本日の課題

いくつかの数が並んだリストが与えらえられたとき，このリストをquicksortアルゴリズムでソートし，このソートに要した数の比較回数を返す関数 `kadai(lst, keyfunc_num)` と作れ． ただし，`lst`はソートするリストであり，`keyfunc_num`はキーの選択方法を切り替えるための整数であるとする．ソートされたリストを返す必要はない．また，keyfunc_numの値とkeyの取り方のアルゴリズムは以下の表の通り．

`keyfunc_num`	挙動
0	配列の先頭の値をkeyにする
1	配列の中央の値をkeyにする
2	ランダムな位置をkeyにする

かならず幾つかの例について動作を確認してからプログラムを提出すること.

解答例 (1)

基本的にはクイックソートアルゴリズムを動かして，比較した回数を数えれば良い．比較関数を配列にいれて，番号によって選択する．

```
function kadai(lst, keyfunc_num){
    var count = 0
    function qsortx(lst, keyfunc){
        if (lst.length <= 1) return lst;
        else {
            var key = keyfunc(lst);
            var lt = [], eq = [], gt = [], i;
            for (i = 0; i < lst.length; i++){
                var x = lst[i];
                if (x < key){
                    lt.push(x);
                    count += 1;
                } else if (x > key){
                    gt.push(x);
                    count += 1;
                } else eq.push(x);
            } /* for */
            return qsortx(lt, keyfunc).concat(eq).
                concat(qsortx(gt, keyfunc));
        } /* if */
    }
```

```
    qsortx(
        lst,
        [function(lst){
            return lst[0];
         },
         function(lst){
            return lst[Math.floor(lst.length / 2)];
         },
         function(lst){
            var n = lst.length;
            return lst[Math.floor(n * Math.random())];
        }][keyfunc_num]);
    return count;
}
```

解答例 (2)

実際に動かしてみる．関数testによって，実行回数および実行時間を測る．

```
function test(){
    var lstx = [];
    for (var i = 0; i < 6000; i++)
        lstx.push(Math.random());
    // lstx.sort(function(x, y){return x - y;})
    for (var num = 0; num < 3; num++){
        var t1, t2;
        t1 = new Date();
        var res = kadai(lstx, num);
        t2 = new Date();
        puts(num + ": " + "count = " + res +
             " : " + (t2 - t1) + "ms" );
    }
}

test();
```

```
0: count = 93301 : 12ms
1: count = 86573 : 13ms
2: count = 88549 : 10ms
```

```
0: count = 17997000 : 1642ms
1: count = 63822 : 8ms
2: count = 83697 : 8ms
```

左のプログラムの青い行のコメントを外したときの結果