



MAC2166 - Introdução a Computação - 2019S1

Avaliação P1

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

<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6	<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9	<input type="checkbox"/> 10	<input type="checkbox"/> 11	<input type="checkbox"/> 12	<input type="checkbox"/> 13	<input type="checkbox"/> 14	<input type="checkbox"/> 20
----------------------------	----------------------------	----------------------------	----------------------------	----------------------------	----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------

← Marque as quadrículas ao lado para formar o seu número USP e escreva seu nome completo em letra legível na linha pontilhada abaixo. **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 a saída impressa do programa.

```
def main():
    n = 22
    i = 4
    a = 1
    b = 0
    while b >= 0:
        if i%2 == 0:
            i = i + 1
            n = n - 3
        else:
            i = i + 3
            n = n - 1
        b = a
        a = a + (n - 2*i)
        print((n%7 + 3) * 7)
main()
```

Rascunho

Selecione o primeiro número impresso:

<input type="checkbox"/> 28	<input type="checkbox"/> 56	<input type="checkbox"/> 35	<input type="checkbox"/> 63	<input type="checkbox"/> 49	<input type="checkbox"/> 42	<input type="checkbox"/> 21
-----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------

Selecione o segundo número impresso:

<input type="checkbox"/> 21	<input type="checkbox"/> 28	<input type="checkbox"/> 35	<input type="checkbox"/> 56	<input type="checkbox"/> 63	<input type="checkbox"/> 49	<input type="checkbox"/> 42
-----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------

Selecione o terceiro número impresso ou N/A (não aplicável) se não ocorre mais que 2 impressões:

<input type="checkbox"/> 28	<input type="checkbox"/> 42	<input type="checkbox"/> 35	<input type="checkbox"/> 21	<input type="checkbox"/> N/A	<input type="checkbox"/> 56	<input type="checkbox"/> 49	<input type="checkbox"/> 63
-----------------------------	-----------------------------	-----------------------------	-----------------------------	------------------------------	-----------------------------	-----------------------------	-----------------------------

Selecione o quarto número impresso ou N/A (não aplicável) se não ocorre mais que 3 impressões:

<input type="checkbox"/> 35	<input type="checkbox"/> 42	<input type="checkbox"/> 21	<input type="checkbox"/> 28	<input type="checkbox"/> N/A	<input type="checkbox"/> 49	<input type="checkbox"/> 56	<input type="checkbox"/> 63
-----------------------------	-----------------------------	-----------------------------	-----------------------------	------------------------------	-----------------------------	-----------------------------	-----------------------------

Selecione o quinto número impresso ou N/A (não aplicável) se não ocorre mais que 4 impressões:

<input type="checkbox"/> N/A	<input type="checkbox"/> 56	<input type="checkbox"/> 28	<input type="checkbox"/> 49	<input type="checkbox"/> 42	<input type="checkbox"/> 63	<input type="checkbox"/> 21	<input type="checkbox"/> 35
------------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------



Q2 [2 pontos] Construir um algoritmo para simular um jogo de *par ou ímpar*, supondo-se que os 2 jogadores digitam seu lance sem saber o que o outro digitou. Os jogadores devem digitar um valor entre 0 e 5 e o jogador 1 vence sempre que a soma dos dois valores for par, caso contrário vence o jogador 2. O jogo é finalizado quando um dos jogadores digitar algum inteiro negativo (esse lance NÃO será considerado como “partida válida”). Ao final deve-se imprimir quantas partidas o jogador 1 ganhou e o total de “partidas válidas”.

* Veja um exemplo:

Entradas	Saídas	DICA 1: As variáveis do programa são APENAS: ganha1 : quantidade de vitórias do jogador 1; total : total de partidas válidas disputadas; j1 : jogada do jogador 1 (entre 0 e 5); j2 : jogada do jogador 2 (entre 0 e 5);
0 1	Partida 0	DICA 2: Não tente usar todas as combinações, tente codificar o programa na área de rascunho e depois escolha os trechos adequados. O RASCUNHO NÃO SERÁ CONSIDERADO NA NOTA.
0 2	Partida 1	DICA 3: No quadro abaixo estão os trechos com seu número e indentação, como T18-2.
3 5	Partida 2	
2 -1	Partida 3 J1: 2 de 3	

