

# アルゴリズムとデータ構造

## 演習第 5 回

### 計算量解析

計算量（アルゴリズム C 第 1 巻 p.77）という概念について学びます。これは、アルゴリズムがどの程度優れているかのひとつの基準となります。

#### 問題 1 [\[印刷用 PostScript\]](#)

(1) 次の  $n$  についての関数を、増加率の小さい順に並べ替えなさい。

$n, n^2, n \log n, \log^{10} n, n^{10}, 2^n, n^3$

(2) 次の各ループの計算量を答えなさい。

1. `for( i=0; i<n; i++) sum++;`
2. `for( i=0; i<n; i++)  
    for( j=0; j<n; j++) sum++;`
3. `for( i=0; i<n; i++)  
    for( j=0; j< n*n;j++) sum++;`
4. `for( i=0; i<n; i++)  
    for( j=0; j<i; j++) sum++;`

(1) 増加率を調べるには微分をします。しかし微分をしなくても、グラフをイメージして並べ換えられると良いです。

(2) アルゴリズムの計算量を表現するにはオーダ記法というものをを用います。例えば、あるアルゴリズムに変数  $n$  があり、計算量が  $n^2$  に比例している場合、これは「計算量が  $O(n^2)$  のアルゴリズム」と呼びます。つまり、 $n$  が3倍になれば、計算量が9倍になります。問題のループの  $n$  の値を変えた場合、計算量（この場合ループ回数）にどのように影響するかを考えてみましょう。

#### 問題 2

問題 1 の (2) のループの実行にかかる時間を、変数  $n$  を変えて計測し、[以下のような表](#)にまとめなさい。時間計測には `gethrtime` 関数を使用しなさい。

1. `+-----+`

n	2000000	4000000	6000000	8000000	10000000
time					

2.

n	2000	4000	6000	8000	10000
time					

3.

n	100	200	300	400	500
time					

4.

n	2000	4000	6000	8000	10000
time					

今回は、プログラムではなく、この表を完成させて提出しなさい。また、結果の考察も書くこと。

実験をして、データを表にする問題です。実験結果が問題 1 (2) で 答えた計算量と一致しているかを確認してください。例えば 1. で、  $n$  が  $2000000 \rightarrow 10000000$  と 5 倍になったとき、 time は何倍に増加していますか？

この問題で使用する gethrtime 関数は、 ナノ秒単位で時間を計測することができます。次の [サンプル](#) を参考にして使用してください。

```
#include <stdio.h>
#include <sys/time.h>

int main(){
    int i,n,sum;
    hrtime_t start,finish;

    printf("n= ");
    scanf("%d",&n);
    start=gethrtime(); /* 計測開始 */

    sum=0;
    for( i=0; i<n; i++) sum++;

    finish=gethrtime(); /* 計測終了 */
    printf("time: %f seconds.\n", (double)(finish-start)/NANOSEC);

    return 0;
}
```

### 問題 3

問題 2 の 4 の結果をグラフ作成ツール（xmgrace）で読み込める形式で出力しなさい。形式は次の通りである。

nの値 時間

$n$ の値は、500 刻みで 500 から 10000 までとする。

実行例：

```
% ./a.out > ex05-3.dat
% cat ex05-3.dat
500 0.006423
1000 0.026013
1500 0.058152
2000 0.103227
2500 0.162220
3000 0.231876
3500 0.316341
4000 0.422904
4500 0.525531
5000 0.645877
5500 0.780517
6000 0.933092
6500 1.095727
7000 1.267396
7500 1.455250
8000 1.655440
8500 1.873892
9000 2.096666
9500 2.342806
10000 2.582010
% xmgrace ex05-3.dat &
```

一回の実験（時間計測開始、ループ実行、時間計測終了）を関数にして、 $n$ の値を引数として受け取るようにすると作りやすいです。

結果を `xmgrace` で表示させて、計算量とグラフの形が一致していることを確認してください。

`xmgrace` の使い方については、[リテラシーの資料](#)（11章、109ページ 11.2 Xmgraceの利用）を参考にしてください。

---

Written by わかまつなおき