



0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9	9

Utilize caneta azul ou preta e preencha completamente a quadrícula.  
Exemplo: ■. Não use ☒.

**Turma:** (somente um número; consulte a pessoa responsável se não souber)

4     5     6     7     8     9     10     11     12     13     14     20

← Marque as quadrículas ao lado para formar o seu número USP e escreva seu nome completo em letra legível na linha pontilhada abaixão. **Se seu número possui menos que 8 dígitos complete com zeros à esquerda.**

Nome:  
.....

Esta prova tem duração de 120 minutos. Não desmonte a prova.

**Q1 [2 pontos]** Simule o código abaixo e selecione as opções correspondentes à saída impressa do programa.

```
x = 1
y = 1
i = 33
while i%3 != x%2:
    i = i+2*y
    if y<x:
        i = i+x
        y = x+y
    else:
        i = i-x
        x = x+y
    y = i//2
    if x>y:
        i = i+1
print(x,y)
```

Rascunho

Selecione o primeiro par de números impressos:

2 18     2 17     4 15     3 13     1 19     2 12     3 21

Selecione o segundo par de números impressos:

13 32     12 28     15 35     18 36     19 38     19 33     22 32

Selecione o terceiro par de números impressos ou N/A (não aplicável) se não há mais que 2 impressões:

52 56     53 51     55 55     53 52     N/A     53 56     52 59     51 53

Selecione o quarto par de números impressos ou N/A (não aplicável) se não há mais que 3 impressões:

110 85     108 82     108 86     109 82     N/A     105 84     102 86     103 83



**Q2 [2 pontos]** Cartões de crédito geralmente tem 16 dígitos. Os primeiros dígitos à esquerda estão relacionados à bandeira do cartão (Visa começa com dígito 4, Mastercard varia de 51 a 55). O último dígito (mais à direita) é um dígito verificador calculado a partir dos demais dígitos pelo algoritmo de Luhn, como medida de segurança, a fim de evitar falsificações, ou erros de digitação, conforme explicado no quadro ao lado.

Preencha as lacunas no código abaixo (L1 até L12), de forma a obter um programa que, dado um inteiro com 16 dígitos de um cartão, verifica se seu dígito verificador está correto. Caso correto, verificar também se é Mastercard.

```

cre = int(input("Digite o cartão de crédito: "))
L1
#laço da soma dos dígitos em posições pares.
while c != 0:
    L2
    if t >= 10:
        L3
        L4
        L5
    L6
    #laço da soma dos dígitos em posições ímpares.
    while L7:
        L8
        L9
    L10 #dígito verificador
    if L11:
        print("cartão válido",end=" ")
        if L12:
            print("(Mastercard)")
    else:
        print("cartão inválido:",end=" ")
        print("dígito verificador deveria ser",ver)

```

**Algoritmo de Luhn:** Exemplo para verificar se o dígito verificador 7 está correto no cartão de número 4984232050486427. Numerando os dígitos da direita para esquerda, iniciando em 1, tomamos os dígitos em posições pares (grifadas) multiplicados por dois. Porém, caso o produto resulte em algum número de 2 dígitos, estes devem ser somados. Logo, para o exemplo temos:  $2 \times 2, 6 \times 2, 4 \times 2, 5 \times 2, 2 \times 2, 2 \times 2, 8 \times 2, 4 \times 2 \Rightarrow 4, 12, 8, 10, 4, 4, 16, 8 \Rightarrow 4, 3, 8, 1, 4, 4, 7, 8$ . Calculamos então a soma dos valores obtidos, que resulta em  $4 + 3 + 8 + 1 + 4 + 4 + 7 + 8 = 39$ . A seguir, adicionamos ao somatório todos os dígitos em posições ímpares que não foram usados na etapa anterior, com exceção do primeiro à direita. Assim, temos  $39 + 9 + 4 + 3 + 0 + 0 + 8 + 4 = 39 + 28 = 67$ . Para 67 virar um múltiplo de 10 faltam 3 unidades. Então 3 é o nosso dígito verificador. Portanto, o valor 7 à direita está incorreto e o cartão 4984232050486427 é inválido.

#### Exemplos de execução do programa completo:

Digite o cartão de crédito: 5313153010488482  
cartão válido (Mastercard)

Digite o cartão de crédito: 4984232050486427  
cartão inválido: dígito verificador deveria ser 3

Para cada um dos 12 itens a seguir, correspondendo as lacunas no código acima, assinale a única resposta correta.