<pre># T1 - 0 fim = 0; ganha1 = -1; total = -1; # T2 - 0 fim = 0; ganha1 = 0; total = 0; # T3 - 0 fim = 0; ganha1 = 1; total = 1; # T4 - 0 while (fim==0) : # T5 - 0 while (ganha1==total) : # T6 - 0 while (ganha1<total) : # T7 - 0 while (jogada<N) : # T8 - 1 print("Partida %d"% total); # T9 - 1 print("Partida %d"% ganha1);</pre>	<pre># T10 - 1 print("Partida %d"% N); # T11 - 1 j1 = input(); j2 = input(); # T12 - 1 j1 = int(input()); j2 = int(input()); # T13 - 1 if (j1<0 or j2<0) : ganha1 = ganha1 + 1; else : # T14 - 1 if (j1<0 or j2<0) : print("J1: %d de %d"% (ganha1, total)); else : # T15 - 1 if (j1<0 or j2<0) : fim = 1; else :</pre>	<pre># T16 - 2 if ((j1+j2) % 2 == 0) : ganha1 = ganha1 + 1; # T17 - 2 if ((j1+j2) / 2 == 0) : ganha1 = ganha1 + 1; # T18 - 2 if ((j1+j2) % 2 == 1) : ganha1 = ganha1 + 1; # T19 - 2 if ((j1+j2) % 2 == 1) : N = N + 1; # T20 - 2 N = N + 1; # T21 - 2 total = total + 1; # T22 - 0 print("J1: %d de %d"% (ganha1, total)); # T23 - 0 print("J1: %d de %d"% (ganha1, N));</pre>
---	--	---

Assinale a ÚNICA alternativa que contém os blocos corretos na ORDEM correta. Marcar mais de uma alternativa implica em ZERO.

- | | | |
|--|--|--|
| <input type="checkbox"/> 3, 4, 8, 11, 15, 16, 21, 22 | <input type="checkbox"/> 1, 6, 9, 12, 14, 17, 19, 23 | <input type="checkbox"/> 3, 4, 8, 12, 15, 16, 21, 22 |
| <input type="checkbox"/> 3, 5, 9, 13, 16, 17, 22, 21 | <input type="checkbox"/> 1, 6, 9, 11, 14, 17, 19, 23 | <input type="checkbox"/> 2, 5, 9, 12, 14, 17, 19, 23 |
| <input type="checkbox"/> 3, 6, 9, 11, 14, 17, 19, 23 | <input type="checkbox"/> 2, 4, 8, 12, 15, 16, 21, 22 | <input type="checkbox"/> 3, 4, 8, 12, 15, 18, 21, 22 |
| <input type="checkbox"/> 2, 5, 9, 12, 14, 17, 21, 22 | <input type="checkbox"/> 1, 5, 9, 12, 14, 17, 19, 23 | <input type="checkbox"/> 2, 5, 9, 12, 14, 17, 19, 22 |

Rascunho



Q3 [3 pontos] O CPF tem a configuração XXX.XXX.XXX-XX, na qual os primeiros nove dígitos (da esquerda para direita) são o número-base e os dois últimos são dígitos de verificação (DV), usados para prevenir erros de transmissão ou digitação. O nono dígito da base define a Região Fiscal onde foi emitido o CPF tendo a seguinte abrangência: 1 (DF-GO-MS-MT-TO), 2 (AC-AM-AP-PA-RO-RR), 3 (CE-MA-PI), 4 (AL-PB-PE-RN), 5 (BA-SE), 6 (MG), 7 (ES-RJ), 8 (SP), 9 (PR-SC) e 0 (RS). O primeiro DV corresponde ao resto da divisão por 11 do somatório dos algarismo da base, visitados da esquerda para a direita, multiplicados respectivamente pelos pesos 1, 2, 3, 4, 5, 6, 7, 8 e 9. O resto 10 é considerado 0. O segundo DV é calculado de modo análogo, porém acrescentando um dígito extra à direita na base, dado pelo primeiro DV calculado, e iniciando com o peso 0, tal como explicado no exemplo.

```

cpf = int(input("Digite o CPF: "))
L1
L2
s1 = s2 = 0
i = 9
while L3:
    L4
    L5
    L6
    L7
    i = i - 1
L8
L9
L10
L11
if L12:
    print("CPF válido da região sul do Brasil")

