

Considere a classe `CorRGB`, implementada durante as aulas, no módulo `cor_rgb_XXXXXX.py` (onde `XXXXXX` representa o seu número de aluno), e cuja especificação está disponível no moodle.

Considere também o programa Python 3 que se segue, na mesma pasta/diretoria do módulo `cor_rgb_XXXXXX.py` (de forma a que o `import` seja executado sem erros). Ignore a variável `seed` e a função `random_float`, que se destinam exclusivamente à geração de números pseudo-aleatórios.

---

```
from cor_rgb_XXXXXX import CorRGB
```

```
seed = 1922216
def random_float():
    global seed
    seed = (16807*seed) % 2147483647
    return seed / 2147483646

s1 = []
s2 = []
s3 = []
for n in range(19500):
    s1.append(CorRGB(random_float(), random_float(), random_float()))
    s2.append(CorRGB(random_float(), random_float(), random_float()))
    s3.append(random_float())
```

---

Acrescente a este programa:

- A lista `s4`. O elemento da lista `s4`, em cada índice, é um objeto `CorRGB` que resulta da soma dos objetos `CorRGB` das listas `s1` e `s2`, no mesmo índice.
- A lista `s5`. O elemento da lista `s5`, em cada índice, é um objeto `CorRGB` que resulta do produto dos objetos `CorRGB` das listas `s1` e `s2`, no mesmo índice.
- A lista `s6`. O elemento da lista `s6`, em cada índice, é um objeto `CorRGB` que resulta do produto do objeto `CorRGB` na lista `s1` pelo `float` da `s3`, no mesmo índice.

Indique se é verdadeiro ou falso.