L1:	<input type="checkbox"/> c,soma = cre, 0	<input type="checkbox"/> c,soma = cre//10, cre	<input type="checkbox"/> c,soma = cre//100, cre%10		
	<input type="checkbox"/> c,soma = cre//10, 0	<input type="checkbox"/> c,soma = cre%10, 1			
L2:	<input type="checkbox"/> t = (c % 10)*2	<input type="checkbox"/> t = (c*2)% 10	<input type="checkbox"/> t = c % 10	<input type="checkbox"/> t = c * 2	<input type="checkbox"/> t += c % 10
L3:	<input type="checkbox"/> t = t%10 + t//10	<input type="checkbox"/> t += t%10	<input type="checkbox"/> t += t//10	<input type="checkbox"/> t = t - (t//10)*10	<input type="checkbox"/> t = t - t%10
L4:	<input type="checkbox"/> soma = t * 2	<input type="checkbox"/> soma += t	<input type="checkbox"/> soma += c * 2	<input type="checkbox"/> soma = soma + c	<input type="checkbox"/> soma = t + c
L5:	<input type="checkbox"/> c = c % 10	<input type="checkbox"/> c = c - 100	<input type="checkbox"/> c = c // 10	<input type="checkbox"/> c = c % 100	<input type="checkbox"/> c = c // 100
L6:	<input type="checkbox"/> c = cre//100	<input type="checkbox"/> c = cre%100	<input type="checkbox"/> c = c//100	<input type="checkbox"/> c = c//10	<input type="checkbox"/> c = cre//10
L7:	<input type="checkbox"/> c >= 0	<input type="checkbox"/> c < 0	<input type="checkbox"/> c <= cre	<input type="checkbox"/> c == 0	<input type="checkbox"/> c != 0
L8:	<input type="checkbox"/> soma += 10 % c	<input type="checkbox"/> soma += c % 10	<input type="checkbox"/> soma = soma % 10	<input type="checkbox"/> soma = soma + c	<input type="checkbox"/> soma += c // 10
L9:	<input type="checkbox"/> c = c - 100	<input type="checkbox"/> c = c // 10	<input type="checkbox"/> c = c % 100	<input type="checkbox"/> c = c % 10	<input type="checkbox"/> c = c // 100
L10:	<input type="checkbox"/> ver = (10 - 10*soma)	<input type="checkbox"/> ver = 10 - soma//10	<input type="checkbox"/> ver = (soma%10 - 10)	<input type="checkbox"/> ver = soma%10	<input type="checkbox"/> ver = (10 - soma%10)%10
L11:	<input type="checkbox"/> cre % 10 != ver	<input type="checkbox"/> cre % 10 == ver	<input type="checkbox"/> cre == ver % 10	<input type="checkbox"/> cre // 10 == ver	<input type="checkbox"/> cre != ver % 10
L12:	<input type="checkbox"/> cre//10**14 >= 51 and cre//10**14 <= 55	<input type="checkbox"/> cre//10**14 >= 51 or cre//10**14 <= 55	<input type="checkbox"/> cre//100 == 51 and cre//100 == 55	<input type="checkbox"/> cre//100 >= 51 and cre//100 <= 55	<input type="checkbox"/> cre//100 >= 51 or cre//100 <= 55



**Q3 [3 pontos]** Divisores próprios de um número positivo  $N$  são todos os divisores inteiros positivos de  $N$  exceto o próprio  $N$ . Por exemplo, para o número 6, seus divisores próprios são 1, 2 e 3.

Considere as seguintes definições:

**Números amigos:** dizemos que dois naturais  $a$  e  $b$  são amigos se cada um deles é igual à soma dos divisores próprios do outro. Um exemplo é o par 220 e 284, uma vez que os divisores próprios de 220 são 1, 2, 4, 5, 10, 11, 20, 22, 44, 55 e 110, que somam 284, e os divisores próprios de 284 são 1, 2, 4, 71 e 142, que somam 220.

**Números mutuamente amigos:** dois números  $a$  e  $b$  são mutuamente amigos, se a razão entre a soma de todos os divisores do número  $a$  e o número  $a$  é igual a razão entre a soma de todos os divisores do número  $b$  e o número  $b$ . Por exemplo, os números 30 e 140 são mutuamente amigos, pois:

- para 30 temos  $(1 + 2 + 3 + 5 + 6 + 10 + 15 + 30) / 30 = 72/30 = 12 / 5$  e
- para 140 temos  $(1 + 2 + 4 + 5 + 7 + 10 + 14 + 20 + 28 + 35 + 70 + 140) / 140 = 336/140 = 12 / 5$ .

