

Avaliado(s)	Avaliador(es)
- RA: Nome:	- RA: Nome:
- RA: Nome:	- RA: Nome:

1. Avalie cada função considerando os seguintes vetores. Adicionalmente, indique em português, de forma concisa, o que realiza cada função.

```
int v1[] = {10, 2, 3, 100};
int v2[] = {7, 10, 7, 7, 80};
```

<pre>int funcao1(int a[], int n) { int i; int s = 0; for (i=0; i<n; i++) { s += a[i]; } return a[--i]; }</pre>	<p>[2 pontos]</p> <pre>printf("%d\n", funcao1(v1, 4)); printf("%d\n", funcao1(v2, 5));</pre>
<pre>int funcao2(int a[], int n) { int i; int s = 0; for (i=0; i<n; i++) { s += a[i]-a[i]; } return s; }</pre>	<p>[1 ponto]</p> <pre>printf("%d\n", funcao2(v1, 4)); printf("%d\n", funcao2(v2, 5));</pre>
<pre>int funcao3(int a[], int n) { int i, j, cont = 0; for (i=0; i<n-1; i++) { for (j=i+1; j<n; j++) { if (a[i]==a[j]) { cont++; } } } return cont; }</pre>	<p>[2 pontos]</p> <pre>printf("%d\n", funcao3(v1, 4)); printf("%d\n", funcao3(v2, 5));</pre>

2. Avalie a função considerando a seguinte matriz. Adicionalmente, indique em português, de forma concisa, o que realiza a função.

```
int M[3][4] = { {-1, -1, -1, -1}, {0, 0, 0, 10}, {10, 10, 10, 10} } ;
```

<pre>int funcao4 (int n, int m, int M[n][m]) { int i, j, cont=0; for (i=0; i<n; i++) { for (j=0; j<m; j++) { if (M[i][j]>0) cont++; if (M[i][j]<0) cont--; } } if (cont>0) return 1; else return 0; }</pre>	<p>[2 pontos]</p> <pre>printf("%d\n", funcao4(3, 4, M));</pre>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------

3. Indique se são verdadeiras ou falsas as seguintes afirmações (resposta correta +0,5, incorreta -0,5). [3 pontos]

- (a) Uma função em C pode devolver simultaneamente mais do que um valor.
- (b) Uma função em C tem que devolver sempre um inteiro.
- (c) Os parâmetros das funções podem ser do tipo `void`.
- (d) A instrução `return`, termina a execução de uma função apenas se for a última instrução da função em que se encontra.
- (e) A instrução `return`, quando executada dentro da função `main`, termina o programa.
- (f) O nome de uma função é opcional.