

Prova-02 - Loops, Functions and Pointers

Prof. Msc. Elias Batista Ferreira
Prof. Dr. Gustavo Teodoro Laureano
Profa. Dra. Luciana Berretta
Prof. Dr. Thierson Rosa Couto

Sumário

1	Números invertíveis em \mathbb{Z}_n (+++)	2
2	Potência prima (+++)	3

1 Números invertíveis em \mathbb{Z}_n (+++)



(+++)

A teoria dos números é um assunto fascinante. Por exemplo, podemos definir o conjunto \mathbb{Z} como o conjunto infinito dos números inteiros de modo que, qualquer operação, por exemplo a multiplicação, entre dois números desse conjunto produza um outro número que também pertence a esse conjunto. Dentro do conjunto \mathbb{Z} podemos definir outros conjuntos, tal o qual o conjunto \mathbb{Z}_n . Esse conjunto é definido em função de n , é finito e representa todos os números de 0 a $n - 1$. Por exemplo, o conjunto \mathbb{Z}_9 é formado pelos números $\{0, 1, 2, 3, 4, 5, 6, 7, 8\}$. Se considerarmos a teoria de anéis, podemos observar que qualquer número de \mathbb{Z} está dentro do conjunto \mathbb{Z}_n de forma cíclica. Por exemplo, o número 15 equivale ao número 6 no conjunto \mathbb{Z}_9 , porque $15 \bmod 9 = 6$, onde **mod** representa o operador de módulo, ou seja, o resto da divisão. Dentro do conjunto \mathbb{Z}_n há outro sub-conjunto, denominado de conjunto dos números invertíveis. Dois números a e b são ditos invertíveis dentro de \mathbb{Z}_n se $a \cdot b = 1$, ou $(a \cdot b) \bmod n = 1$. Isso significa dizer que o produto de a e b gera um número equivalente ao número 1 dentro de \mathbb{Z}_n .

Faça um programa que, dado o valor de n , apresente todos os pares de números invertíveis dentro do conjunto \mathbb{Z}_n .

Entrada

Um número inteiro que corresponde ao n do conjunto \mathbb{Z}_n .

Saída

O programa deverá apresentar todos os pares de números invertíveis em uma linha seguindo o formato " (x,y) ". Como x e y formam os pares (x,y) e (y,x) , para evitar duplicidades, seu programa deve apresentar somente os pares (x,y) tal que $x \leq y$.

Exemplo

Entrada	Saída
7	(1, 1) (2, 4) (3, 5) (6, 6)

Entrada	Saída
9	(1, 1) (2, 5) (4, 7) (8, 8)

2 Potência prima (+++)



(+++)

Uma potência prima n é um número maior que 1 que é escrito por uma potência de um único número primo. Ou seja, $n = k^p$, sendo k um número primo e maior do que 1. Exemplos de potências primas: $7 = 7^1$, $4 = 2^2$, $25 = 5^2$, $27 = 3^3$ e $81 = 3^4$. Escreva um programa que imprima os N primeiros termos da sequência de potências primas e suas representações em potências primas correspondentes.

Você deve implementar a função:

```
1  /**
2   * Função que verifica e decompõe um número n como uma potência prima. Esta
3   * função recebe o valor n e retorna o resultado da decomposição de n como uma
4   * potência prima via ponteiros. Quando a decomposição é possível, a função
5   * retorna 1.
6   * Quando a decomposição não é possível, a função retorna 0.
7   *
8   * @param n valor inteiro a ser verificado
9   * @param k ponteiro para a base da potência prima.
10  * @param p valor da potencia.
11  * @return retorna 1 caso n seja uma potência prima e 0 caso contrário.
12  * no último caso, quando n não é uma potência prima, os conteúdos de k
13  * e p devem ser desconsiderados.
14  */
15  int potencia_prima( int n, int * k, int * p );
```

Entrada

O programa deve ler um número inteiro N referente a quantidade de termos da sequência de potências primas.

Saída

O programa deve apresentar N linhas, cada uma correspondendo a uma potência prima e sua representação correspondente. A saída deve seguir o formato $n : k^p$.

Dica

Para facilitar, você pode construir outras funções auxiliares. Quando for imprimir o caracter $^$, certifique-se que ele foi digitado pelo teclado, ou seja, não coloque o símbolo via Ctrl+C | Ctrl+V.

Exemplo

Entrada	Saída
7	2 : 2^1 3 : 3^1 4 : 2^2 5 : 5^1 7 : 7^1 8 : 2^3 9 : 3^2