|  |  |  |  |
| --- | --- | --- | --- |
| **ALUNO** | **Ramon da Silva Ferreira** | **MATRÍCULA** | **202007404823** |
| **DISCIPLINA** | **Paradigmas de Linguagens de Programação em Python** | **DATA DA PROVA** | **26/11/2020** |
| **PROFESSOR** | **Ana Carolina Costa de Oliveira** | **NOTA** |  |

**2ª AVALIAÇÃO DE PARADIGMAS DE LINGUAGENS DE PROGRAMAÇÃO EM PYTHON**

1. **Em ciência da computação, acesso aleatório é a capacidade de acessar um elemento arbitrário de uma sequência, em tempo igual, ou qualquer dado de uma população de elementos endereçáveis, de maneira tão fácil e eficiente quanto qualquer outro, não importa quantos elementos possam estar no conjunto. Diante do exposto, desenvolva um código utilizando a biblioteca Random. (2,0)**
2. **A estrutura de repetição é um recurso das linguagens de programação responsável por executar um bloco de código repetidas vezes enquanto determinada condição é atendida. Qual estrutura de repetição em Python é utilizada para lista e para cada lista uma atribuição do elemento corrente à variável definida no comando e executa o bloco de código associado a essa variável disponível? (1,5)**
3. **if**
4. **for**
5. **while**
6. **print**
7. **def**
8. **Uma aluna do grupo de pesquisa Data mining em Python necessita gerar um número randômico entre 1 e 100 na linguagem Python. Assinale a alternativa que apresenta o código correto que essa cientista de dados deve escrever. (1,5)**
9. init{

from random import randint print (randint(1, 100))}

1. from random import randint

print (randint(1, 100))

1. import random;

print (randint(1, 100));

1. import random print (random(1, 100))
2. from random import randnum print (randnum(1, 100))
3. **Python é uma linguagem interpretada largamente utilizada atualmente. Não requer tipagem de variáveis e sua sintaxe endentada favorece a organização do código. Ela é de alto nível, interpretada, orientada a objetos, funcional, de tipagem dinâmica e forte. Levando isso em conta, analise o código em Python abaixo.**

def funcao(b):

a=b\*b

return a

a=10

print(funcao(a))

**Nesse caso, ao executar o programa, o valor impresso será: (1,5)**

1. 1000.
2. 10.
3. 100.
4. 10000.
5. 1
6. **Uma função é dita recursiva quando dentro dela é feita uma ou mais chamadas a ela mesma. A ideia é dividir um problema original em subproblemas menores de mesma natureza (divisão) e depois combinar as soluções obtidas para gerar a solução do problema original de tamanho maior (conquista). Com a linguagem de programação Python, é possível realizar a criação de funções recursivas, que são bastante úteis para resolução de problemas computacionais. Tendo isso em mente, analise o código em Python abaixo.**

def f(i):

if i == 1 or i ==2:

return 1

return f(i-1) + f(i-2)

print(f(13))

**Nesse caso, ao executar o programa, o valor impresso será: (1,5)**

1. 220.
2. 223
3. 110
4. 55
5. 233
6. **Python é uma linguagem interpretada largamente utilizada atualmente. Não requer tipagem de variáveis e sua sintaxe endentada favorece a organização do código. Uma das suas funcionalidades mais poderosas são as listas. Considere o código em Python do quadro abaixo:**

a = ['UF'] + [ 'PB'] + [‘USA’]

print( len(a))

b = ['7']\*3

print( len(b))

**A saída correta correspondente às linhas 2 e 4 do código é: (1,5)**

1. 7 e 21.
2. 3 e 21.
3. 7 e 3.
4. 3 e 3.
5. 3 e 14.
6. **Desenvolva utilizando a linguagem de programação em Python, uma calculadora contendo (+ (adição), - (subtração), \* (multiplicação) e / divisão)). (2,0)**

primeiro = input("Digite o primeiro número: ")

segundo = input("Digite o segundo número: ")

operacao = input("Digite a operação: ")

resultado = None

if operacao == "+":

resultado = float(primeiro) + float(segundo)

elif operacao == "-":

resultado = float(primeiro) - float(segundo)

elif operacao == "\*":

resultado = float(primeiro) \* float(segundo)

elif operacao == "/":

resultado = float(primeiro) / float(segundo)

else:

print("Operação não existente!")

if resultado:

print("Resultado: {0}".format(resultado))

<https://repl.it/>

Respostas das questões:

Questão 1° (Em branco)

Questão 2° (Letra B = for)

Questão 3° (Letra B)

Questão 4° (Letra C)

Questão 5° (Letra E)

Questão 6° (Letra D)