



ep05 (25min)

I (10 val.) Considere o algoritmo `heapsort` e o vector `a` indicado abaixo. Indique o conteúdo do vector `a` após três iterações do ciclo `while` na chamada à função `heapsort(a, 0, 10)`. Na sua resposta, indique a sequência das trocas de elementos do vector na função `buildheap` e em cada uma das iterações do ciclo.

```
1 typedef int Item;
2 #define key(A) (A)
3 #define less(A, B) (key(A) < key(B))
4 #define exch(A, B) { Item t = A; A = B; B = t; }
5
6 void heapsort(Item a[], int l, int r) {
7     buildheap(a, l, r);
8     while (r-l > 0) {
9         exch(a[l], a[r]);
10        fixDown(a, l, --r, l);
11    }
12 }
```

`a = { 35, 25, 30, 15, 20, 23, 22, 31, 28, 17, 19 }`

II (10 val.) Considere a aplicação do algoritmo `RadixSort MSD`, em que cada passo os elementos do vector `a` são ordenados considerando um bit. O algoritmo deve processar apenas os 5 dígitos menos significativos de cada número.

`a = { 17, 7, 3, 1, 4, 8, 13, 10, 5, 19 }`

Indique o conteúdo do vector após as três primeiras iterações do algoritmo, ilustrando o conteúdo do vector em cada iteração.

Sugestão: Comece por representar os números do vector `a` em binário.