```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2];
  uint a = -1;
  uint b = t.len;
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

.len	15	_	3	13	5	0	7	11	4	9	14	2	10	1	12	8	6
.val	\ \										8						

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2];
  uint a = -1;
  uint b = t.len;
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
len 15 3 13 5 0 7 11 4 9 14 2 10 1 12 8 6
val V 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1;
  uint b = t.len;
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

.len	15	,	3	13	5	0	7	11	4	9	14	2	10	1	12	8	6
.val	~		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1;
  uint b = t.len;
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

.len	15	,	3	13	5	0	7	11	4	9	14	2	10	1	12	8	6
.val	~		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* -1 */
  uint b = t.len;
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

.len	15	 3	13	5	0	7	11	4	9	14	2	10	1	12	8	6
.val	1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* -1 */
  uint b = t.len;
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

.len	15	 3	13	5	0	7	11	4	9	14	2	10	1	12	8	6
.val	~		1													

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* -1 */
  uint b = t.len; /* 15 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
    Jen
    15
    3
    13
    5
    0
    7
    11
    4
    9
    14
    2
    10
    1
    12
    8
    6

    .val
    0
    1
    2
    3
    4
    5
    6
    7
    8
    9
    10
    11
    12
    13
    14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* -1 */
  uint b = t.len; /* 15 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
    Jen
    15
    3
    13
    5
    0
    7
    11
    4
    9
    14
    2
    10
    1
    12
    8
    6

    .val
    0
    1
    2
    3
    4
    5
    6
    7
    8
    9
    10
    11
    12
    13
    14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 0 */
  uint b = t.len; /* 15 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 0 */
  uint b = t.len; /* 15 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 1 */
  uint b = t.len; /* 15 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 1 */
  uint b = t.len; /* 15 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
    .len
    15

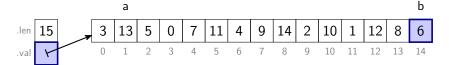
    .val

    3
    13
    5
    0
    7
    11
    4
    9
    14
    2
    10
    1
    12
    8
    6

    .val
    0
    1
    2
    3
    4
    5
    6
    7
    8
    9
    10
    11
    12
    13
    14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 1 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

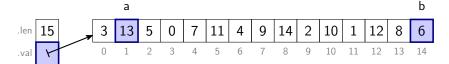
```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 1 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```



```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 1 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
h
                    a
.len | 15
                   13
                        5
                                     11
                                                  14
                                                       2
                                                           10
                                                                    12
                             0 |
                             3
                                      5
                                          6
                                                           10
                                                                11
                                                                    12
.val
```

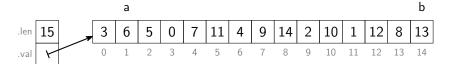
```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 1 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
    swap(t, a, b);
```



```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 1 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
    swap(t, a, b);
```

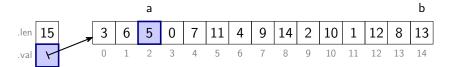
```
h
                    a
.len | 15
                                     11
                                                   14
                                                        2
                                                            10
                                                                     12
                             0
                             3
                                      5
                                           6
                                                            10
                                                                11
                                                                     12
.val
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 1 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```



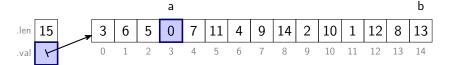
```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 2 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 2 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```



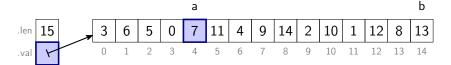
```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 3 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 3 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```



```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 4 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 4 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```



```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 5 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 5 */
  uint b = t.len; /* 14 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
a b

len 15 3 6 5 0 7 11 4 9 14 2 10 1 12 8 13

.val 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 5 */
  uint b = t.len; /* 13 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

							а								b	
.len	15	 3	6	5	0	7	11	4	9	14	2	10	1	12	8	13
.val	~	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

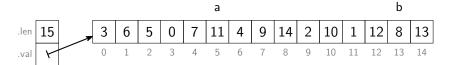
```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 5 */
  uint b = t.len; /* 13 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
a b

len 15 3 6 5 0 7 11 4 9 14 2 10 1 12 8 13

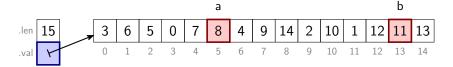
.val 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 5 */
  uint b = t.len; /* 13 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

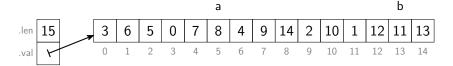


```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 5 */
  uint b = t.len; /* 13 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
    swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 5 */
  uint b = t.len; /* 13 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
    swap(t, a, b);
```

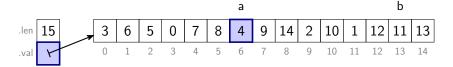


```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 5 */
  uint b = t.len; /* 13 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

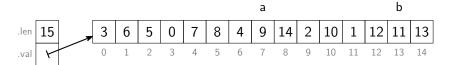


