

アルゴリズム基礎

2

与えられた整数 $a_i (i = 1, 2, \dots, n)$ を整列したい．以下の問 (i)–(ii) に答えよ．なお，回数の解析はいずれも最悪の場合でよい．また，空欄のサイズは答えの長さとは関係ない．

(i) 下記の擬似コードに対し再帰呼び出しを用いて空欄 (a) に入るアルゴリズムを示せ．また，Quicksort(1, n) により配列が整列されるまでの再帰呼び出しの回数を解析せよ．

```
Quicksort(left, right)
   $i := \text{left}, j := \text{right}$ 
   $\text{pivot} := (a_i + a_j)/2$ 
  WHILE ( $i < j$ )
    WHILE ( $a_i < \text{pivot}$ )  $i := i + 1$  END WHILE
    WHILE ( $a_j > \text{pivot}$ )  $j := j - 1$  END WHILE
    IF ( $i \leq j$ )
       $\text{tmp} := a_i, a_i := a_j, a_j := \text{tmp}$ 
       $i := i + 1, j := j - 1$ 
    END IF
  END WHILE
  (a)
END
```

(ii) 以下の三つの操作をサポートするデータ構造があるとする．これを用いて再帰呼び出しを使わず (i) のクイックソートを実現せよ (空欄 (b) に入るアルゴリズムを示せ)．また，配列が整列されるまで PUSH と POP の行われる回数を解析せよ．

操作 INITIALIZE: 要素数 0 の空のデータ構造を作る．

操作 PUSH(x): x を先頭要素として挿入する．ただし x は正整数と仮定する．

操作 POP: 先頭の要素を削除しその値を返す．ただし要素数が 0 の場合は 0 を返す．

```
INITIALIZE, PUSH( $n$ ), PUSH(1)
loop:
   $\text{left} := \text{POP}, \text{right} := \text{POP}$ 
  IF ( $\text{left} < 1$ ) GOTO done END IF
   $i := \text{left}, j := \text{right}$ 
  (b)
done:
  (配列  $a$  は整列済み)
```