

Iniciado em	terça, 3 ago 2021, 16:13
Estado	Finalizada
Concluída em	domingo, 15 ago 2021, 11:55
Tempo empregado	11 dias 19 horas
Avaliar	10,00 de um máximo de 10,00(100%)

Aviso!



Questão 1

Correto

Atingiu 1,00 de 1,00

Tá restando soma

Nesta questão, você deve criar um [Programa](#) que recebe três números, a , b e d e imprime o resto da divisão de $a + b$ por d . A [função](#) responsável por esse cálculo deve chamar `resto` que irá receber esses três valores.

Entrada

A entrada consiste de 3 linhas, com cada uma contendo 3 inteiros a , b , $d > 0$, separadas por espaço. Os números devem ser informados pelo usuário, obedecendo a formatação de [input](#) dos testes apresentados.

Saída

Para cada um dos três números, a chamada da [função](#) deverá imprimir resto da divisão de $(a + b)$ por d .

Particularidade do Tópico

Atenção, a criação de uma [função](#) com o nome determinado pelo enunciado é fundamental para a prática do aluno e o Moodle irá descontar pontos, caso a criação não tenha sido feita corretamente (sendo *case-sensitive* o nome da [função](#)).

Observações

- No primeiro exemplo de teste, os resultados obtidos são:
 - Para a primeira consulta: $(1 + 2) \% 3 = 3 \% 3 = 0$
 - Para a segunda consulta: $(3 + 2) \% 2 = 5 \% 3 = 1$
 - Para a terceira consulta: $(5 + 10) \% 20 = 15 \% 20 = 15$

For example:

Input	Result
1 2 3	0
3 2 2	1
5 10 20	15

Aviso!

Input	Result
8 9 10	7
5 5 5	0
555 445 999	1
362 11 325	48
568 459 551	476
800 502 144	6

Answer: (penalty regime: 0, 0, 10, 20, ... %)

```

1 def resto(a, b, d):
2     a = int(a)
3     b = int(b)
4     d = int(d)
5     print ((a+b) % d)
6
7 for x in range(3):
8     entradas = input().split(' ')
9     resto(entradas[0], entradas[1], entradas[2])
10

```

	Input	Expected	Got	
✓	1 2 3	0	0	✓
	3 2 2	1	1	
	5 10 20	15	15	

Aviso!

?

	Input	Expected	Got	
✓	8 9 10 5 5 5 555 445 999	7 0 1	7 0 1	✓
✓	362 11 325 568 459 551 800 502 144	48 476 6	48 476 6	✓
✓	21 530 756 75 983 987 407 811 20	551 71 18	551 71 18	✓
✓	322 295 343 619 534 689 916 989 310	274 464 45	274 464 45	✓
✓	150 981 529 373 339 149 932 328 501	73 116 258	73 116 258	✓

Passou em todos os teste! ✓

Para resolver a questão para cada caso de teste, basta somar os valores de a e b e calcular o resto da divisão desta soma por d . Ao final, deve-se imprimir o valor do resultado.

Assim, deve-se processar cada linha da entrada, realizando a leitura das três variáveis a , b e d separadas por espaço em branco, e chamando a [função](#) supracitada.

Question author's solution (Python3):

```

1 def resto(a, b, m):
2     print((a + b) % m)
3
4
5 a, b, m = [int(x) for x in input().split()]
6 resto(a, b, m)

```

Aviso!

```

a, b, m = [int(x) for x in input().split()]

```

?

```
9 resto(a, b, m)
10
11 a, b, m = [int(x) for x in input().split()]
12 resto(a, b, m)
```

Correto

Notas para este envio: 1,00/1,00.

Aviso!

?

Questão 2

Correto

Atingiu 1,00 de 1,00

Conversão v2

Quando Sam e Jean estão viajando, eles sempre veem esses terminais que mostram a temperatura do local onde eles se encontram. Um problema muito comum que causa diversas confusões é que dependendo do lugar, a temperatura é exibida em graus Celsius e em outros lugares a temperatura é exibida em Fahrenheit. Atualmente eles estão viajando pela América do Sul onde é comum medir a temperatura em Celsius. Escreva uma [função](#) chamada **converte** que recebe uma temperatura em ponto flutuante e imprima o resultado após a conversão para Fahrenheit.

Entrada

Essa [função](#) recebe como entrada o valor da temperatura em ponto flutuante.

Saída

Imprimir o valor convertido de Celsius para Fahrenheit. O valor convertido deve ser apresentado com somente uma casa decimal.

