

Aluno(s) avaliado(s)	
Aluno(s) validador(es)	

1. O seguinte algoritmo promete ordenar, de forma crescente, um vetor de n números inteiros $v[0, \dots, n-1]$.

<pre>void S1(int v[], int n) { int i, aux, hasChanged; do { hasChanged = 0; for (i=0; i<n-1; i++) { if (v[i]>v[i+1]) { aux = v[i]; v[i] = v[i+1]; v[i+1] = aux; hasChanged = 1; } } } while(hasChanged==1); }</pre>	<ul style="list-style-type: none"> • A ordenação é realizada no mesmo vetor? • Qual modificação deve ser feita para ordenar na forma decrescente? • Este algoritmo é similar ao Bubble sort? Justifique
---	--

2. O seguinte algoritmo promete ordenar, de forma crescente, um vetor de n números inteiros $v[0, \dots, n-1]$.

<pre>void S2(int v[], int n) { int i, t, trocou = 1; int inicio = 0; int fim = n-1; while (trocou==1) { trocou = 0; for (i=inicio; i<fim; i++) { if (v[i] > v[i+1]) { t = v[i]; v[i] = v[i+1]; v[i+1] = t; trocou = 1; } } fim--; if (trocou==0) break; trocou = 0; for (i=fim-1; i>=inicio; i--) { if (v[i] > v[i+1]) { t = v[i]; v[i] = v[i+1]; v[i+1] = t; trocou = 1; } } inicio++; } }</pre>	<ul style="list-style-type: none"> • A ordenação é realizada no mesmo vetor? • Qual modificação deve ser feita para ordenar na forma decrescente? • Este algoritmo é similar ao Bubble sort? Justifique
---	--

3. Para cada algoritmo indique o número de comparações necessárias para ordenar uma sequência de n elementos.

Algoritmo	No melhor caso	No pior caso
Selection sort		
Insertion sort		
Bubble sort		
S1 sort		
S2 sort		

<pre>void SelectionSort (int v[], int n) { int i, j, iMin, aux; for (i=0; i<n-1; i++) { iMin = i; for (j=i+1; j<n; j++) { if (v[iMin]>v[j]) iMin = j; } if (iMin!=i) { aux = v[iMin]; v[iMin] = v[i]; v[i] = aux; } } }</pre>	<pre>void InsertionSort (int v[], int n) { int i, j, aux; for (i=1; i<n; i++) { aux = v[i]; for (j=i-1; j>=0 && v[j]>aux ; j--) v[j+1] = v[j]; v[j+1] = aux; } }</pre>
<pre>Void BubbleSort (int v[], int n) { int i, j, aux; for (i=n-1; i>=1; i--) { for (j=0; j<i; j++) { if (v[j]>v[j+1]) { aux = v[j]; v[j] = v[j+1]; v[j+1] = aux; } } } }</pre>	