

Implement some algorithms to sort lists.

1. Define a function `insert :: [Int] → Int → [Int]` that, given a sorted list and an element, correctly inserts the new element in the list.

Define a function `isort :: [Int] → [Int]` that implements insertion sort using the previous function.

2. Define a function `remove :: [Int] → Int → [Int]` that, given a list and an element  $x$ , erases the first occurrence of  $x$  from the list. You can assume that the element is always in the list.

Define a function `ssort :: [Int] → [Int]` that implements selection sort using the previous function.

3. Define a function `merge :: [Int] → [Int] → [Int]` that, given two sorted lists, merges them to get a list with all the elements in sorted order.

Define a function `msort :: [Int] → [Int]` that implements merge sort using the previous function.

4. Define a function `qsort :: [Int] → [Int]` that implements quick sort.

5. Generalize the previous function into `genQsort :: Ord a ⇒ [a] → [a]` that sorts elements of any type.

### Scoring

Each sorting algorithm scores 20 points.

#### Sample input

```
insert [10,20,30,40] 25
insert [10,20,30,40] 20
isort [6,5,2,5,6,8]
remove [6,4,3,5,2,3] 2
remove [6,4,3,5,2,3] 6
ssort [6,5,2,5,6,8]
merge [1,2,5,7,8] [2,4,7,9]
msort [6,5,2,5,6,8]
qsort [6,5,2,5,6,8]
genQsort [5.0,3.0,2.5]
genQsort ["jordi", "albert", "josep"]
genQsort "antaviana"
```

#### Sample output

```
[10,20,25,30,40]
[10,20,20,30,40]
[2,5,5,6,6,8]
[6,4,3,5,3]
[4,3,5,2,3]
[2,5,5,6,6,8]
[1,2,2,4,5,7,7,8,9]
[2,5,5,6,6,8]
[2,5,5,6,6,8]
[2.5,3.0,5.0]
["albert","jordi","josep"]
"aaaainntv"
```

### Problem information

Author : Albert Rubio / Jordi Petit

Translator : Jordi Petit

Generation : 2024-04-30 18:37:05

© Jutge.org, 2006–2024.

<https://jutge.org>