, &max);

    printf ("Menor: %.4f\n", min);
    printf ("Maior: %.4f\n", max);
    printf ("Média: %.4f\n", media);

    return 0;
}
```