

AVISO: Note que os exercícios abaixo pedem somente para criar uma função. Nenhuma delas envolve a entrada de dados com a função `scanf` nem a saída de dados usando a função `printf`. Nenhuma delas exige a criação de uma função `main`. Isso não quer dizer que você precisa criar e testar a função sem ter uma função `main`, ou sem visualizar resultados. Você **deve** criar um programa que chama a função passando os parâmetros adequados e apresenta os resultados, mas este programa não faz parte da solução que você vai apresentar. Conseguir separar as coisas faz parte de aprender a decompor problemas!

(* ou $\frac{1}{2}$, depende da sua solução) 1. Escreva uma função que arredonda um valor dado. O número deve ser arredondado para o inteiro mais próximo. Se o número for equidistante de dois inteiros, deve ser arredondado para o valor de maior magnitude. Ou seja, 1.5 é arredondado para 2, e -1.5 é arredondado para -2. O protótipo da função deve ser:

```
int arredonda (double x);
```

(*) 2. Escreva uma função que retorna apenas a parte não-inteira de um número dado. Por exemplo, se a função receber o valor 1.43, deve retornar 0.43. O protótipo da função deve ser:

```
double casasDecimais (double x);
```

(***) 3. A sequência de Fibonacci é uma sequência de números naturais na qual os primeiros termos são 0 e 1, e cada termo subsequente é a soma dos dois termos anteriores. Desta forma, o terceiro termo é $0+1=1$, o quarto termo é $1+1=2$, o quinto termo é $1+2=3$, o sexto termo é $2+3=5$ e assim por diante. Os 10 primeiros números da sequência são:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34

Escreva uma função que recebe um número inteiro n passado por parâmetro e devolve o primeiro número da série de Fibonacci que é maior ou igual a n . Por exemplo, para $n=7$, a função retorna 8, pois 8 é o primeiro termo que pertence à sequência de Fibonacci que é maior ou igual a 7. Da mesma forma, para $n=15$, a função retorna 21.

```
int proxFibonacci (int n);
```

(***) 4. Escreva uma função que receba uma base b e um expoente e , e calcule o valor de b^e . Considere que todos os valores envolvidos são inteiros positivos. O protótipo da função deve ser:

```
unsigned long long potencia (unsigned int base, unsigned int expoente);
```

(***) 5. Escreva uma função que recebe um número com uma quantidade arbitrária de dígitos, e retorna o número com os dígitos invertidos. Por exemplo, se a função receber o valor 1234, deve retornar 4321. Zeros à esquerda devem ser ignorados – portanto, se a função receber o valor 1400, deve retornar 41. O protótipo da função deve ser:

```
unsigned int inverteNum (unsigned int n);
```

(*) 6. Escreva uma função que recebe um caractere e retorna 1 se o caractere for uma vogal maiúscula, 2 se for uma vogal minúscula, 3 se for uma consoante maiúscula, 4 se for uma consoante minúscula, 5 se for um dígito, e 0 do contrário. O protótipo da função deve ser:

```
int testaTipoChar (char c);
```

Dica: observe atentamente como é a tabela ASCII. Você não precisa usar NENHUM valor dela, mas entender a forma como os caracteres estão dispostos pode ajudar.