

ゲノムリテラシー講座 講義資料2

加藤 毅

<http://www.net-machine.net/~kato/>

平成 21 年 7 月 25 日

ソート 算法と計算量

ソートとは，入力された数値列を大きさの順に並べ替える問題をさす．ソートの問題は，平成 19 年度にも平成 20 年度にも複数出題されている．その中で，3 つの算法

- 選択ソート，
- バブルソート，
- クイックソート

が取り扱われた．

選択ソート

選択ソートの算法は以下で与えられる：

```
1: function selection_sort (A[0...N - 1])
2: begin
3: for i = 0 to N - 2 do
4:   i_min := i;
5:   for j = i + 1 to N - 1 do
6:     if A[j] < A[i_min] then
7:       i_min := j;
8:     end if
9:   end for
10:  if i ≠ i_min then
11:    swap(A[i], A[i_min]);
12:  end if
13: end for
14: end.
```

(http://en.wikipedia.org/wiki/Selection_sortを参考に作成)

例 1 選択ソートの実行例を以下に示す：

35241	最小値を見つける
13524	交換
13524	最小値を見つける
12354	交換
12354	最小値を見つける
12345	交換
12345	確定！

バブルソート

バブルソートの算法は以下で与えられる：

```

1: function bubble_sort( $A[0 \dots N - 1]$ )
2: begin
3:  $n := N$ ;
4: repeat
5:   swapped := false;
6:    $n := n - 1$ ;
7:   for  $i := 0$  to  $n-1$  do
8:     if  $A[i] > A[i + 1]$  then
9:       swap( $A[i], A[i + 1]$ );
10:      swapped := true;
11:     end if
12:   end for
13: until not swapped;
14: end.

```

(http://en.wikipedia.org/wiki/Bubble_sortを参考に作成)

例 2 バブルソートの実行例を以下に示す：

53471
35471
34571
34571
34517
34517
34517
34157
34157
31457
13457

クイックソート

クイックソートの算法を以下に示す：

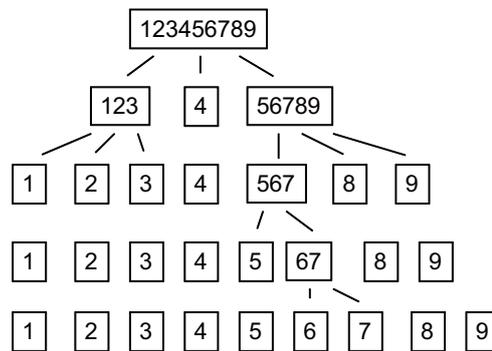
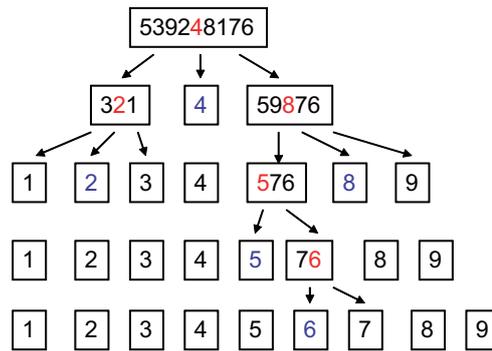
```

1: function quicksort ( $A$ : リスト)
2: begin
3:  $B, C$  は空のリストとする;
4: if  $A$  の長さ  $\leq 1$  then
5:   return  $A$ ;
6: end if
7: ピボットなる要素  $m$  を選び,  $A$  から  $m$  を抜く;
8: for all  $x$  in  $A$  do
9:   if  $x \leq m$  then
10:    リスト  $B$  に  $x$  を追加する;
11:   else
12:    リスト  $C$  に  $x$  を追加する;
13:   end if
14: end for
15:  $B := \text{quicksort}(B)$ ;
16:  $C := \text{quicksort}(C)$ ;
17: return  $B, m, C$  の順番につなげたリスト;
18: end.