```

Exemplo: Para CPFs da forma 280.012.389-XX, com base 280012389 e região fiscal 9, temos:

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

2+16+ 0+ 0+ 5+12+21+64+81 = 201									
									201÷11=18, com resto 3
2	8	0	0	1	2	3	8	9	3
*	*	*	*	*	*	*	*	*	*
0	1	2	3	4	5	6	7	8	9

0+ 8+ 0+ 0+ 4+10+18+56+72+27 = 195									
									195÷11=17, com resto 8

Logo, o único CPF válido com a base fornecida é 280.012.389-38 (isto é, XX = 38). Note portanto, por exemplo, que o CPF 280.012.389-14 é inválido.

Utilizando um único laço while, preencha as lacunas no código acima (L1 até L12), de forma a obter um programa em Python que dado um número de CPF (sem '.' e '-'), fornecido como um inteiro com 11 dígitos, verifica se ele é um CPF válido pertencente a Região Sul do Brasil (PR-Paraná, SC-Santa Catarina e RS-Rio Grande do Sul).

L1:	<input type="checkbox"/> DVs = cpf%100	<input type="checkbox"/> DVs = cpf%10 + cpf%100	<input type="checkbox"/> DVs = (cpf*10)%10	<input type="checkbox"/> DVs = cpf%10	<input type="checkbox"/> DVs = cpf//100
L2:	<input type="checkbox"/> base = cpf//100	<input type="checkbox"/> base = cpf%10	<input type="checkbox"/> base = base - cpf%100	<input type="checkbox"/> base = cpf//10	<input type="checkbox"/> base = cpf%100
L3:	<input type="checkbox"/> cpf >= 0	<input type="checkbox"/> base >= 0	<input type="checkbox"/> cpf > 0	<input type="checkbox"/> i > 1	<input type="checkbox"/> base > 0
L4:	<input type="checkbox"/> r = base%10	<input type="checkbox"/> base = base%10	<input type="checkbox"/> r = base//10	<input type="checkbox"/> r = cpf%10	<input type="checkbox"/> r = cpf//10
L5:	<input type="checkbox"/> s1 = s1 + r*(9-i-1)	<input type="checkbox"/> s1 = s1 + r*(9-i)	<input type="checkbox"/> s1 = s1 + r*(i+1)	<input type="checkbox"/> s1 = s1 + r*(i-1)	<input type="checkbox"/> s1 = s1 + r*i
L6:	<input type="checkbox"/> s2 = s2 + r*(i+1)	<input type="checkbox"/> s2 = s2 + r*i	<input type="checkbox"/> s2 = s2 + r*(9-i-1)	<input type="checkbox"/> s2 = s2 + r*(9-i+1)	<input type="checkbox"/> s2 = s2 + r*(9-i)
L7:	<input type="checkbox"/> base = base-r	<input type="checkbox"/> base = base-r*10	<input type="checkbox"/> base = base%10	<input type="checkbox"/> base = base//10	<input type="checkbox"/> base = base//(i*10)
L8:	<input type="checkbox"/> DV1 = s1%11	<input type="checkbox"/> DV1 = s1	<input type="checkbox"/> DV1 = (s1%11)%10	<input type="checkbox"/> DV1 = (s1%11)//10	while DV1 > 11: DV1 = DV1 - 11
L9:	<input type="checkbox"/> s2 = s2 + DV1*9	<input type="checkbox"/> s2 = s2 + r*9	<input type="checkbox"/> s2 = s2 + DV1*r	<input type="checkbox"/> s2 = s2 + DV1*i	<input type="checkbox"/> s2 = s2 + i*9
L10:	<input type="checkbox"/> DV2 = (s2%11)//10	<input type="checkbox"/> DV2 = s2	<input type="checkbox"/> DV2 = s2%11	<input type="checkbox"/> DV2 = (s2%11)%10	while DV2 > 11: DV2 = DV2 - 11
L11:	<input type="checkbox"/> reg = cpf//(10**8)	<input type="checkbox"/> reg = base%10	<input type="checkbox"/> reg = (cpf//100)%10	<input type="checkbox"/> reg = cpf - DVs	<input type="checkbox"/> reg = base//(10**8)
L12:	<input type="checkbox"/> (reg==9 and reg==0) or DVs == DV1*10+DV2	<input type="checkbox"/> reg%9 == 0 and DVs == DV1+DV2*10	<input type="checkbox"/> (reg==9 or reg==0) and DVs == DV1*10+DV2	<input type="checkbox"/> reg==9 or (reg==0 and DVs == DV1*10+DV2)	<input type="checkbox"/> reg%9 == 0 and DVs == DV1+DV2