Observação

Submeta somente o que foi solicitado.

Particularidade do Tópico

Atenção, a criação de uma [função](#) com o nome determinado pelo enunciado é fundamental para a prática do aluno e o Moodle irá descontar pontos, caso a criação não tenha sido feita corretamente (sendo *case-sensitive* o nome da [função](#)).

For example:

Test	Result
<code>converte(-1.0)</code>	30.2
<code>converte(0)</code>	32.0
<code>converte(1.0)</code>	33.8

Answer (penalty regime: 0, 0, 10, 20, ... %)

Aviso!

```
f converte(temp_celsius):  
    print(f'{0*temp_celsius / 5 + 32:.1f}')
```

?

```

2     print(' {0}° temp_celsius / 5 + 32.111' )
3
4

```

	Test	Expected	Got	
✓	converte(-1.0)	30.2	30.2	✓
✓	converte(0)	32.0	32.0	✓
✓	converte(1.0)	33.8	33.8	✓
✓	converte(1.0)	33.8	33.8	✓
✓	converte(40)	104.0	104.0	✓
✓	converte(-1.0)	30.2	30.2	✓

Passou em todos os teste! ✓

Aviso! on author's solution (Python3):

?

```
1 ▾ def converte(valor):  
2   res = ((valor * 9.0) / 5.0) + 32.0  
3   print(f'{res:.1f}')
```

Correto

Notas para este envio: 1,00/1,00.

Aviso!

?

Questão 3

Correto

Atingiu 1,00 de 1,00

Cálculo de Potência v2

Implemente uma [função](#) chamada **powAPC** que calcule X^Y . Sua [função](#) deve receber os valores de X e Y, ambos inteiros, e imprimir o valor de X^Y em ponto flutuante.

Entrada

Dois valores inteiros X e Y.

Saída

O resultado da potenciação X^Y em ponto flutuante.

Observações

- Utilize a [função](#) pow da [biblioteca](#) math.
- Submeta somente o que foi solicitado.
- Não é necessário se preocupar com o número de casas decimais no número flutuante.

Particularidade do Tópico

Atenção, a criação de uma [função](#) com o nome determinado pelo enunciado é fundamental para a prática do aluno e o Moodle irá descontar pontos caso a criação não tenha sido feita corretamente (sendo *case-sensitive* o nome da [função](#)).

For example:

Test	Result
powAPC(2, 2)	4.0
powAPC(2, 3)	8.0
powAPC(2, 4)	16.0

Answer (penalty regime: 0, 0, 10, 20, ... %)

Aviso! port math

def powAPC(x, y):

```

2 def powAPC(x, y):
3     print(float(math.pow(x, y)))
4

```

	Test	Expected	Got	
✓	powAPC(2, 2)	4.0	4.0	✓
✓	powAPC(2, 3)	8.0	8.0	✓
✓	powAPC(2, 4)	16.0	16.0	✓
✓	powAPC(4, 2)	16.0	16.0	✓
✓	powAPC(5, 5)	3125.0	3125.0	✓
✓	powAPC(10, 10)	10000000000.0	10000000000.0	✓

Passou em todos os teste! ✓

Aviso! You author's solution (Python3):

?

```
1 | import math
2 |
3 | def powAPC(x, y):
4 |     print(math.pow(x, y))
```

Correto

Notas para este envio: 1,00/1,00.

Aviso!

?

Questão 4

Correto

Atingiu 1,00 de 1,00

AProCedimento

Escreva uma [função](#) chamada **imprimeAPC** que deverá imprimir uma palavra usando caracteres ASCII.

Entrada

Não há.

Saída

A impressão da palavra APC, utilizando ASCII Art, de acordo com o exemplo.


Observação

- Submeta somente o que foi solicitado.
- A [função](#) não recebe argumentos.

Particularidade do Tópico

Atenção, a criação de uma [função](#) com o nome determinado pelo enunciado é fundamental para a prática do aluno e o Moodle irá descontar pontos caso a criação não tenha sido feita corretamente (sendo *case-sensitive* o nome da [função](#)).

For example:

Test	Result
<code>imprimeAPC()</code>	

Aviso!





penalty regime: 0, 0, 10, 20, ... %)

?

```



1
2 def imprimeAPC():
3     print('
4     print('
5     print('
6     print('
7     print('
8
9

```

	Test	Expected	Got	
✓	imprimeAPC()			✓
✓	imprimeAPC() imprimeAPC()			✓

Aviso!

?