```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 6 */
  uint b = t.len; /* 13 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 6 */
  uint b = t.len; /* 13 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```



```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 7 */
  uint b = t.len; /* 13 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```



```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 7 */
  uint b = t.len; /* 13 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

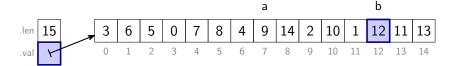
```
a b

len 15 3 6 5 0 7 8 4 9 14 2 10 1 12 11 13

.val 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 7 */
  uint b = t.len; /* 12 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 7 */
  uint b = t.len; /* 12 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```



```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 7 */
  uint b = t.len; /* 11 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

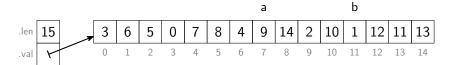
```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 7 */
  uint b = t.len; /* 11 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
a b

len 15 3 6 5 0 7 8 4 9 14 2 10 1 12 11 13

.val 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 7 */
  uint b = t.len; /* 11 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```



```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 7 */
  uint b = t.len; /* 11 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
    swap(t, a, b);
```

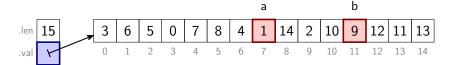
```
a b

len 15 3 6 5 0 7 8 4 9 14 2 10 1 12 11 13

.val 

o 1 2 3 4 5 6 7 8 9 10 11 12 13 14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 7 */
  uint b = t.len; /* 11 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
    swap(t, a, b);
```



```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 7 */
  uint b = t.len; /* 11 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```



```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 8 */
  uint b = t.len; /* 11 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 8 */
  uint b = t.len; /* 11 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
a b

len 15 3 6 5 0 7 8 4 1 14 2 10 9 12 11 13

.val 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 8 */
  uint b = t.len; /* 10 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 8 */
  uint b = t.len; /* 10 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
a b

len 15 3 6 5 0 7 8 4 1 14 2 10 9 12 11 13

.val 

o 1 2 3 4 5 6 7 8 9 10 11 12 13 14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 8 */
  uint b = t.len; /* 9 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
    .len
    15

    .val

    3
    6
    5
    0
    7
    8
    4
    1
    14
    2
    10
    9
    12
    11
    13

    .val
    0
    1
    2
    3
    4
    5
    6
    7
    8
    9
    10
    11
    12
    13
    14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 8 */
  uint b = t.len; /* 9 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
.len 15 3 6 5 0 7 8 4 1 14 2 10 9 12 11 13 .val
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 8 */
  uint b = t.len; /* 9 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

										-	~					
.len	15	3	6	5	0	7	8	4	1	14	2	10	9	12	11	13
.val	Y	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

h

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 8 */
  uint b = t.len; /* 9 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
    swap(t, a, b);
```

```
    Jen
    15

    3
    6
    5
    0
    7
    8
    4
    1
    14
    2
    10
    9
    12
    11
    13

    .val
    0
    1
    2
    3
    4
    5
    6
    7
    8
    9
    10
    11
    12
    13
    14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 8 */
  uint b = t.len; /* 9 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
    swap(t, a, b);
```

```
    .len
    15

    .val

    3
    6
    5
    0
    7
    8
    4
    1
    2
    14
    10
    9
    12
    11
    13

    .val
    0
    1
    2
    3
    4
    5
    6
    7
    8
    9
    10
    11
    12
    13
    14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 8 */
  uint b = t.len; /* 9 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
    .len
    15

    .val

    3
    6
    5
    0
    7
    8
    4
    1
    2
    14
    10
    9
    12
    11
    13

    .val
    0
    1
    2
    3
    4
    5
    6
    7
    8
    9
    10
    11
    12
    13
    14
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 9 */
  uint b = t.len; /* 9 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

a,b

len 15
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 9 */
  uint b = t.len; /* 9 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
.len 15 3 6 5 0 7 8 4 1 2 14 10 9 12 11 13 .val
```

a,b

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 9 */
  uint b = t.len; /* 8 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
| Second Second
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 9 */
  uint b = t.len; /* 8 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
    Jen
    15
    3
    6
    5
    0
    7
    8
    4
    1
    2
    14
    10
    9
    12
    11
    13

    .val
    0
    1
    2
    3
    4
    5
    6
    7
    8
    9
    10
    11
    12
    13
    14
```

b a

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 9 */
  uint b = t.len; /* 8 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 9 */
  uint b = t.len; /* 8 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
| Second Second
```

```
uint partition (Tab t) {
  int pivot = t.val[(t.len-1)/2]; /* 9 */
  uint a = -1; /* 9 */
  uint b = t.len; /* 8 */
  while (1) {
    do { a++; } while (t.val[a] < pivot);</pre>
    do { b--; } while (t.val[b] > pivot);
    if (a >= b) { return b; }
     swap(t, a, b);
```

```
| Second Second
```