```

(<http://en.wikipedia.org/wiki/Quicksort>を参考に作成)

例 3 クイックソートの実行例を以下に示す：



計算量

それぞれの算法の計算量は次のように与えられる。

選択ソート	$O(N^2)$
バブルソート	$O(N^2)$
クイックソート	平均計算量 $O(N \log N)$ 最悪計算量 $\Omega(N^2)$

***平成 19 年度 問 27**

コンピュータ上では，数値は二進数で扱われる．十進数の 50 を二進数で表現すると「00110010」となる．ここで，十進数の 200 は，二進数ではどのように表現されるか．選択肢の中から適するものを一つ選べ．

- 1 01100100
- 2 01100110
- 3 11001000
- 4 11001100

解説

一般に， n 桁の 2 進数で「 $x_{n-1}x_{n-2}\dots x_2x_1x_0$ 」と表記される数値は

$$x_{n-1}2^{n-1} + x_{n-2}2^{n-2} + \dots + x_22^2 + x_12^1 + x_02^0$$

を意味する．設問にあるように，値 50 は二進数で「00110010」と表される．
確かに，

$$0 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 50$$

を満たしている．

では，200 は 50 の 4 倍なので，

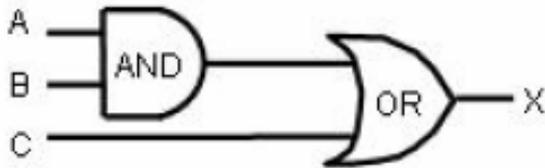
$$\begin{aligned} 200 &= 4 \cdot 50 = 2^2 \cdot 50 \\ &= 2^2 (1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0) \\ &= 1 \cdot 2^{5+2} + 1 \cdot 2^{4+2} + 0 \cdot 2^{3+2} + 0 \cdot 2^{2+2} + 1 \cdot 2^{1+2} + 0 \cdot 2^{0+2} + 0 \cdot 2^1 + 0 \cdot 2^0 \end{aligned}$$

なので，二進数では「11001000」と表される．

よって，選択肢 3 が正しい．

*平成 19 年度 問 28

下記の図(ア)は、いくつかの論理素子を接続した論理回路を表現している。この回路は、A, B, C は入力であり、X が出力である。この回路に対する入出力の結果によって真理値表(イ)を作成した。ここで、真理値表の(a), (b)に入る値の組み合わせとして適切なものを選択肢の中から一つ選べ。



A	B	C	X
0	0	0	0
0	0	1	(a)
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	(b)
1	1	1	1

図(ア) 論理回路図

図(イ) 真理値表

- 1 (a) = 0, (b) = 0,
- 2 (a) = 0, (b) = 1,
- 3 (a) = 1, (b) = 0,
- 4 (a) = 1, (b) = 1,

解説

論理演算 and, or, not は次のように値を返す：

X	Y	X and Y	X or Y
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

X	not X
0	1
1	0

論理回路図より、論理式

$$X = ((A \text{ and } B) \text{ or } C)$$

を得る。

よって、(a) の値を求めるには、 $A = 0$ と $B = 0$ と $C = 1$ を代入すると、

$$\begin{aligned} (a) &= ((0 \text{ and } 0) \text{ or } 1) \\ &= (0 \text{ or } 1) \\ &= 1 \end{aligned}$$

を得る。

(b) の値を求めるには、 $A = 1$ と $B = 1$ と $C = 0$ を代入すると、

$$\begin{aligned} (b) &= ((1 \text{ and } 1) \text{ or } 0) \\ &= (1 \text{ or } 0) \\ &= 1 \end{aligned}$$

を得る。

よって、選択肢 4 が正しい。

***平成 19 年度 問 26**

スタックは、コンピュータ上でよく用いられるデータ構造の一つで、最後に入ったデータが最初に出る、後入れ先出し (Last In First Out: LIFO) が特徴である。ここで、十分な大きさのスタックがあるとし、PUSH 命令はスタックにデータを格納する操作、POP 命令は、データを取り出す操作とする。下記のように 9 回の命令を実行したとき、最後の POP 命令によって出力されるデータは何か。適切なものを選択肢の中から一つ選べ。

- (1) PUSH a
 - (2) PUSH b
 - (3) PUSH c
 - (4) PUSH b
 - (5) POP
 - (6) POP
 - (7) PUSH b
 - (8) POP
 - (9) POP
- 1 a
 - 2 b
 - 3 c
 - 4 空なので何も得られない。

解説

スタックの中身は

- (1) PUSH a
a
- (2) PUSH b
b a
- (3) PUSH c
c b a
- (4) PUSH b
b c b a
- (5) POP
b c b a
- (6) POP
c b a
- (7) PUSH b
b b a
- (8) POP
b b a
- (9) POP
b a

選択肢 2 が正しい。

***平成 20 年度 問 27**

データ格納方法としてキューを用いる．アルファベットの文字データをキューに出し入れする．データの操作を

A enqueue
 B enqueue
 C enqueue
 dequeue
 dequeue
 D enqueue
 A enqueue
 dequeue
 dequeue