	Test	Expected	Got	
✓	<pre>imprimeAPC() imprimeAPC() imprimeAPC()</pre>			✓

Passou em todos os teste! ✓

Question author's solution (Python3):

```

1 def imprimeAPC():
2     print('
3     print('
4     print('
5     print('

```

Correto

Notas para este envio: 1,00/1,00.

Aviso!

?

Questão 5

Correto

Atingiu 1,00 de 1,00

Vestimentas country v1

Bonifácio é fã de moda country e isso fez com que ele adquirisse uma extensa coleção de botas e chapéus de diferentes modelos. Em seu dia-a-dia, Bonifácio utiliza um jogo de vestimentas country, composto por duas peças: um chapéu e um par de botas. Agora Bonifácio precisa viajar para uma conferência no exterior e planeja incluir em sua bagagem o máximo de jogos de vestimentas country que puder.

Durante sua viagem, Bonifácio planeja utilizar todos os seus jogos de vestimenta country, mas com a restrição de que ele veste um jogo apenas uma única vez e que não há distinção entre os chapéus e os pares de botas.

Sabe-se que Bonifácio possui em sua coleção de vestimentas x pares de botas e y chapéus, elabore uma [função vestimentas](#) que receba os dois números inteiros x e y como parâmetros e calcule a quantidade máxima de jogos de pares de botas e chapéus m que Bonifácio levará em sua bagagem para utilizar nos dias em que estiver viajando.

Entrada

A entrada compreende os parâmetros da [função vestimentas](#), que são dois números inteiros x e y ($1 \leq x \leq 10^2, 1 \leq y \leq 10^2$) associados à quantidade x de pares de botas e y de chapéus, respectivamente.

Saída

A [função vestimentas](#) deve imprimir a quantidade máxima de jogos de vestimentas country que Bonifácio poderá levar na sua bagagem.

Observações

- No primeiro caso de teste, Bonifácio possui 2 chapéus e 2 pares de botas. Ele pode pegar um chapéu (qualquer um dos dois chapéus) e um par de botas (qualquer um dos pares de botas) e formar um jogo de vestimentas para ser utilizado em um dia. No dia seguinte, Bonifácio pega o chapéu e o par de botas restantes, formando um novo jogo. Ao final, ele consegue formar dois jogos de vestimentas.
- No segundo caso de teste, Bonifácio possui 4 chapéus e 1 par de botas. No primeiro dia, ele poderá pegar um par de chapéu (dentre os 4 disponíveis - pode ser qualquer chapéu) e formar um jogo de vestimenta com o único par de botas disponível. Não será

Aviso!

?

possível formar outro jogo de vestimenta, pois não há outros pares de botas disponíveis. Veja que ele não repete peças já utilizadas em dias anteriores.

- No terceiro caso de teste, Bonifácio possui 6 chapéus e 9 pares de botas. A partir desses 6 chapéus, ele consegue utilizá-lo com outros 6 pares de botas apenas, sem repetir as peças de vestimentas.

For example:

Test	Result
vestimentas(2,2)	2
vestimentas(4,1)	1
vestimentas(6,9)	6

Answer: (penalty regime: 0, 0, 10, 20, ... %)

```
1 ▾ def vestimentas(botas, chapéus):  
2   print(min(botas, chapéus))
```

Aviso!

st

Expected

Got

?

	Test	Expected	Got	
✓	vestimentas(2,2)	2	2	✓
✓	vestimentas(4,1)	1	1	✓
✓	vestimentas(6,9)	6	6	✓
✓	vestimentas(1,100)	1	1	✓
✓	vestimentas(50,14)	14	14	✓
✓	vestimentas(23,100)	23	23	✓

Passou em todos os teste! ✓

Repare que o problema não solicita que se calcule as diferentes maneiras com que as vestimentas podem ser combinadas.

Imagine que Bonifácio tenha dois chapéus e três pares de botas. Como não há distinção entre essas vestimentas, podemos pegar um chapéu (qualquer um dos dois chapéus) e um par de botas (qualquer um dos pares de botas) e formar um jogo de vestimentas para ser utilizado em um dia. No dia seguinte, Bonifácio pega o chapéu restante e um par de botas (entre dois ainda disponíveis) , formando um novo jogo. Ao final, restou apenas um par de botas, mas não temos outro chapéu disponível. Podemos formar dois jogos de vestimentas country. Mas lembre-se que , por definição do problema, cada jogo possui duas peças: um chapéu e um par de botas.