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 n números inteiros positivos, e verifica se cada um dos n números são compostos por somente dígitos ímpares. O programa deve imprimir se cada número é composto somente por dígitos ímpares ou não. Por exemplo, para $n = 4$ e os números 1353, 90756, 5317, 95850, o programa deve imprimir:

```
1353 eh composto somente por impares
90756 nao eh composto somente por impares
5317 eh composto somente por impares
95850 nao eh composto somente por impares
```

```
def main():
    n = int(input("Digite n:"))
    L1
    L2
    L3
    L4
    L5
    L6
    L7
    L8
    L9
    L10
    print("%d nao eh composto somente por impares"%(d_copia))
    L11
    print("%d eh composto somente por impares"%(d_copia))
main()
```

Rascunho

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"/> while n > 0:	<input type="checkbox"/> if n >= 0:	<input type="checkbox"/> if n > 0:	<input type="checkbox"/> while n >= 0:
	<input type="checkbox"/> while i < n:		
L2: <input type="checkbox"/> d = d_copia = n	<input type="checkbox"/> d = n	<input type="checkbox"/> d_copia = d = float(input("Numero:"))	<input type="checkbox"/> d_copia = d = int(input("Numero:"))
	<input type="checkbox"/> d = d_copia = float(input("Numero:"))		
L3: <input type="checkbox"/> n = n + 1	<input type="checkbox"/> d = d + 1	<input type="checkbox"/> d = d - 1	<input type="checkbox"/> n = n - 1
	<input type="checkbox"/> i = i + 1		
L4: <input type="checkbox"/> par = d // 10	<input type="checkbox"/> d = d % 10	<input type="checkbox"/> par = False	<input type="checkbox"/> d = d // 10
	<input type="checkbox"/> par = True		
L5: <input type="checkbox"/> while (par) :	<input type="checkbox"/> while (not par):	<input type="checkbox"/> while (d>0 and not par) :	<input type="checkbox"/> while (d >= 0) :
	<input type="checkbox"/> while (d_copia > 0) :		
L6: <input type="checkbox"/> dig = n	<input type="checkbox"/> dig = d%10	<input type="checkbox"/> dig = n / 10	<input type="checkbox"/> dig = dig + 1
	<input type="checkbox"/> dig = n // 10		
L7: <input type="checkbox"/> if (dig == 0) :	<input type="checkbox"/> if (dig%2 == 0) :	<input type="checkbox"/> while (dig%2 == 0) :	<input type="checkbox"/> while (dig %2 == d) :
	<input type="checkbox"/> if (dig % 10 == 0) :		
L8: <input type="checkbox"/> dig = dig % 2	<input type="checkbox"/> par = False	<input type="checkbox"/> par = True	<input type="checkbox"/> dig = dig % 10
	<input type="checkbox"/> par = dig % 2		
L9: <input type="checkbox"/> par = True	<input type="checkbox"/> par = False	<input type="checkbox"/> d = d % 10	<input type="checkbox"/> dig = dig + 1
	<input type="checkbox"/> d = d // 10		
L10: <input type="checkbox"/> if (dig) :	<input type="checkbox"/> if (dig %2 != 0) :	<input type="checkbox"/> if (d_copia == 0 or par) :	<input type="checkbox"/> if (not par) :
	<input type="checkbox"/> if (dig == 0) :		
L11: <input type="checkbox"/> if (n < 0) :	<input type="checkbox"/> elif (n < 0) :	<input type="checkbox"/> elif (n > 0) :	<input type="checkbox"/> if (n > 0) :
	<input type="checkbox"/> else :		