

H17年度ソフトウェア工学試験問題

日時：平成17年7月29日（金）14:30-16:00 場所：K318

注意事項：

- ① 指定された席に着席すること。
- ② 答えは机に残して退室すること。
- ③ 問題用紙は持ち帰ること。
- ④ 試験用紙、計算用紙が必要なときには申し出ること。
- ⑤ 解の簡単な導出過程も書くこと。

1. 漸化式の計算

フィボナッチ数列は、次の漸化式で定義される。

$$f(n) = \begin{cases} 0 & n = 0 \\ 1 & n = 1 \\ f(n-1) + f(n-2) & n \geq 2 \end{cases}$$

このフィボナッチ数列を計算するアルゴリズム2つを、C言語風に以下に示す。なお、変数宣言等は省略している。

アルゴリズム A:

```
int algoA(int n){
    if(n==0){
        return 0;
    }else if(n==1){
        return 1;
    }else{
        return algoA(n-1)+algoA(n-2);
    }
}
```

アルゴリズム B:

```
int algoB(int n){
    F[0]=0;
    F[1]=1;
    for(i=2;i<=n;i++){
        F[i]=F[i-1]+F[i-2];
    }
    return F[n];
}
```

この2つのアルゴリズムに関して、以下の問いに答えよ。

- (1) 入力サイズを n としたとき、アルゴリズム A の最悪時間計算量 $T_A(n)$ が満たすべき漸化式を示せ。
- (2) (1) の漸化式を解き、アルゴリズム A の最悪時間計算量を O 記法で答えよ。
- (3) アルゴリズム B の最悪時間計算量 $T_B(n)$ を O 記法で答えよ。

2. クイックソート

配列 A に次のようにデータが蓄えられているとする。

	A[0]	A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]
A	12	4	5	11	8	2	15	7

この配列をソートするために以下に示すようなクイックソートを用いる。

```
void qsort(int left, int right){
    int pos ; /*分割位置*/
    if(left >= right){
        return;
    }else{
        pos = partition(left, right);
        print_array(left, right);
        qsort(left, pos - 1);
        qsort(pos + 1, right);
    }
}
```

ここで、`partition` はピボット（基準値）の位置を求める関数で、部分配列 `A[left]-A[right]` をピボットより小さい要素、ピボット、ピボットより大きい要素の順に並べかえて、ピボットの入る配列の添え字を返す関数である。また、`print_array(left, right)` は、部分配列 `A[left]-A[right]` を表示する関数である。

このとき、クイックソートにおける動作について答えよ。

- (1) ピボットを部分配列の右端、すなわち、`A[right]` とする。このとき、`print_array` によって表示される配列の内容をすべて示せ。
- (2) クイックソートが最も悪い動作をする入力例を示せ。ただし、ピボットは部分列右端をいつも選ぶものとし、データ個数は8個としてよい。
- (3) n 個のデータをソートするとき、クイックソートの最悪時間計算量を O 記法で示せ。また、なぜそのような最悪時間計算量になるのかの理由も記述せよ。

3. 2分探索

配列Bに次のようにデータが蓄えられているとする。

	B[0]	B[1]	B[2]	B[3]	B[4]	B[5]	B[6]	B[7]
B	b ₀	b ₁	b ₂	b ₃	b ₄	b ₅	b ₆	b ₇

この配列に次のようなプログラムで2分探索を行なおうとしている。

```
int binary_search(int key,int left,int right){
    int mid;/*中間位置*/
    if(left>right){return -1;/*存在しない*/}
    else{
        mid=(left+right)/2;
        if(B[mid]==key)return mid;
        else if(key < B[mid]) {
            return binary_search( (a) );/*小さい方*/
        }
        else {
            return binary_search( (b) );/*大きい方*/
        }
    }
}
```

- (1) 2分探索に必要な条件を述べよ。
- (2) (a)、(b)に入る適切な記号を示せ。
- (3) 2分探索の最悪時間計算量をO記法で示せ。また、なぜそのような最悪時間計算量になるのかの理由も記述せよ。

4. データ構造

スタックおよびキューに関して、以下の問いに答えよ。

(1) 空のスタックに、次のような操作系列を行なったとき、`pop()`で取り出されるデータの系列および、最終的なスタックの状態を示せ。

`push(7)→push(3)→pop()→push(4)→push(2)→push(9)→pop()→push(1)→pop()→pop()→push(8)→push(5)→pop()`

(2) 空のキューに次のような操作系列を行なったとき、`dequeue ()`で取り出されるデータ系列および、最終的なキューの状態を示せ。

`enqueue(7)→enqueue(3)→dequeue()→enqueue(4)→enqueue(2)→enqueue(9)→dequeue()→enqueue(1)→dequeue()→dequeue()→enqueue(8)→enqueue(5)→dequeue()`