Question author's solution (Python3):

```
1 def vestimentas(x,y):  
2     print(min(x,y))
```

Correto

Notas para este envio: 1,00/1,00.

Aviso!

?

Questão 6

Correto

Atingiu 1,00 de 1,00

Vestimentas country v2

Bonifácio é fã de moda country e isso fez com que ele adquirisse uma extensa coleção de botas e chapéus de diferentes modelos. Em seu dia-a-dia, Bonifácio utiliza um jogo de vestimentas country, composto por duas peças: um chapéu e um par de botas. Agora Bonifácio precisa viajar para uma conferência no exterior e planeja incluir em sua bagagem o máximo de jogos de vestimentas country que puder.

Durante sua viagem, Bonifácio planeja utilizar todos os seus jogos de vestimenta country, mas com a restrição de que ele veste um jogo apenas uma única vez e que não há distinção entre os chapéus e os pares de botas.

Sabe-se que Bonifácio possui em sua coleção de vestimentas x pares de botas e y chapéus, elabore uma [função vestimentas](#) que receba os dois números inteiros x e y como parâmetros e calcule a quantidade máxima de peças nos jogos de vestimentas country m que Bonifácio levará em sua bagagem.

Entrada

A entrada compreende os parâmetros da [função vestimentas](#), que são dois números inteiros x e y ($1 \leq x \leq 10^2, 1 \leq y \leq 10^2$) associados à quantidade x de pares de botas e y de chapéus, respectivamente.

Saída

A [função vestimentas](#) deve imprimir a quantidade máxima de peças nos jogos de vestimentas country que Bonifácio poderá levar na sua bagagem.

Observações

- No primeiro caso de teste, Bonifácio possui 2 chapéus e 2 pares de botas. Ele pode pegar um chapéu (qualquer um dos dois chapéus) e um par de botas (qualquer um dos pares de botas) e formar um jogo de vestimentas para ser utilizado em um dia. No dia seguinte, Bonifácio pega o chapéu e o par de botas restantes, formando um novo jogo. Ao final, ele consegue formar dois jogos de vestimentas, resultando em 4 peças no total, conforme definição de que cada jogo possui 2 peças.
- No segundo caso de teste, Bonifácio possui 4 chapéus e 1 par de botas. No primeiro dia, ele poderá pegar um par de chapéu (dentre os 4 disponíveis - pode ser qualquer chapéu) e formar um jogo de vestimenta com o único par de botas disponível. Não será

Aviso!

?

possível formar outro jogo de vestimenta, pois não há outros pares de botas disponíveis. Veja que ele não repete peças já utilizadas em dias anteriores, resultando em 2 peças no total.

- No terceiro caso de teste, Bonifácio possui 6 chapéus e 9 pares de botas. A partir desses 6 chapéus, ele consegue utilizá-lo com outros 6 pares de botas apenas, sem repetir as peças de vestimentas. Assim, o resultado é 12 peças no total.

For example:

Test	Result
vestimentas(2,2)	4
vestimentas(4,1)	2
vestimentas(6,9)	12

Answer: (penalty regime: 0, 0, 10, 20, ... %)

```
1 ▾ def vestimentas(botas, chapéus):  
2   print(2*min(botas, chapéus))
```

Aviso!

?

	Test	Expected	Got	
✓	vestimentas(2,2)	4	4	✓
✓	vestimentas(4,1)	2	2	✓
✓	vestimentas(6,9)	12	12	✓
✓	vestimentas(1,100)	2	2	✓
✓	vestimentas(50,14)	28	28	✓
✓	vestimentas(23,100)	46	46	✓

Passou em todos os teste! ✓

Repare que o problema não solicita que se calcule as diferentes maneiras com que as vestimentas podem ser combinadas.

Imagine que Bonifácio tenha dois chapéus e três pares de botas. Como não há distinção entre essas vestimentas, podemos pegar um chapéu (qualquer um dos dois chapéus) e um par de botas (qualquer um dos pares de botas) e formar um jogo de vestimentas para ser utilizado em um dia. No dia seguinte, Bonifácio pega o chapéu restante e um par de botas (entre dois ainda disponíveis) , formando um novo jogo. Ao final, restou apenas um par de botas, mas não temos outro chapéu disponível. Podemos formar dois jogos de vestimentas country. Mas lembre-se que , por definição do problema, cada jogo possui duas peças: um chapéu e um par de botas.

Question author's solution (Python3):

```
1 def vestimentas(x,y):  
2     print(2*min(x,y))
```

Correto

Notas para este envio: 1,00/1,00.

Aviso!