**Números primos entre si:** dois inteiros  $a$  e  $b$  são primos entre si se o único divisor comum a ambos é 1.

Preencha as lacunas no código abaixo (L1 até L10), de forma a obter um programa que, dados dois inteiros positivos distintos  $a$  e  $b$  ( $a > 0$ ,  $b > 0$  e  $a \neq b$ ), calcula a soma dos seus divisores próprios (em  $sa$  e  $sb$ , respectivamente) e verifica se eles satisfazem as definições anteriores.

```

a = int(input("Digite a: "))
b = int(input("Digite b: "))
L1
while L2:
    if L3:
        L4
    L5
    L6
    i = i + 1
print(a,"gera soma dos divisores próprios",sa)
print(b,"gera soma dos divisores próprios",sb)
if L7:
    print("números amigos")
if L8:
    print("números mutuamente amigos")
mdc = a #máximo divisor comum
while L9:
    mdc = mdc - 1
if L10:
    print("primos entre si")

```

#### Exemplos de execução do programa completo:

```

Digite a: 220
Digite b: 284
220 gera soma dos divisores próprios 284
284 gera soma dos divisores próprios 220
números amigos

```

```

Digite a: 30
Digite b: 140
30 gera soma dos divisores próprios 42
140 gera soma dos divisores próprios 196
números mutuamente amigos

```

```

Digite a: 8
Digite b: 9
8 gera soma dos divisores próprios 7
9 gera soma dos divisores próprios 4
primos entre si

```

Para cada um dos 10 itens a seguir, correspondendo as lacunas no código acima, assinale a única resposta correta.

L1:	<input type="checkbox"/> sa,sb,i = 1,1,0	<input type="checkbox"/> sa,sb,i = 0,0,2	<input type="checkbox"/> sa,sb,i = 0,0,1	<input type="checkbox"/> sa,sb,i = 2,2,0
		<input type="checkbox"/> sa,sb,i = 1,1,1		
L2:	<input type="checkbox"/> i < a - b	<input type="checkbox"/> i < a and i < b	<input type="checkbox"/> i < a	<input type="checkbox"/> i < a or i < b
			<input type="checkbox"/> i < b	
L3:	<input type="checkbox"/> a % i == 0 or i < a	<input type="checkbox"/> a % i != 0 and i < a	<input type="checkbox"/> a % i != 0 or i < a	
	<input type="checkbox"/> a % i == 0 and i <= a	<input type="checkbox"/> a % i == 0 and i < a		
L4:	<input type="checkbox"/> sa = sa - 1	<input type="checkbox"/> sa += 1	<input type="checkbox"/> sa = sa + i	<input type="checkbox"/> sa = a - i
			<input type="checkbox"/> sa += a	
L5:	<input type="checkbox"/> elif b % i == 0 and i <= b:	<input type="checkbox"/> if b % i != 0 and i < b:	<input type="checkbox"/> if b % i == 0 and i < b:	
	<input type="checkbox"/> if b % i == 0 or i < b:	<input type="checkbox"/> if b % i != 0 or i < b:		
L6:	<input type="checkbox"/> sb = sb + i	<input type="checkbox"/> sb = sb - 1	<input type="checkbox"/> sb += 1	<input type="checkbox"/> sb += b
		<input type="checkbox"/> sb = b - i		
L7:	<input type="checkbox"/> sa != b or sb != a	<input type="checkbox"/> sa == sb	<input type="checkbox"/> sa == b or sb == a	<input type="checkbox"/> sa == b and sb == a
		<input type="checkbox"/> sa != b and sb != a		
L8:	<input type="checkbox"/> (sa + a)//b == (sb + b)//a	<input type="checkbox"/> (sa + b)//a == (sb + a)//b	<input type="checkbox"/> (sa + a)//a == (sb + b)//b	
	<input type="checkbox"/> (sa + a)*b == (sb + b)*a	<input type="checkbox"/> (sa + a)*a == (sb + b)*b		
L9:	<input type="checkbox"/> a%mdc == b%mdc	<input type="checkbox"/> a%mdc == 0 or b%mdc == 0	<input type="checkbox"/> not(a%mdc == 0 and b%mdc == 0)	
	<input type="checkbox"/> not(a%mdc == 0 or b%mdc == 0)	<input type="checkbox"/> a%mdc == 0 and b%mdc == 0		
L10:	<input type="checkbox"/> mdc > 1	<input type="checkbox"/> mdc >= 1	<input type="checkbox"/> mdc == 1	<input type="checkbox"/> a % mdc == b % mdc
			<input type="checkbox"/> mdc != 1	



