**Questão 1 – 2 pontos**

Chamamos de binômio de Newton todo o binômio da forma , sendo um número natural. Os termos do polinômio resultante são dados por:

Onde o coeficiente

**Tarefa:**

Escreva uma função **Python** chamada **sbin**, que recebe como parâmetro de entrada um número natural , correspondente ao expoente do binômio de Newton, e retorna como parâmetro de saída o resultado da soma da multiplicação de cada coeficiente pelo quadrado de seu índice. Ou seja,

**Questão 2[[1]](#footnote-1) – 4 pontos**

A antiga raça de Gulamatu é muito avançada no seu esquema de cálculo dos anos. Eles entendem o que é ano bissexto (ano que é divisível por 4 e não é divisível por 100, com a ressalva de que anos que são divisíveis por 400 são também anos bissextos). Em alguns anos, ocorrem festivais. Nos anos divisíveis por 15, acontece o Festival Huluculu, e nos anos bissextos divisíveis por 55 acontece o Festival Bulukulu.

Com base nisto, podemos criar um escore a atribuir a um ano, da seguinte maneira:

* Se um ano é bissexto, soma-se 1 ao escore;
* Se neste ano ocorre o Festival Huluculu, soma-se 10 ao escore;
* Se neste ano ocorre o Festival Bulukulu, soma-se 100 ao escore.

Assim, por exemplo, se for possível que em um ano bissexto aconteçam ambos os festivais, o escore será 1+10+100=111. Se acontecer apenas o festival Hulukulu, o escore será 10. Em anos simplesmente bissextos, o escore será 1, e em anos normais, o escore será zero.

**Tarefas:**

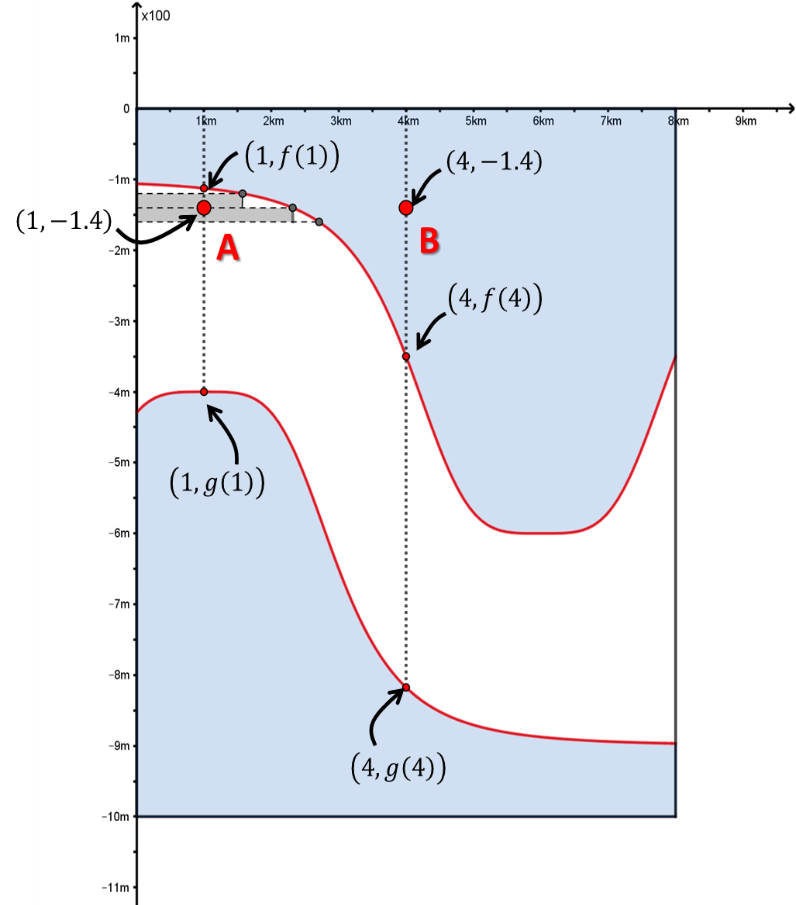
1. (2 pontos) Escreva uma função **Python** chamada **EscoreGulumatu**, que recebe como parâmetro de entrada um número natural , com até quatro dígitos, correspondente ao ano que se deseja verificar, e retorna como parâmetro de saída o escore do ano, calculado conforme descrito acima;
2. (2 pontos) Escreva uma função **Python** chamada **ContaGulumatu**, que recebe como parâmetros de entrada dois números e , com até quatro dígitos, correspondentes a um intervalo de anos, e retorna como parâmetros de saída, nesta ordem: (i) a quantidade de anos bissextos; (ii) a quantidade de anos com Festival Huluculu; (iii) a quantidade de anos com Festival Bulukulu; e (iv) a quantidade de anos com ambos os festivais.