?

Questão 7

Correto

Atingiu 1,00 de 1,00

Idade Real

Faça um [programa](#) que leia as idades de três pessoas em dias e mostre suas idades em anos, meses e dias. Considere que todo ano contém 360 dias e todo mês tem 30 dias. Você deve criar uma [função](#) de nome **age** que calcula a idade para uma ÚNICA pessoa. Essa [função](#) deve receber como parâmetro SOMENTE a quantidade de dias em uma única [variável](#).

Entrada

A entrada contém três números inteiros $1 \leq d1, d2, d3 \leq 10^6$, a idade de cada uma das 3 pessoas, em dias. Os números devem ser informados pelo usuário, obedecendo a formatação de [input](#) dos testes apresentados.

Saída

Para cada uma das três idades, você deve imprimir: Três inteiros separados por espaço **A**, **M** e **D**, representando a idade da pessoa em anos, meses e dias, respectivamente.

Observações

- Não é necessário validar se os números estão dentro do intervalo definido.
- No primeiro caso de teste, a primeira pessoa tem 15 dias de vida o que dá o total de 0 anos, 0 meses e 15 dias. A segunda pessoa tem 35 dias de vida o que dá o total de 0 anos, 1 mês e 5 dias. A terceira pessoa tem 366 dias de vida o que dá o total de 1 ano, 0 meses e 6 dias.

Particularidade do Tópico

Atenção, a criação de uma [função](#) com o nome determinado pelo enunciado é fundamental para a prática do aluno e o Moodle irá descontar pontos, por esse critério, caso a criação não tenha sido feita corretamente (sendo case-sensitive o nome da [função](#)).

For example:

Input	Result
15 35 366	0 0 15 0 1 5 1 0 6

Aviso!

Input	Result
60 90 1200000	0 2 0 0 3 0 3333 4 0
360 361 390	1 0 0 1 0 1 1 1 0

Answer: (penalty regime: 0, 0, 10, 20, ... %)

```

1 def age(age_in_days):
2     years = age_in_days // 360
3     days = age_in_days % 360
4     months = days // 30
5     days = days % 30
6
7     print(f'{years} {months} {days}')
8
9 ages = input().split(' ')
10 for x in ages:
11     age(int(x))

```

	Input	Expected	Got	
✓	15 35 366	0 0 15 0 1 5 1 0 6	0 0 15 0 1 5 1 0 6	✓

Aviso!

?

	Input	Expected	Got	
✓	60 90 1200000	0 2 0 0 3 0 3333 4 0	0 2 0 0 3 0 3333 4 0	✓
✓	360 361 390	1 0 0 1 0 1 1 1 0	1 0 0 1 0 1 1 1 0	✓
✓	1200 3050 409	3 4 0 8 5 20 1 1 19	3 4 0 8 5 20 1 1 19	✓
✓	12345 54321 33333	34 3 15 150 10 21 92 7 3	34 3 15 150 10 21 92 7 3	✓
✓	8390 7997 169	23 3 20 22 2 17 0 5 19	23 3 20 22 2 17 0 5 19	✓

Passou em todos os teste! ✓

Question author's solution (Python3):

```

1 def age(days):
2     year = days // 360
3     days = days % 360
4     month = days // 30
5     days = days % 30
6     print(year, month, days)
7
8
9 d1, d2, d3 = [int(x) for x in input().split()]
10 ...
11 Solução (para a operação acima) utilizando somente estruturas apresentadas até esta lição
    d1, d2, d3 = map(int, input().split())
12 ...

```

Aviso!

?

14

15 | 300/410



Correto

Notas para este envio: 1,00/1,00.

Aviso!

?

Questão 8

Correto

Atingiu 1,00 de 1,00

Pacotes de bolacha v2

O prof. Fagundes comprou uma caixa com m pacotes de bolacha recheada para distribuí-los igualmente entre os n alunos da sua turma de [Estruturas de Dados](#) na Universidade de Brasília (UnB). Apesar dos alunos terem adorado essa surpresa, cada um consegue comer no máximo k pacotes de bolacha, portanto, alguns desses pacotes entregues por aluno podem sobrar. A generosidade do querido professor é tamanha, tanto que ele pretende pegar esses pacotes restantes e entregá-los aos funcionários do Departamento de [Ciência da Computação](#) (CIC) da UnB.

Sabendo-se que cada aluno sempre recebe e come ao menos um pacote de bolacha, elabore uma [função](#) chamada **pacotesDeBolacha** que receba os três números inteiros m , n e k como parâmetros e imprima a quantidade máxima de pacotes de bolacha que o prof. Fagundes entregará aos alunos da sua turma.

