

次の問 8 は必須問題です。必ず解答してください。

問 8 次のプログラムの説明及びプログラムを読んで、設問 1～3 に答えよ。

与えられた n 個のデータの中から k 番目に小さい値を選択する方法として、クイックソートを応用したアルゴリズムを考える。クイックソートとは、 n 個のデータがある基準値以下の値のグループと基準値以上の値のグループに分割し（基準値はどちらのグループに入れても構わない）、更にそれぞれのグループで基準値を選んで二つのグループに分割するという処理を繰り返してデータを整列するアルゴリズムである。クイックソートを応用して k 番目に小さい値を選択するアルゴリズムでは、データを二つのグループに分割した時点で、求める値はどちらのグループに含まれるかが確定するので、そのグループだけに、更に分割する処理を繰り返し適用する。グループの分割ができなくなった時点で、 k 番目に小さい値が選択されている。

〔プログラムの説明〕

n 個の数値が格納されている配列 x と値 k を与えて、 k 番目に小さい値を返す関数 `Select` である。ここで、配列 x の要素番号は 1 から始まる。また、配列 x の大きさは、配列に格納される数値の個数分だけ確保されているものとする。`Select` の処理の流れを次に示す。

(1) 行番号 3～4

k 番目に小さい値を選択するために走査する範囲（以下、走査範囲という）の左端を `Top`、右端を `Last` とし、まず配列全体を走査範囲とする。

(2) 行番号 5～32

- ① 走査範囲に含まれる要素の数が 1 以下になるまで、②、③ を繰り返す。
- ② 基準値 `Pivot` を選び、走査範囲内の値で基準値以下のものを左に、基準値以上のものを右に集める（行番号 6～24）。
- ③ 走査範囲が基準値以下の値から成るグループと基準値以上の値から成るグループに分割されるので、 k 番目に小さい値が含まれるグループを新たな走査範囲とする（行番号 25～30）。

- ④ 繰返しが終了したときに，要素 $x[k]$ の値が k 番目に小さい値として，選択される。

Select の引数と返却値の仕様は次のとおりである。

[関数 Select の引数／返却値の仕様]

引数名／返却値	データ型	入力／出力	意味
$x[]$	整数型	入力	数値が格納されている一次元配列
n	整数型	入力	数値の個数
k	整数型	入力	選択する数値の小ささの順位を示す値
返却値	整数型	出力	選択された数値

[プログラム]

(行番号)

```

1 ○整数型: Select(整数型: x[], 整数型: n, 整数型: k)
2 ○整数型: Top, Last, Pivot, i, j, work

3 • Top ← 1                               /* 走査範囲の左端の初期値を設定 */
4 • Last ← n                               /* 走査範囲の右端の初期値を設定 */
5 ■ Top < Last
6   • Pivot ← x[k] }
7   • i ← Top      } ← α
8   • j ← Last    }
9   ■ true        /* ループ */
10  ■ x[i] < Pivot ← β
11  |   • i ← i + 1
12  ■
13  ■ Pivot < x[j]
14  |   • j ← j - 1
15  ■
16  ▲ i ≥ j
17  |   • break      /* ループから抜ける */
18  ▼
19  • work ← x[i] }
20  • x[i] ← x[j] }
21  • x[j] ← work } ← γ
22  • i ← i + 1
23  • j ← j - 1
24  ■
25  ▲ i ≤ k
26  |   • Top ← j + 1
27  ▼
28  ▲ k ≤ j
29  |   • Last ← i - 1
30  ▼
31  ■
32 • return x[k]

```

設問 1 関数 Select の追跡に関する次の記述中の に入れる正しい答えを、解答群の中から選べ。

関数 Select の引数で与えられた配列 x の要素番号 1 ~ 7 の内容が 3, 5, 6, 4, 7, 2, 1 であり, n が 7, k が 3 のとき, 配列 x の走査範囲の左端 Top と右端 Last の値は次のとおりに変化する。

- ・ Top と Last の初期値は, それぞれ 1 と 7 である。
- ・ Top < Last が成り立つ間, 次に示す (1) 選択処理 1 回目の ① ~ ③, (2) 選択処理 2 回目の ① ~ ③, … と実行する。

(1) 選択処理 1 回目

- ① 配列 x の走査範囲を二つの部分に分ける基準値 Pivot に配列 x の 3 番目の要素 $x[3]$ の値 6 を設定する。次に, i に Top の値 1, j に Last の値 7 を設定する。
- ② 配列 x の Top から Last までの走査範囲内にある数値を, 6 以下の数値のグループと 6 以上の数値のグループの二つに分ける処理を行う。その結果, 配列 x の内容は次のとおりになる。

3, 5, 1, 4, 2, 7, 6

- ③ a を設定して選択処理の 2 回目に進む。

(2) 選択処理 2 回目

- ① 基準値 Pivot に $x[3]$ の値 1 を設定する。
- ② 配列 x の Top から Last までの走査範囲内にある数値を, 1 以下の数値のグループと 1 以上の数値のグループの二つに分ける処理を行う。その結果, 配列 x の内容は次のとおりになる。

1, 5, 3, 4, 2, 7, 6

- ③ b を設定して選択処理の 3 回目に進む。

(3) 選択処理 3 回目

⋮

この選択処理を繰り返して、 $Top < Last$ でなくなったときに処理を終了する。
このとき、関数の返却値 $x[k]$ には与えられた数値の中から k 番目に小さい値が
選択されている。

a, b に関する解答群

- | | |
|-----------------------|-----------------------|
| ア Top に値 1, Last に値 5 | イ Top に値 1, Last に値 6 |
| ウ Top に値 2, Last に値 5 | エ Top に値 2, Last に値 6 |
| オ Top に値 3, Last に値 5 | カ Top に値 3, Last に値 6 |

設問 2 次の記述中の に入れる正しい答えを、解答群の中から選べ。

引数で与えられた配列 x の要素番号 1～7 の内容が 1, 3, 2, 4, 2, 2, 2 であり、 n が 7, k が 3 のとき、選択処理が終了するまでにプログラム中の α の部分は c 回実行され、 γ の部分は d 回実行される。

c, d に関する解答群

- | | | | | | |
|-----|-----|-----|-----|-----|-----|
| ア 1 | イ 2 | ウ 3 | エ 4 | オ 5 | カ 6 |
|-----|-----|-----|-----|-----|-----|

設問 3 次の記述中の に入れる正しい答えを、解答群の中から選べ。

プログラム中の β の行 $x[i] < Pivot$ を誤って $x[i] \leq Pivot$ とした。この場合、引数で与えられた配列 x の要素番号 1～6 の内容が 1, 1, 1, 1, 1, 1 であり、 n が 6, k が 3 のとき、 e 。また、引数で与えられた配列 x の要素番号 1～6 の内容が 1, 3, 2, 4, 2, 2 であり、 n が 6, k が 3 のとき、 f 。

e, f に関する解答群

- | | |
|--------------------|---------------------|
| ア Last に値 0 が設定される | イ Pivot に値 0 が設定される |
| ウ Top に値 0 が設定される | エ 処理が終了しない |
| オ 配列の範囲を越えて参照する | |