**Questão 3[[2]](#footnote-2) – 4 pontos**

A mineradora Xingue Lingue faz prospecções avançadas para extração de minério. Para saber se um terreno é lucrativo, ela precisa estimar a quantidade de minério que poderá extrair. O minério se encontra em uma faixa do solo, entre camadas de terra, como na figura abaixo, que representa um corte vertical num terreno com 8 km de comprimento e 1.000 metros de profundidade (cada marca na escala vertical corresponde a 100 metros). Podemos considerar que a distribuição do minério no solo só depende do comprimento e da profundidade, se mantendo constante ao longo da largura do terreno). A mineradora fez intensas pesquisas e conseguiu determinar as funções que definem as duas interfaces entre as camadas de terra e a camada de minério.

;

Para estimar a quantidade de minério, o método que a Xingue Lingue pretende implementar é um cálculo iterativo aproximado, que calcule a quantidade de minério em faixas de 20 metros de profundidade.

Partindo do pressuposto de que o perfil é constante ao longo da largura do terreno, é possível considerar os cálculos apenas para as áreas estimadas em um corte vertical. Os valores informados, portanto, referem-se ao custo e à receita a cada quilômetro de largura

A escavação do comprimento total (8km) de cada faixa de 20 metros de profundidade acarreta num custo operacional de R$ 250.000,00, e cada quilômetro de minério proporciona uma receita líquida de R$ 90.000,00. A lucratividade de um terreno depende da quantidade de minério que se consegue encontrar.

Assim, se numa faixa forem encontrados 3,5km de minério, o custo desta faixa será de R$ 250.000,00 e a receita será de R$ 315.000,00.

O método consiste em dividir o terreno em faixas horizontais de 20 metros de profundidade (ou seja, vamos ter 50 faixas, indo de 20 a 1000 metros de profundidade: desde até , com intervalos ). Para cada faixa de profundidade, precisamos verificar onde o terreno tem minério e onde tem apenas terra. Para isso, em cada faixa de profundidade vamos “marcar” o comprimento terreno em pontos a cada 20 metros (desde até , com intervalos ). Portanto, cada ponto será identificado por uma coordenada .

Com estas coordenadas, podemos analisar a composição em cada ponto do terreno, substituindo o valor de nas funções delimitadoras e , e calculando os limites de cada faixa. Se a profundidade do ponto analisado () está entre os limites das duas funções para o valor de (ou seja, e ), significa que ele está localizado dentro da faixa de minério. Se estiver antes do primeiro limite ou depois do segundo, ele está fora da faixa de minério, e corresponde a um ponto onde só tem terra.

Por exemplo, na figura, os pontos A e B estão ambos a uma profundidade de 140 metros (). Dentro da faixa de profundidade de 140 metros, o ponto A está numa posição a 1km de comprimento () e o ponto B está a 4 km ().

Como , pode-se concluir que A está na faixa de minério. Já no ponto B, temos que , ou seja, B está antes do primeiro limite da faixa de minério naquele comprimento do terreno, portanto B está fora da faixa de minério.

Ao se identificar que um ponto pertencente à faixa de minério, podemos, de maneira aproximada, considerar que os próximos 20 metros (correspondente ao pedaço de terreno daquele ponto) também pertencem à faixa. Assim, a cada ponto localizado na faixa de minério, somamos 20 metros à quantidade total de minério encontrada.

Para calcular a receita esperada decorrente da quantidade de minério em um corte vertical do terreno, multiplicamos o comprimento total de minério encontrado (em quilômetros) pela receita por quilômetro (R$ 90.000,00). O custo total vai ser igual a quantidade de faixas de terreno escavadas multiplicada pelo custo de cada faixa (R$ 250.000,00).