Entrada

Não há [entrada de dados](#). A [função](#) é chamada para valores arbitrários definidos nos casos de teste que são três números inteiros m , n e k ($1 \leq n \leq 10^4$, $n \leq m \leq 10^4$, $1 \leq k \leq 10^4$) associados ao número total de pacotes de bolacha, a quantidade de alunos da turma e quantos pacotes de bolacha cada aluno dessa turma consegue comer, respectivamente.

Saída

A [função](#) **pacotesDeBolacha** deve imprimir a quantidade máxima de pacotes de bolacha que o prof. Fagundes entregará aos alunos da sua turma, respeitando as restrições descritas no enunciado.

Observação

Submeta somente o que foi solicitado. Não é necessário validar se os números estão dentro do intervalo definido.

- No primeiro caso de teste, existem 4 pacotes de bolacha, 4 alunos e cada aluno consegue comer um pacote de bolacha, totalizando 4 pacotes.
- No segundo caso de teste, existem 13 pacotes de bolacha e 5 alunos, sendo que cada aluno consegue comer $k = 2$ pacotes de bolacha. Assim, tentando dividir igualmente os 13 pacotes para os 5 alunos, temos que cada aluno poderia comer 2 pacotes, o que é possível de acordo com k . Assim, a turma toda comerá 10 pacotes.
- No terceiro caso de teste, existem 10 pacotes de bolacha e 9 alunos, sendo que cada aluno consegue comer $k = 2$ pacotes de bolacha. A divisão igualitária dos pacotes entre os alunos diz que cada aluno come 1 pacote de bolacha, resultando em 9 pacotes

Aviso!

?

para toda a turma.

Particularidade do Tópico

Atenção, a criação de uma [função](#) com o nome determinado pelo enunciado é fundamental para a prática do aluno e o Moodle irá descontar pontos, por esse critério, caso a criação não tenha sido feita corretamente (sendo *case-sensitive* o nome da [função](#)).

For example:

Test	Result
pacotesDeBolacha(4,4,1)	4
pacotesDeBolacha(13,5,2)	10
pacotesDeBolacha(10,9,2)	9

Answer: (penalty regime: 0, 0, 10, 20, ... %)

```
1 |
2 | def pacotesDeBolacha(qtd_pacotes, qtd_alunos, max_por_aluno):
3 |
4 |     qtd_por_aluno = qtd_pacotes // qtd_alunos
5 |     sobram = qtd_pacotes % qtd_alunos
6 |     total = qtd_pacotes - sobram
7 |     if qtd_por_aluno > max_por_aluno:
8 |         total -= qtd_alunos*(qtd_por_aluno-max_por_aluno)
9 |
10 |     print(total)
11 |
12 |
```

Aviso!

?

	Test	Expected	Got	
✓	pacotesDeBolacha(4,4,1)	4	4	✓
✓	pacotesDeBolacha(13,5,2)	10	10	✓
✓	pacotesDeBolacha(10,9,2)	9	9	✓
✓	pacotesDeBolacha(60,20,3)	60	60	✓
✓	pacotesDeBolacha(4,4,1)	4	4	✓
✓	pacotesDeBolacha(23,6,6)	18	18	✓

Passou em todos os teste! ✓

Para resolver o problema, temos que levar em consideração que queremos determinar a quantidade **mínima** de pacotes de bolacha que serão entregues aos funcionários do CIC. Para isso, temos que **maximizar** a quantidade de pacotes que os alunos do professor Fagundes comem. Assim, deve-se dividir igualmente os pacotes entre os alunos do prof. Caetano, respeitando a restrição que cada aluno come pelo menos um pacote, mas não come mais do que k pacotes. Finalmente, calcula-se a quantidade total de pacotes de bolachas que os alunos comem.

Question author's solution (Python3):

```
1 def pacotesDeBolacha(m,n,k):  
2     comer = min(k, m//n)  
3     total = comer * n
```

Correto

Notas para este envio: 1,00/1,00.

Aviso!

?

Questão 9

Correto

Atingiu 1,00 de 1,00

Piscina da Mansão

No ano passado, Prof. Nerynho havia publicado sua pesquisa na conferência mais aclamada do mundo ocidental, All Papers Considered (APC), e como prêmio pela maior contribuição ao mundo acadêmico, ganhou uma mansão no Lago Sul. Ele está meio entediado na pandemia e quer construir uma piscina para ficar deitado e tomar suco de laranja enquanto lê bons livros de teoria da [Ciência da Computação](#).