**Q4 [3 pontos]** Preencha as lacunas no código abaixo (L1 até L11), de forma a obter um programa que lê um número inteiro  $n$  ( $n > 0$ ) e os dados de GPS de  $n$  participantes de uma corrida de rua (um número inteiro por participante) e calcula qual o participante mais rápido e sua velocidade. Os dados do GPS têm 4 dígitos, sendo os dois primeiros a distância percorrida (em metros) e os dois últimos o tempo que o participante levou para percorrer aquela distância (em segundos).

OBS: Por simplicidade, considere o cálculo da velocidade apenas com valores inteiros, descartando a parte fracionária.

```

n = int(input("Digite o número de participantes: "))
L1
L2
L3
L4
L5
L6
L7
L8
L9
L10
L11
print("O participante %d é o mais rápido (%d m/s)." % (imax, vmax))
    
```

Para cada um dos 11 itens a seguir, correspondendo as lacunas no código acima, assinale a única resposta que torna o programa acima correto.

L1:	<input type="checkbox"/> i = 1	<input type="checkbox"/> i = 0	<input type="checkbox"/> while n>0:	<input type="checkbox"/> n = 0	<input type="checkbox"/> i = n-1
L2:	<input type="checkbox"/> vmax = gps	<input type="checkbox"/> vmax = 0	<input type="checkbox"/> i = 1	<input type="checkbox"/> if n>0:	<input type="checkbox"/> while i<n:
L3:	<input type="checkbox"/> i = 1	<input type="checkbox"/> imax = gps	<input type="checkbox"/> imax = 0	<input type="checkbox"/> if n>0:	<input type="checkbox"/> while i<n:
L4:	<input type="checkbox"/> if i<n:	<input type="checkbox"/> if n>0:	<input type="checkbox"/> while i>0:	<input type="checkbox"/> while n>0:	<input type="checkbox"/> while i<n:
L5:	<input type="checkbox"/> if i<n:	<input type="checkbox"/> if vmax>v:	<input type="checkbox"/> i=i+1	<input type="checkbox"/> while i>0:	<input type="checkbox"/> vmax = vmax+1
L6:	<input type="checkbox"/> gps = int(input())	<input type="checkbox"/> gps = gps + 1	<input type="checkbox"/> print("Digite os dados para o participante %d.\n")	<input type="checkbox"/> print("Digite os dados para o participante %d.\n")	
L7:	<input type="checkbox"/> if i<n:	<input type="checkbox"/> if vm>vmax:	<input type="checkbox"/> i = i+1	<input type="checkbox"/> print("Digite os dados para o participante %d.\n")	
L8:	<input type="checkbox"/> vm = (gps%100)//(gps//100)	<input type="checkbox"/> if i<n:	<input type="checkbox"/> vm = gps%100	<input type="checkbox"/> gps = gps//100	<input type="checkbox"/> vm = (gps//100)//(gps%100)
L9:	<input type="checkbox"/> if vm>vmax:	<input type="checkbox"/> i = i+1	<input type="checkbox"/> if imax>i:	<input type="checkbox"/> while n>0:	<input type="checkbox"/> while i<n:
L10:	<input type="checkbox"/> i = i+1	<input type="checkbox"/> while vm>vmax:	<input type="checkbox"/> if vm>vmax:	<input type="checkbox"/> while i<n:	<input type="checkbox"/> vmax = vm
L11:	<input type="checkbox"/> while i<n:	<input type="checkbox"/> imax = i	<input type="checkbox"/> if imax>i:	<input type="checkbox"/> imax = imax+1	<input type="checkbox"/> i = i+1

#### Exemplo de execução do programa:

```

Digite o número de participantes: 2
Digite os dados para o participante 1.
4010
Digite os dados para o participante 2.
6030
O participante 1 é o mais rápido (4 m/s).
    
```

Rascunho