Números Inteiros e Criptografia, 2020.2

Lista de Exercícios 6[†]

Submeta as soluções das questões marcadas com *
até 27 de maio às 18:00 salvando um arquivo na sua pasta no Google Drive[‡]

***Questão 1.** Dado um número natural $k > 0$, vamos chamar de *primos k-gêmeos* os números primos que estão à distância k um do outro. Por exemplo, para $k = 2$ temos 3 & 5, 5 & 7, 11 & 13, etc., para $k = 4$ temos 3 & 7, 7 & 11, 13 & 17, etc.

Faça uma função em `python` que receba dois naturais `k` e `limite` e retorne a lista de todos os pares de primos k-gêmeos que existem até (incluindo, se for o caso) o `limite`.

Para receber a pontuação máxima, sua função deve funcionar em tempo razoável (no máximo alguns segundos no Google Colab) pelo menos para `limite` com até 5 algarismos.

Questão 2 (Reescrevendo expressões)**.** Em matemática, o uso de reticências (i.e., "$\cdots$" ou "$\ldots$") em expressões é bastante comum; por exemplo, a função $f : \mathbb{N} \to \mathbb{N}$ dada por

$$f(n) = \text{ a soma dos } n \text{ primeiros números naturais}$$

é comumente escrita da forma

$$f(n) = 1 + 2 + 3 + \cdots + (n - 1). \qquad (\star)$$

Entretanto, o uso de reticências pode causar problemas de incerteza e ambiguidade, pois assume que o leitor será capaz de *deduzir* o conteúdo ocultado pelas reticências, o que pode não ser imediato. De fato, é bem questionável deduzir o valor "correto" de $f(0)$ a partir da expressão ($\star$). (O valor que funciona melhor, e que se usa por convenção, é $f(0) = 0$.)

Em geral, o uso de reticências esconde uma definição recursiva; *oficialmente* a função f acima é definida por

$$\begin{cases} f(0) = 0 \\ f(n) = f(n-1) + (n-1), \quad \text{para } n > 0. \end{cases}$$

Em cada item abaixo, reescreva a expressão que define $g : \mathbb{N} \to \mathbb{N}$ de forma recursiva, sem o uso de reticências (nem de *somatórios*, *produtórios* ou afins).

[†]Publicada em 17/5

[‡]Link recebido por email em 1/4/2021. A pasta tem um nome similar a `<seu nome> - Cripto 2020.2 - Submissões e Feedback`; em caso de qualquer dúvida entre em contato com o professor.

a. $g(n) = 1^2 + 2^2 + \cdots + (n-1)^2$

b. $g(n) = 1 + 3 + 5 + \cdots + (2n-1)$

c. $g(n) = 1^3 + 2^3 + \cdots + (n-1)^3$

*** d.** $g(n) = \dfrac{1}{1 \cdot 2} + \dfrac{1}{2 \cdot 3} + \cdots + \dfrac{1}{(n+1)(n+2)}$

e. $g(n) = 2 \cdot 4 \cdot 6 \cdots (2n)$

*** f.** $g(n) = 2 \cdot 3 \cdot 5 \cdot 7 \cdot 11 \cdot 13 \cdot 17 \cdots p_n$, onde p_n é o n-ésimo primo (veja a Questão 8 para mais detalhes). (Você pode usar a expressão p_n na definição recursiva de g.)

*** g.** $g(n) = 2 \cdot 3 \cdot 5 \cdot 7 \cdot 11 \cdot 13 \cdot 17 \cdots p$, onde p é o *maior* primo tal que $p \leq n$. (Logo $g(3) = 6 = g(4)$, por exemplo. Qual deve ser a definição do caso base $g(0)$ para que a definição recursiva *funcione bem*?)

Questão 3. Prove que

a. $1^2 + 2^2 + \cdots + n^2 = \frac{n(n+1)(2n+1)}{6}$, para todo natural n.

*** b.** $1+3+5+\cdots+(2n+1) = (n+1)^2$, para todo natural n. (*Dica*: Reescreva o lado esquerdo da igualdade de forma recursiva, para depois facilitar a sua prova!)

c. $1^3 + 2^3 + \cdots + n^3 = \frac{n^2(n+1)^2}{4}$, para todo natural n.

d. $n^2 < 2^n$, para todo natural $n \geq 5$.

e. $n^2 < n!$, para todo natural $n \geq 4$.

f. $n^3 - n$ é divisível por 3, para todo natural n.

*** g.** $\mathrm{mdc}(F(n), F(n+1)) = 1$, para todo natural n, onde $F(n)$ é o n-ésimo número de Fibonacci.

h. $3^{n+1} - 2$ é ímpar, para todo natural n.

***Questão 4.** Encontre uma fórmula fechada (i.e., uma fórmula não-recursiva e que não use reticências, somatórios ou produtórios) para a seguinte expressão (em função de n) e depois prove (por indução) que a fórmula encontrada está correta para todo natural n:

$$\frac{1}{1 \cdot 2} + \frac{1}{2 \cdot 3} + \cdots + \frac{1}{(n+1)(n+2)}$$

Questão 5. Seja $f : \mathbb{N} \to \mathbb{N}$ uma função definida recursivamente:

$$\begin{cases} f(0) = 3 \\ f(k) = 7 \cdot f(k-1) & \text{se } k \geq 2 \end{cases}$$

Prove que $f(n) = 3 \cdot 7^n$ para todos os naturais n.

Questão 6. Seja $g : \mathbb{N} \to \mathbb{N}$ uma função definida recursivamente:

$$\begin{cases} g(0) = 1 \\ g(1) = 3 \\ g(k) = g(k-2) + 2 \cdot g(k-1) \quad \text{se } k \geq 3 \end{cases}$$

Prove que $g(n)$ é ímpar para todos os naturais n.

***Questão 7.** São dadas 3^n moedas de um real, uma das quais foi adulterada e pesa menos do que devia. Você tem uma balança de dois pratos mas não tem pesos; a única forma de pesagem permitida consiste em pôr algumas moedas em cada prato e verificar se a balança está equilibrada. Mostre, por indução, que n pesagens deste tipo são suficientes para achar a moeda adulterada.

Questão 8. Vamos denotar o n-ésimo primo por p_n, começando a contagem em $n = 0$. Assim $p_0 = 2$, $p_1 = 3$, $p_2 = 5$, etc. O objetivo ao final desta questão é achar um limite superior para o n-ésimo primo em função de n.

*** a.** Mostre que $p_{n+1} \leq (p_0 \cdot p_1 \cdots p_n) + 1$. (*Dica*: $(p_0 \cdot p_1 \cdots p_n) + 1$ é um número natural maior ou igual a 2, logo tem algum fator primo)

*** b.** Use indução e o item anterior para mostrar que o n-ésimo número primo satisfaz a desigualdade $p_n \leq 2^{(2^n)}$.

Questão 9. Prove que qualquer número natural $n \geq 8$ pode ser escrito como uma soma onde todas as parcelas são 3 ou 5 (por exemplo, $11 = 3 + 3 + 5$).