Seu número preferido é 3 e por isso ele gosta sempre de ter 3 opções disponíveis. Dessa forma, sua tarefa é criar um [Programa](#) para imprimir como a piscina ficaria com cada uma das três opções de comprimento. Para isso você deve implementar uma [função](#) chamada **print_rectangle** que recebe esse valor e imprime a piscina.

Entrada

A entrada consiste de 3 números diferentes guardados pelas variáveis $a, b, c \geq 3$, separadas por espaço.

Saída

Para cada um dos 3 tamanhos, a [função print_rectangle](#) deve imprimir como ficaria a piscina usando o seguinte formato:

1. O comprimento da piscina
2. As bordas da piscina usando o caractere (+).

Particularidade do Tópico

Atenção, a criação de uma [função](#) com o nome determinado pelo enunciado é fundamental para a prática do aluno e o Moodle irá descontar pontos, caso a criação não tenha sido feita corretamente (sendo *case-sensitive* o nome da [função](#)).

For example:

Input	Result
-------	--------

Aviso!

?

Input	Result
3 4 5	<pre> 3 +++ + + +++ 4 ++++ + + ++++ 5 +++++ + + +++++ </pre>
6 7 8	<pre> 6 ++++++ + + ++++++ 7 +++++++ + + +++++++ 8 ++++++++ + + ++++++++ </pre>
9 10 11	<pre> 9 +++++++ + + +++++++ 10 ++++++++ + + ++++++++ 11 ++++++++ + + ++++++++ </pre>

Aviso!

?

Answer: (penalty regime: 0, 0, 10, 20, ... %)

```

1 |
2 | def print_rectangle(size):
3 |     borda = '+'*size
4 |     print(borda)
5 |     print('+', end='')
6 |     for i in range(size-2):
7 |         print(' ', end='')
8 |     print('+')
9 |     print(borda)
10 |
11 | opcoes = input().split(' ')
12 | for x in opcoes:
13 |     print(x)
14 |     print_rectangle(int(x))

```

	Input	Expected	Got	
✓	3 4 5	3 +++ + + +++ 4 ++++ + + ++++ 5 +++++ + + +++++	3 +++ + + +++ 4 ++++ + + ++++ 5 +++++ + + +++++	✓

Aviso!

?

	Input	Expected	Got	
✓	6 7 8	<pre> 6 +++++ + + +++++ 7 +++++ + + +++++ 8 +++++ + + +++++ </pre>	<pre> 6 +++++ + + +++++ 7 +++++ + + +++++ 8 +++++ + + +++++ </pre>	✓
✓	9 10 11	<pre> 9 +++++ + + +++++ 10 +++++ + + +++++ 11 +++++ + + +++++ </pre>	<pre> 9 +++++ + + +++++ 10 +++++ + + +++++ 11 +++++ + + +++++ </pre>	✓
✓	5 5 5	<pre> 5 +++++ + + +++++ 5 +++++ + + +++++ 5 +++++ + + +++++ </pre>	<pre> 5 +++++ + + +++++ 5 +++++ + + +++++ 5 +++++ + + +++++ </pre>	✓

Aviso!

?

	Input	Expected	Got	
✓	12 16 20	<pre> 12 +++++ + + +++++ 16 +++++ + + +++++ 20 +++++ + + +++++ </pre>	<pre> 12 +++++ + + +++++ 16 +++++ + + +++++ 20 +++++ + + +++++ </pre>	✓
✓	24 30 45	<pre> 24 +++++ + + +++++ 30 +++++ + + +++++ 45 +++++ + + +++++ </pre>	<pre> 24 +++++ + + +++++ 30 +++++ + + +++++ 45 +++++ + + +++++ </pre>	✓

Passou em todos os teste! ✓

Question author's solution (Python3):

```

1 def print_rectangle(x):
2     print(x)
3     print('+ ' * x)
4     print('+ ', end='')
5     print(' ' * (x-2), end='')
6     print('+ ')
7     print('+ ' * x)

```

Aviso!


```
9 a, b, c = [int(x) for x in input().split()]
10 '''
11 Solução (para a operação acima) utilizando somente estruturas apresentadas até esta lição
12 a, b, c = map(int, input().split())
13 '''
14
```

Correto

Notas para este envio: 1,00/1,00.

Aviso!

?