**Tarefas:**

1. (2 pontos) Escreva uma função **Python** chamada **minerio**, que recebe como parâmetro de entrada um número real entre 0 e 10, correspondente à profundidade do terreno que se deseja avaliar; e retorna como parâmetro de saída o lucro esperado da escavação do terreno até esta profundidade, de acordo com as especificações descritas anteriormente;
2. (2 pontos) É possível que, a partir de um certo ponto, não seja mais interessante continuar a escavar, pois os custos superariam as receitas e a lucratividade total tenderia a se reduzir.

Para avaliar isto, escreva uma função **Python** chamada **lmax**, que **não recebe** parâmetros de entrada, e retorna como parâmetros de saída: (i) a profundidade de escavação que possibilita obter o maior lucro possível; e (ii) o lucro máximo que é possível obter.

**Questão 4 – 2,5 pontos**

Dependendo do valor de , a série abaixo pode ser convergente ou divergente.

**Tarefa:**

Escreva uma função **Python** chamada **serie**, que recebe **dois parâmetros de entrada**: (i) o valor de ; e (ii) um valor de precisão para seu cálculo, que será usado como critério de parada; e retorne **um parâmetro de saída**, duas possibilidades: a) o valor para o qual a série converge, caso ela seja convergente; ou b) o valor , caso sejam somados 100 termos e a série não tenha chegado a um valor de convergência.

**Questão 5 – 2,5 pontos**

Dois números inteiros e são chamados de equivalentes quando a soma dos divisores inteiros positivos de coincide com a soma dos divisores inteiros positivos de .

**Tarefa:**

Escreva uma função **Python** chamada **equivalentes**, que recebe como **parâmetros de entrada** os dois números, e retorne como **parâmetro de saída** duas possibilidades: 1, caso os números sejam equivalentes; ou 0 , caso contrário.

**Questão 6 – 2,5 pontos**

Três segmentos de reta podem constituir os lados de um triângulo, caso a medida de qualquer um deles seja menor que a soma das medidas dos outros dois. No caso de poderem ser os lados de um triângulo, este pode ser: equilátero (quando todos os lados são iguais), isósceles (quando dois lados são iguais e diferentes do terceiro), ou escaleno (quando todos os lados são diferentes).

**Tarefa:**

Escreva uma função **Python** chamada **triangulo** que recebe como **parâmetros de entrada** três valores numéricos, e testa se estes valores podem ser as medidas dos lados de um triângulo e, caso possam, o tipo de triângulo que eles formam, retornando como **parâmetros de saída** um código numérico de um algarismo, de acordo com a seguinte regra: 0: valores não formam os lados de um triângulo; 1: valores formam um triângulo equilátero; 2: valores formam um triângulo isósceles; 3: valores formam um triângulo escaleno.

**Questão 7 – 2,5 pontos**

Considere a seguinte série matemática:

Sabe-se (vocês estudaram Cálculo...) que esta soma é...

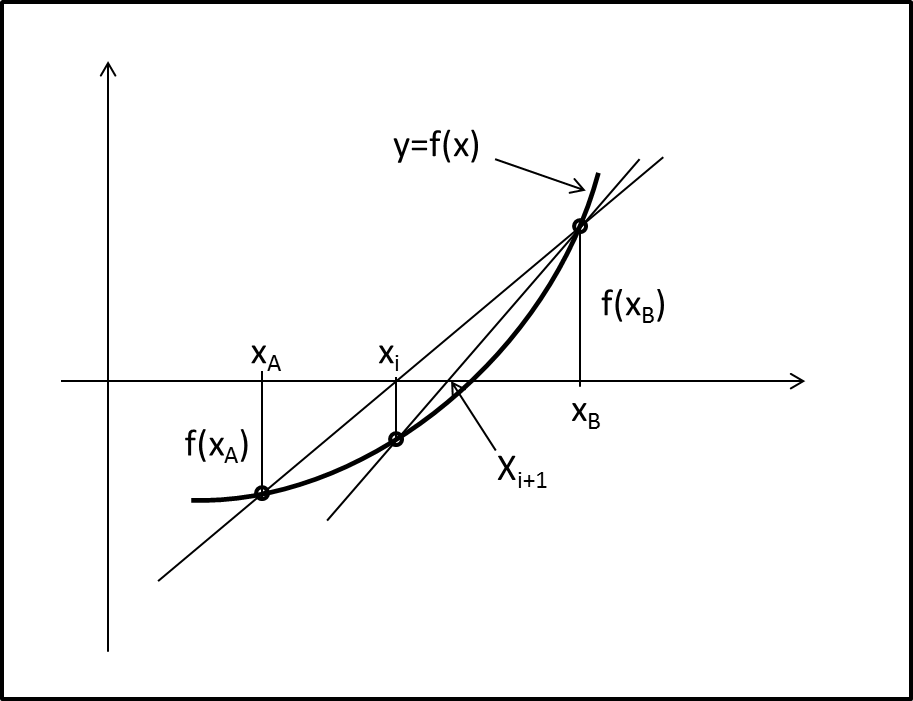
* *Convergente* (ou seja, ela tem um *limite*, aproxima-se de um certo valor quando a quantidade de termos cresce “bastante”) quando o valor de é tal que .
* *Divergente* (ou seja, ela **não** tem um *limite*, não se aproxima de qualquer valor quando a quantidade de termos cresce “bastante”) quando o valor de assume valores tais que ou