のように行ったとき，最後の dequeue で取り出されるデータとして正しいものを選択肢の中から一つ選べ．ただし，enqueue はデータを追加する操作，dequeue はデータを取り出す操作である．

- 1 A
- 2 B
- 3 C
- 4 D

解説

キューの中身は

A enqueue
A
 B enqueue
BA
 C enqueue
CBA
 dequeue
CBA
 dequeue
CB
 D enqueue
DC
 A enqueue
ADC
 dequeue
ADC
 dequeue
AD

のようになる．

よって，最後の dequeue で取り出されるのは D となる．
 よって，選択肢 4 が正しい．

*平成 19 年度 問 31

一定の順序に並べられたデータの列から目的の値を見つけ出す探索法の一つに二分探索法がある。二分探索法によって下記に示したデータから「4」を見つけ出す例を示す。

データの列：1, 3, 4, 6, 7, 9, 10

- (1) 並べられたデータの中央の値「6」と目的の値「4」を比較する。
「4」のほうが小さいため、範囲を 1, 3, 4 の 3 個に絞る。
- (2) 中央の値「3」と目的の値「4」を比較する。
「4」のほうが大きいため、範囲を 4 に絞る。
- (3) 目的の値「4」と等しいため、目的の値が見つかる。

このように、二分探索法では、探索の範囲を半分に絞り込む操作を繰り返すことによって探索を行う方法である。ここで、対象のデータが 1,000 個与えられたとき、そこに含まれているデータを見つけ出すのに最大何回の二分操作が必要であるか。もっとも適した回数を選択肢の中から選べ。

- 1 8 回
- 2 10 回
- 3 12 回
- 4 14 回

解説

二分探索法は次のように表される算法である：

```

1: function binary_search (A[0...n-1], i_min, i_max, x)
2: begin
3:   if i_min > i_max then
4:     return -1;
5:   end if
6:   i_mid := i_min + ⌊(i_max - i_min) / 2⌋;
7:   if A[i_mid] > x then
8:     return binary_search(A, i_min, i_mid - 1, x);
9:   else if A[i_mid] < x then
10:    return binary_search(A, i_mid, i_max, x);
11:   else
12:    return i_mid;
13:   end if
14: end.

```

では、1,000 個のデータから、最大何回の二分操作が必要か調べてみる。

- (1) 1,000 個を 500 個 500 個に分ける。
- (2) 500 個を 250 個 250 個に分ける。
- (3) 250 個を 125 個 125 個に分ける。
- (4) 125 個を 62 個 63 個に分ける。
キー値が中央の値より大きかったとする。
- (5) 63 個を 31 個 32 個に分ける。
キー値が中央の値より大きかったとする。
- (6) 32 個を 16 個 16 個に分ける。
- (7) 16 個を 8 個 8 個に分ける。
- (8) 8 個を 4 個 4 個に分ける。
- (9) 4 個を 2 個 2 個に分ける。
- (10) 2 個を 1 個 1 個に分ける。

よって、最大 10 回二分操作を行う必要がある。

***平成 19 年度 問 32**

与えられたデータの列を一定の順序に並べ替えるアルゴリズムにバブルソートがある。バブルソートを用いてデータを昇順に並べ替える手順の一部を次に示す。

データの列：1, 3, 4, 6, 7, 9, 10

- (1) 一番左の要素「3」とその右隣の要素「2」を比較し、右のほうが小さければお互いを交換する。

交換前	3	2	1
交換後	2	3	1

- (2) 操作を一つ右にずらし「3」について、同様な右隣の要素「1」を比較し、右のほうが小さければお互いに交換する。

交換前	2	3	1
交換後	2	1	3

この操作によって、もっとも大きな値「3」が一番右に移動することがわかる。また、この操作を左から交換操作が起こらなくなるまで繰り返すことによって、全ての並びを昇順にすることができる。ここで「5, 3, 4, 7, 1」のデータが与えられたとき、このデータをバブルソートによって昇順に並べ替えるには、何回の交換操作が必要となるか。もっとも適した値を選択肢の中から一つ選べ。

- 1 4 回
- 2 6 回
- 3 8 回
- 4 10 回

解説

交換回数を数える：

交換回数	
0 回	53471
1 回	35471
2 回	34571
2 回	34571
3 回	34517
3 回	34517
3 回	34517
4 回	34157
4 回	34157
5 回	31457
6 回	13457

