



Escola Politécnica da PUC-Campinas

Faculdade de Análise de Sistemas

Curso de Sistemas de Informação

Algoritmos de Programação, Projetos e Computação

Exercícios de Repetição

1º Semestre de 2023

Prof. André Luís dos R.G. de Carvalho

Melhorias

Exercícios de 1 a 47

Refaça os exercícios de 1 a 47 da segunda lista de exercícios (exercícios de *if e try*), agora, oferecendo a oportunidade ao usuário de redigitar, quando tiver feito uma digitação inválida, bem como, oferecendo ao usuário a possibilidade de realizar a(s) funcionalidade(s) disponibilizada(s) tantas vezes quantas julgar necessário, isso sem a necessidade de reexecutar o programa.

Básicos

Em todos os exercícios abaixo, ofereça ao usuário a oportunidade de redigitar, quando tiver feito uma digitação inválida, bem como, a possibilidade de realizar a(s) funcionalidade(s) disponibilizada(s) tantas vezes quantas julgar necessário, isso sem a necessidade de reexecutar o programa.

48. Escreva um programa em Python que solicita a digitação de dois números inteiros que devem delimitar um intervalo de números; seu programa deve então exibir os termos entre eles da sequência oscilante com primeiro termo igual ao limite inferior do intervalo e com seus demais termos obtidos alternadamente somando 4 e subtraindo 2 do termo anterior.
 49. Escreva um programa em Python que calcula e exibe a soma dos divisores de um número natural dado.
 50. Escreva um programa em Python que solicita a digitação de um número natural, verifica se o mesmo é primo (número divisível apenas por 1 e por si próprio), exibindo na tela uma mensagem diagnóstica.
 51. Escreva um programa em Python que solicita a digitação dos limites de um intervalo de números naturais, exibindo então na tela todos os números naturais do intervalo (incluindo os limites) que são primos.
 52. Escreva um programa em Python que solicita a digitação de um número natural, verifica se o mesmo é perfeito (número que é igual à soma de seus divisores distintos de si próprio), exibindo na tela uma mensagem diagnóstica (6 é; 10 não é).
 53. Escreva um programa em Python que solicita a digitação dos limites de um intervalo de números naturais, exibindo então na tela todos os números naturais do intervalo (incluindo os limites) que são perfeitos.
 54. Escreva um programa em Python que solicita a digitação de dois números naturais, verifica se o mesmos são amigos (ambos tem a soma de seus divisores distintos de si próprio igual ao outro), exibindo na tela uma mensagem diagnóstica (220 e 284 são; 123 e 321 não são).
 55. Escreva um programa em Python que solicita a digitação dos limites de um intervalo de números naturais, exibindo então na tela todos os pares de números naturais do intervalo (incluindo os limites) que são amigos.
-

56. Escreva em Python um programa que calcula e mostra o número natural que é o inverso (tem os mesmos dígitos, porém em ordem reversa) de um número natural dado.
 57. Escreva um programa em Python que solicita a digitação de um número natural, verifica se o mesmo é palíndromo (número cujos dígitos lidos da esquerda para direita ou da direita para a esquerda se equivalem), exibindo na tela uma mensagem diagnóstica.
-

Concretos

Em todos os exercícios abaixo, ofereça ao usuário a oportunidade de redigitar, quando tiver feito uma digitação inválida, bem como, a possibilidade de realizar a(s) funcionalidade(s) disponibilizada(s) tantas vezes quantas julgar necessário, isso sem a necessidade de reexecutar o programa. Detalhe: jamais use na resolução desses exercícios a operação de multiplicação entre um número e um texto, o que geraria a replicação do texto em questão.

58. Faça um programa em Python que solicita a digitação de um número natural e, em seguida, escreve na tela uma quantidade de linhas igual ao número digitado, conforme a figura (retângulo sólido) abaixo:

Número Digitado	Linhas Escritas Na Tela
≤ 1	Indicar erro
2	oo oo
3	ooo ooo ooo
4	oooo oooo oooo oooo
5	ooooo ooooo ooooo ooooo ooooo
6	oooooo oooooo oooooo oooooo oooooo oooooo
7	ooooooo ooooooo ooooooo ooooooo ooooooo ooooooo ooooooo
etc	etc

59. Faça um programa em Python que solicita a digitação de um número natural e, em seguida, escreve na tela uma quantidade de linhas igual ao número digitado, conforme a figura (retângulo vazado) abaixo:

