

(\*\*) 1. Desenvolva uma função que recebe um número de segundos e converte para o tempo correspondente no formato de horas, minutos e segundos. Por exemplo, 10000 segundos corresponde a 2 horas, 46 minutos e 40 segundos. O protótipo da função deve ser:

```
void segundosParaHMS (int total_segundos, int *h, int *m, int *s);
```

Inclua na sua resposta uma função `main` simples, que invoque a função acima e mostre os valores retornados, no formato hh:mm:ss.

---

(\*\*) 2. Suponha a seguinte função, a qual retorna o primeiro e o último dígito de um número:

```
void removeExtremos (int *n, int *pri, int *ult)
{
    int tn, pot = 1;

    tn = *n;
    while(tn >= 10)
    {
        tn = tn/10;
        pot *= 10;
    }

    *pri = *n / pot;
    *ult = *n % 10;
    *n = *n % pot;
    *n = *n / 10;
}
```

Escreva um programa que utilize esta função para determinar se um dado inteiro positivo  $n$  é um número palíndromo (i.e. se pode ser lido igualmente da esquerda para a direita e da direita para a esquerda). Suponha que  $n$  não contém o dígito 0. Atenção: você não deve modificar a função dada no enunciado! Apenas invoque-a de forma adequada. (E também ignore o fato de que usar uma função assim não é a forma mais eficiente de testar se um número é palíndromo, se você se lembra!)

(☹☹) 3. Dizemos que uma sequência de  $n$  inteiros positivos é  $k$ -alternante se for composta alternadamente por segmentos de números pares de tamanho  $k$  e segmentos de números ímpares de tamanho  $k$ . Exemplos:

A sequência 1 3 6 8 9 11 2 4 1 7 6 8 é 2-alternante.

A sequência 2 1 4 7 8 9 12 é 1-alternante.

A sequência 4 2 3 1 6 4 2 9 3 não é alternante.

A sequência 1 3 5 é 3-alternante.

Escreva uma função que recebe como parâmetro um valor  $n$ , e lê (através da função `scanf`) até  $n$  números inteiros, verificando se a sequência dada é  $k$ -alternante. Você pode considerar que o  $n$  dado é maior que 1. A função deve retornar 1 caso a função seja  $k$ -alternante, e 0 do contrário; e deve retornar imediatamente, sem ler os  $n$  valores, caso detecte que a sequência não é  $k$ -alternante. Adicionalmente, caso ela seja  $k$ -alternante, o valor encontrado para  $k$  deve ser inserido em uma variável passada por referência à função.

---

(\*\*\*\*) 4. OPCIONAL! Em algumas tarefas de programação de software básico ou embarcado (software de “baixo nível”), a troca de dados com periféricos é feita lendo e escrevendo valores diretamente em endereços pré-determinados de memória – essa estrutura é conhecida como DMA (*Direct Memory Access*). Suponha que temos ligado ao processador um painel que possui 128 colunas de LEDs, cada coluna contendo 8 LEDs. Podemos criar padrões escrevendo dados nos 128 bytes a partir do endereço de memória 0x1000. Cada byte se refere a uma coluna, cada bit em um byte define o estado de um LED da coluna, com o bit mais significativo se referindo ao primeiro LED de cima para baixo e cada bit subsequente se referindo ao LED abaixo. Por exemplo, podemos acender o primeiro LED do painel usando o código abaixo:

```
#define LED_BASE_ADDR 0x1000
(...)
unsigned char* panel_ptr = LED_BASE_ADDR;
*panel_ptr = 0x80; /* O primeiro bit se refere ao LED do topo. */
```

Podemos também intercalar LEDs acesos e desligados na 2ª coluna da seguinte forma:

```
unsigned char* panel_ptr = LED_BASE_ADDR + 1; /* 2ª coluna */
*panel_ptr = 0xAA; /* Bits intercalados. */
```

Escreva um programa que acenda os LEDs como mostra o padrão abaixo (um `o` indica um LED aceso):

```
|o      o      o      o      o      o      o      o      | (...)
| o      o      o      o      o      o      o      o      | (...)
|  o      o      o      o      o      o      o      o      | (...)
|   o      o      o      o      o      o      o      o      | (...)
|    o      o      o      o      o      o      o      o      | (...)
|     o      o      o      o      o      o      o      o      | (...)
|      o      o      o      o      o      o      o      o      | (...)
|       o      o      o      o      o      o      o      o      | (...)
```

Por se tratar de um exemplo hipotético, você não poderá compilar e testar o programa – apenas escreva como seria o código.