よって、交換回数は 6 回となり、選択肢 2 が正しい。

*平成 19 年度 問 33

与えられたデータの列を一定の順序に並べ替えるソートアルゴリズムの一つに選択法がある。選択法によってデータを昇順にソートするには、「与えられたデータの中から最小値を選び出し取り出す。次に、それを除いたデータの中から最小値を取り出す。」という操作をデータがなくなるまで繰り返す。この選択法に必要な計算の回数に関する説明として、もっとも適切なものを選択肢の中から一つ選べ。ここで、データ数を N とする。

- 1 データの内容には依存せず、およそ N の 2 乗に比例する。
- 2 データの内容には依存せず、およそ N の 3 乗に比例する。
- 3 データの内容によって異なるが、平均的にはおよそ N の 2 乗に比例する。
- 4 データの内容によって異なるが、平均的にはおよそ N の 3 乗に比例する。

解説

選択ソートの算法は以下で与えられる：

```

1: function selection_sort (A[0...N-1])
2: begin
3:   for i = 0 to N - 2 do
4:     i_min := i;
5:     for j = i + 1 to N - 1 do
6:       if A[j] < A[i_min] then
7:         i_min := j;
8:       end if
9:     end for
10:    if i ≠ i_min then
11:      swap(A[i], A[i_min]);
12:    end if
13:  end for
14: end.      ( http://en.wikipedia.org/wiki/Selection_sortを参考に作成 )

```

この算法より、キー値の比較回数は、

$$\begin{aligned} & \sum_{i=0}^{N-2} (N-1-(i+1)+1) \\ &= \frac{N(N-1)}{2} \end{aligned}$$

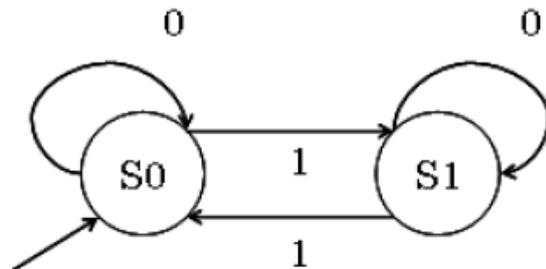
となり、データに依存しないことが分かる。
よって、選択肢 1 が正しい。

*平成 20 年度 問 28

下図の決定性有限オートマトンは 2 つの状態 S_0 と S_1 を持つ。 S_0 は初期状態である。 入力数列は 1 あるいは 0 である。 このオートマトンに関する記述で不適切なものを選択肢の中から一つ選べ。

表 状態遷移表

入力状態	1	0
S_0	S_1	S_0
S_1	S_0	S_1



- 1 入力数列が 00111 のとき、状態は S_1 となる。
- 2 入力数列中に 1 が偶数個あるときは、状態は S_0 となる。
- 3 入力数列中に 01010010101 のとき、状態は S_1 となる。
- 4 入力数列中に 0 が偶数個あるときは、状態は S_1 となる。

解説

選択肢 1 入力系列「00111」によって

$$S_0 \xrightarrow{0} S_0 \xrightarrow{0} S_0 \xrightarrow{1} S_1 \xrightarrow{1} S_0 \xrightarrow{1} S_1$$

と遷移する。よって選択肢 1 は正しい。

選択肢 2 この状態遷移表では、0 のときは現在の状態に留まり、1 のときに状態を変えていることが分かる。

よって、入力数列中に 1 が偶数個あるときは、状態は S_0 となる。

選択肢 3 入力数列中に 1 が奇数個あるので、状態は S_1 となる。

選択肢 4 例えば、入力数列が 00 のとき、状態は S_0 となるので、この選択肢に誤りがある。よって、選択肢 4 が正しい。

*問題文はバイオインフォマティクス技術者認定試験（日本バイオインフォマティクス学会主催）問題（平成19年度または平成20年度）から引用

平成19年度日本バイオインフォマティクス学会（JSBi）バイオインフォマティクス技術者認定試験試験問題
Copyright©2007 Japanese Society for Bioinformatics. All Rights Reserved. :

http://www.jsbi.org/modules/jsbi/index.php/nintei/H19/H19mondai_kaitou.pdf

平成20年度日本バイオインフォマティクス学会（JSBi）バイオインフォマティクス技術者認定試験試験問題
Copyright©2008 Japanese Society for Bioinformatics. All Rights Reserved. :

http://www.jsbi.org/modules/jsbi/index.php/nintei/H20/H20mondai_kaitou.pdf

日本バイオインフォマティクス学会：<http://www.jsbi.org/>