Número Digitado	Linhas Escritas Na Tela
≤ 1	Indicar erro
2	<pre> OO OO </pre>
3	<pre> OOO O O OOO </pre>
4	<pre> OOOO O O O O OOOO </pre>
5	<pre> OOOOO O O O O O O OOOOO </pre>
6	<pre> OOOOOO O O O O O O O O OOOOOO </pre>
7	<pre> OOOOOOO O O O O O O O O O O OOOOOOO </pre>
8	<pre> OOOOOOOO O O O O O O O O O O O O OOOOOOOO </pre>
etc	etc

60. Faça um programa em Python que solicita a digitação de um número natural e, em seguida, escreve na tela uma quantidade de linhas igual ao número digitado, conforme a figura (triângulo sólido) abaixo:

Número Digitado	Linhas Escritas Na Tela
≤ 1	Indicar erro
2	<pre> O 000</pre>
3	<pre> O 000 00000</pre>
4	<pre> O 000 00000 0000000</pre>
5	<pre> O 000 00000 0000000 000000000</pre>
6	<pre> O 000 00000 0000000 000000000 00000000000</pre>
7	<pre> O 000 00000 0000000 000000000 00000000000 0000000000000</pre>
etc	etc

61. Faça um programa em Python que solicita a digitação de um número natural e, em seguida, escreve na tela uma quantidade de linhas igual ao número digitado, conforme a figura (triângulo vazado) abaixo:

Número Digitado	Linhas Escritas Na Tela
≤ 1	Indicar erro
2	<pre> o oo </pre>
3	<pre> o oo ooo </pre>
4	<pre> o oo ooo oooo oooooo </pre>
5	<pre> o oo ooo oooo ooooo oooooo </pre>
6	<pre> o oo ooo oooo ooooo oooooo ooooooo </pre>
7	<pre> o oo ooo oooo ooooo oooooo ooooooo ooooooo </pre>
8	<pre> o oo ooo oooo ooooo oooooo ooooooo ooooooo ooooooo </pre>
etc	etc

62. Faça um programa em Python que solicita a digitação de um número natural e, em seguida, escreve na tela uma quantidade de linhas igual ao número digitado, conforme a figura (losango sólido) abaixo:

Número Digitado	Linhas Escritas Na Tela
≤ 2	Indicar erro
3	<pre> 0 000 000 </pre>
4	Indicar erro
5	<pre> 0 000 00000 000000 0000000 </pre>
6	Indicar erro
7	<pre> 0 000 00000 0000000 00000000 000000000 </pre>
8	Indicar erro
9	<pre> 0 000 00000 0000000 000000000 0000000000 00000000000 </pre>
etc	etc

63. Faça um programa em Python que solicita a digitação de um número natural e, em seguida, escreve na tela uma quantidade de linhas igual ao número digitado, conforme a figura (losango vazado) abaixo:

Número Digitado	Linhas Escritas Na Tela
≤ 2	Indicar erro
3	<pre> o o o o</pre>
4	Indicar erro
5	<pre> o o o o o o o o</pre>
6	Indicar erro
7	<pre> o o o o o o o o o o</pre>
8	Indicar erro
9	<pre> o o o o o o o o o o o o</pre>
etc	etc

Sobre bits em bytes

Em todos os exercícios abaixo, ofereça ao usuário a oportunidade de redigitar, quando tiver feito uma digitação inválida, bem como, a possibilidade de realizar a(s) funcionalidade(s) disponibilizada(s) tantas vezes quantas julgar necessário, isso sem a necessidade de reexecutar o programa.

64. Escreva um programa em Python que solicita a digitação de dois números naturais, rodando os bits do primeiro número digitado para a esquerda tantas vezes quanto expressar o segundo número digitado, exibindo na tela o número natural resultante. Assuma que o número rodado é um número de 4 bytes.
65. Escreva um programa em Python que solicita a digitação de dois números naturais, rodando os bits do primeiro número digitado para a direita tantas vezes quanto expressar o segundo número digitado, exibindo na tela o número natural resultante. Assuma que o número rodado é um número de 4 bytes.
66. Escreva um programa em Python que solicita a digitação de dois números naturais, tornando 1 no primeiro número digitado o bit expresso pelo segundo número digitado, exibindo então na tela o número natural resultante. Assuma que o número que tem o bit ajustado é um número de 4 bytes.
67. Escreva um programa em Python que solicita a digitação de dois números naturais, tornando 0 no primeiro número digitado o bit expresso pelo segundo número digitado, exibindo então na tela o número natural resultante. Assuma que o número que tem o bit ajustado é um número de 4 bytes.