

2ª Chamada da 1ª Prova de Estrutura de Dados
UniAcademia – Engenharia de Software

Nome : _____
Professor : Luiz Thadeu Grizendi

Nota : _____
Data : 05/07/2022

1ª Questão - valor 2 pontos

Em Python, para obtermos o resto da divisão inteira de um número por 10, fazemos da seguinte forma $n \% 10$.

Por exemplo: $15 \% 10$ resultará 5.

Para obtermos o quociente da divisão inteira de um número por 10, fazemos da seguinte forma $n // 10$.

Por exemplo: $15 // 10$ resultará 1 e $325 // 10$ resultará 32.

Considere a função Mystery a seguir. Qual é o valor retornado se ativarmos essa função com o comando `Mystery(432)`.

```
def mystery(n):  
    if n == 0:  
        return 1  
    else:  
        d = n % 10  
        return d * mystery(n // 10)
```

2ª Questão - valor 2 pontos

Considere a função a seguir em Python.

```
def F(n):  
    if n < 4:  
        return 3 * N  
    else:  
        return 2 * F(n - 4) + 5
```

Determine o resultado de $F(3) + F(7)$

3ª Questão - valor 2 pontos

Considere as funções a seguir em Python.

```
def X(n):  
    if n == 0:  
        return True  
    elif n > 0:  
        return Y(n-1)  
  
def Y(n):  
    if n == 0:  
        return False  
    elif n > 0:  
        return X(n-1)
```

Determine $X(3) + X(4)$

2ª Chamada da 1ª Prova de Estrutura de Dados
UniAcademia – Engenharia de Software

4ª Questão - valor 2 pontos

Um número diferente de 0 (zero), significa inserir esse número na fila e, o número 0, significa remover o elemento da que está na frente da fila.

Dar a sequência de valores retornados pelas operações dequeue (remoção na fila), quando esta sequência de operações é realizada em um fila, inicialmente vazia, após a leitura da seguinte sequência de números

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

5ª Questão - valor 2 pontos

Pilhas são comumente usadas em sistemas computacionais. Por exemplo: as operações desfazer e refazer nos editores de texto e, voltar e avançar nos navegadores, são exemplos de implementações de pilhas.

Considere um simulador de editor de texto usando os dados de entrada como strings.

O usuário poderá utilizar três strings especiais que funcionarão como comandos:

‘ctrl z’ faz o simulador descartar a última string digitada
‘ctrl y’ faz o simulador recuperar a última string descartada
‘stop’ faz o simulador parar a execução.

Por exemplo:

1. String digitada ‘a’
Saída do console: ‘a => ’
2. String digitada ‘b’
Saída do console: ‘b => a => ’
3. String digitada ‘ctrl z’
Saída do console: ‘a => ’
4. String digitada ‘ctrl y’
Saída do console: ‘b => a => ’
5. String digitada ‘stop’

Qual será a saída do console se digitarmos a seguinte sequência de comandos:

‘b’, ‘a’, ‘c’, ‘ctrl z’, ‘ctrl z’, ‘ctrl y’, ‘d’, ‘ctrl z’, ‘ctrl y’, ‘stop’