**Tarefa:**

Escreva uma função **Python** chamada **serie** que recebe dois **parâmetros de entrada**: e , e retorne, como **parâmetro de saída**, uma destas duas alternativas: (i) O limite da série, com uma precisão de 0,000001, caso ela seja convergente; ou (ii) quantos termos são necessários para que a soma ultrapasse um valor equivalente ao cubo do valor informado (), caso ela seja divergente.

**Questão 8 – 2,5 pontos**

Existem vários métodos para encontrar o valor de uma raiz de uma função. Quando se tem uma ideia de um intervalo que contenha uma raiz, é possível usar um método iterativo chamado “falsa posição”, como descrito a seguir.



Sejam xA e xB os pontos que definem o intervalo onde uma raiz está localizada. Nesta situação, f(xA) e f(xB) têm sinais opostos. Uma aproximação para o valor da raiz será o ponto onde a reta definida pelos pontos (xA, f(xA)) e (xB, f(xB)) corta o eixo das abscissas. Com este ponto, podemos definir um novo intervalo, menor que o anterior, onde a raiz se localiza, substituindo o ponto original cujo valor da função tem o mesmo sinal deste novo ponto (o que indica que a raiz está entre o novo ponto e o outro ponto original). Este processo continua até que o valor da função correspondente ao ponto encontrado seja muito próximo a zero, para uma precisão definida.

Para os pontos xA e xB que definem o intervalo, o ponto xi pode ser calculado com a fórmula:

Então, o método pode ser resumido como:

* Dados dois pontos xA e xB que definem um intervalo onde se localiza uma raiz
* Encontrar um próximo ponto xi pela fórmula acima
* Substituir as coordenadas do ponto original cujo valor da função tenha o mesmo sinal que o valor da função no novo ponto, definindo os novos limites do intervalo
* Repetir enquanto o valor da função no novo ponto for maior que uma precisão pré‑definida

Escreva uma função **Python** chamada **FalsaPosicao** que recebe quatro parâmetros (**a**, **b**, **c** e **precisão**) e encontra a raiz da função **y=ax2+bx+c** com uma precisão determinada, pela implementação do método acima descrito.

Observação: Use a função **intervalo**, descrita abaixo, para encontrar os limites do intervalo inicial a ser usado no método.

def intervalo(a,b,c):

#Estima um intervalo que contém as raízes de uma função quadrática

#Se a função não tiver raízes, a função retorna um valor infinito

#Se a função tiver apenas uma raiz, a função retornará dois valores

# iguais à coordenada do vértice

delta=b\*\*2-4\*a\*c

if delta<0:

xa=inf

xb=inf

else:

xa=(-b)/(2\*a) #o primeiro ponto é o vértice da função

xb=xa+1.5\*sqrt(delta)/(2\*a) #o segundo ponto está a 1,5 delta do

#primeiro

return xa,xb

1. Adaptado de URI Online Judge: problema 1279 (<https://www.urionlinejudge.com.br/judge/pt/problems/view/1279>) acessado em 17/09/2018 [↑](#footnote-ref-1)
2. Adaptado de URI Online Judge: problema 1297 (<https://www.urionlinejudge.com.br/judge/pt/problems/view/1297>) [↑](#footnote-ref-2)