Questão 10

Correto

Atingiu 1,00 de 1,00

Longevidade na UnB II

O Decanato de Graduação (DEG) está testando uma nova funcionalidade no SIGAA que permite apresentar em qual semestre um aluno da Universidade de Brasília se encontra. Para isso, o DEG aproveita as [informações](#) de ingresso na UnB existentes no próprio número de matrícula dos alunos e compara com o ano e o semestre atual. Como se sabe, os quatro dígitos mais significativos d_1, d_2, d_3 e d_4 do número de matrícula indicam o ano e o semestre de ingressos do aluno na UnB.

Por exemplo, se o número de matrícula de um estudante é 190199999, os quatro dígitos mais significativos formam o número 1901, em que adota-se a convenção ($d_1 = 1, d_2 = 9, d_3 = 0, d_4 = 1$). O número formado da combinação de d_1 e d_2 , 19, indicam que o aluno ingressou na UnB em 2019 e o $d_4 = 1$ mostra que o aluno ingressou no segundo semestre desse ano. Outro exemplo se trata do número de matrícula 180099899, que indica que estudante ingressou na UnB no primeiro semestre de 2018.

Elabore uma [função](#) chamada **qualPeriodo** que receba como parâmetros três números inteiros m, a e s associados ao número de matrícula do aluno, o ano atual e o semestre atual, respectivamente. Essa [função](#) deve calcular o semestre atual em que o aluno se encontra na UnB.

Entrada

A entrada compreende os parâmetros da [função qualPeriodo](#), que são três números inteiros m, a e s ($100000000 \leq m \leq 500000000, 0 \leq a \leq 2050, s \in \{0, 1\}$) associados ao número de matrícula do aluno, o ano atual e o semestre atual, respectivamente. É garantido que o ano e o semestre atual do aluno são maiores ou iguais ao ano e semestre de ingresso na UnB.

Saída

A [função qualPeriodo](#) deve retornar em qual semestre o aluno se encontra desde que ingressou na UnB.

Observação

Submeta somente o que foi solicitado. Não é necessário validar se os números estão dentro dos intervalos definidos.

- No primeiro caso de teste, de acordo com o número de matrícula, o aluno ingressou na UnB no primeiro semestre de 2020 e o semestre atual também é o primeiro semestre de 2020. Portanto, o aluno se encontra no primeiro semestre.

Aviso!

?

- No segundo caso de teste, de acordo com o número de matrícula, o aluno ingressou na UnB no primeiro semestre de 2020 e o semestre atual é o segundo semestre de 2020. Portanto, o aluno está no segundo semestre.

For example:

Test	Result
qualPeriodo(200012345,2020,0)	1
qualPeriodo(200054321,2020,1)	2
qualPeriodo(180134242,2020,0)	4

Answer: (penalty regime: 0, 0, 10, 20, ... %)

```
1 |
2 | def qualPeriodo(m, a, s):
3 |     ano_ingresso = 2000 + int(str(m)[0:2])
4 |     semestre_ingresso = int(str(m)[2:4])
5 |     print((a - ano_ingresso)*2 + (s - semestre_ingresso) + 1)
```

Test	Expected	Got
------	----------	-----

Aviso!

?

	Test	Expected	Got	
✓	qualPeriodo(200012345,2020,0)	1	1	✓
✓	qualPeriodo(200054321,2020,1)	2	2	✓
✓	qualPeriodo(180134242,2020,0)	4	4	✓
✓	qualPeriodo(160199999,2019,1)	7	7	✓
✓	qualPeriodo(170199999,2017,1)	1	1	✓
✓	qualPeriodo(100199999,2018,1)	17	17	✓

Passou em todos os teste! ✓

Para resolver o problema, deve-se formar um novo número d a partir dos quatro dígitos mais significativos do número de matrícula pegando-se o resultado da divisão do número por 100000. Em seguida, pode-se obter o ano a partir dos dois dígitos mais significativos de d , isto é, dividindo-se d por 100 e somando 2000. O semestre sem é obtido pegando-se o dígito menos significativo de d . A partir dos valores ano e sem é possível determinar o período atual do aluno, considerando a quantidade mínima de períodos que está matriculado na UnB.

Question author's solution (Python3):

```

1 def qualPeriodo(m,a,s):
2
3     d1 = m//100000
4     anom = 2000+(d1//100)
5     semm = (d1%100)
6
7     qtde_periodos = (a-anom)*2 + (s-semm)
8
9     periodo_atual = qtde_periodos+1

```

Aviso!

?

Correto

Notas para este envio: 1,00/1,00.



Aviso!

?