2007年度ソフトウェア工学再試験問題

日時：2007年9月7日(金) 3時限 12:50-14:20

場所：K325

注意事項：

① 答案は教卓へ提出すること。

② 問題用紙は持ち帰ること。

③ 試験用紙、計算用紙が必要なときには申し出ること。

④ 解の簡単な導出過程も書くこと。

1.漸近量評価

（１）次の関数を O 記法で示せ。(出来るだけ漸近計算量が小さいもので表すこと。)

$(a)\quad T_a(n) = 2n^3 - 10n^2 + 7$

$(b)\quad T_b(n) = (4n-1)(3n-3) + n(n+5)(n-3) + 8(n^2+5)(n-3)$

$(c)\quad T_c(n) = n\sqrt{n} + n\log n + n\cos n$

$(d)\quad T_d(n) = n^2 + \left(\log n\right)^2 + 8^{\log n}$

$(e)\quad T_e(n) = \dfrac{7(n+5)(2n-1)}{\sqrt{n+3}} + \dfrac{2(n^2+5)(n-3)}{n-4} + \dfrac{(3n^3+1)(n-2)}{\sqrt{2n^3+5}}$

（２）次の C 言語風擬似コードの漸近計算量を O 記法で示せ。

ただし、以下のコードでは引数 n が入力サイズとし、すべて $n = 2^k$ の形であるとする。

(a) 漸近計算量 $T_A(n)$

アルゴリズム A：

```
void algoA(int n){
    for(i=0;i<10;i++){
        for(j=0;j<n;j++){
            (定数 c_a 時間の処理)
        }
    }
    return;
}
```

(b) 漸近計算量 $T_B(n)$

アルゴリズム B：

```
void algoB(int n){
    for(i=0;i<n;i++){
        for(j=i;j<n;j++){
            (定数 c_b 時間の処理)
        }
    }
    return;
}
```

(c) 漸近計算量 $T_C(n)$

アルゴリズム C：

```
void algoC(int n){
   for(i=0;i<n;i=++){
      for(j=1;j<n;2*j){
           (定数 c_c 時間の処理)
       }
   }
   return;
}
```

(d) 漸近計算量 $T_D(n)$

アルゴリズム D：

```
void algoD(int n){
   if(n==0){
      return;
   }else{
      algoD(n-1);
      (定数 c_d 時間の処理)
      algoD(n-1);
     return;
   }
}
```

２．多項式の計算

下のアルゴリズムは多項式 $f(x)=a_0+a_1x^1+\cdots+a_nx^n$ の関数値を計算するホーナーの方法である。ホーナーの方法では、

$$f(x)=a_0+a_1x^1+\cdots+a_nx^n=\left(a_0+\left(\cdots\left(a_{n-1}+\left(a_n\right)x\right)\cdots\right)x\right)$$

の順序で関数値を計算する。各係数 a_i は配列 $A[]$、最高次数 n は n として引数として与えられる。また、引数 x は $f(x)$ における x を表す。この方法について以下の問いに答えよ。

ホーナーの方法:

```
/*多項式を計算するホーナーの方法(繰り返し版)*/
1. double for_horner(double A[],int n,double x){
2. int i;/*ループカウンタ*/
3. double fx＝A[n]; /*関数値f(x)を表す*/
4.   for(i=1;i<=n;i++){
5.     printf(“%f¥n”,fx);
6.     fx=A[n-i]+fx*x;
7.   }
8.   return fx;
9. }
```

(1)関数 $f(x)=6+5x-7x^2+3x^3+5x^4-2x^5$ の $x=-2$ における関数値 $f(-2)$ を上のアルゴリズムで求めるとき、５行目の printf 文で表示される系列を示せ。

(2)上の疑似コードの最悪時間計算量 $T_h(n)$ を O 記法で示せ。

(3)ホーナーの方法を再帰的に実現するアルゴリズムを C 言語風の疑似コードにより示せ。ただし、以下のプロトタイプ宣言を持つとする。

```
/*ホーナーの方法を実現する再帰的関数*/
double rec_horner(double A[],int n,double x);
```

３．マージソート

配列 A には n 個の要素 $a_0, a_1, \cdots, a_{n-1}$ が $A[0], A[1], \cdots, A[n-1]$ にそれぞれ蓄えられているとする。この配列 A をマージソートにより昇順にソートすることを考える。マージソートのアルゴリズムを下に擬似コードで示す。$l < m < r$ に対して、関数 merge は昇順の部分配列 $A[l] \cdots A[m]$ と昇順の部分配列 $A[m+1] \cdots A[r]$ から、部分配列 $A[l] \cdots A[r]$ を昇順にする。関数 `print_array(l,r,A)`は部分配列 $A[l] \cdots A[r]$ の内容を表示する。このとき、以下の問いに答えよ。

マージソート：

```
1.  void m_sort(int l,int r,double A[N]){
2.    int m=(l+r)/2;/*中央の添え字*/
3.    if(l>=r){
4.      return;
5.    }else{
6.         /*print_array(l,r,A);*/
7.      m_sort( (ア) );/*前半のソート*/
8.      m_sort( (イ) );/*後半のソート*/
9.        /*print_array(l,r,A);*/
10.     merge(l,m,r,A);
11.       /*print_array(l,r,A);*/
12.     return;
13.   }
14. }
```

(1) マージソートが動作するように、(ア)、(イ) に入る適切な引数を示せ。

(2) $n = 8$ とし、配列 A に下のように値が蓄えられているとする。`m_sort(0,7,A)`を実行したときに 6,9,11 行目の各 `print_array(0,7,A)`により表示される配列の内容を示せ。

	A[0]	A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]
A	79	28	60	41	57	11	85	73

(3)マージソートの時間計算量を $T_m(n)$ と表す。このとき、$T_m(n)$ が満たすべき漸化式を示せ。ただし、関数 merge の時間計算量は $O(r-l) = O(n)$ 時間であるとし、表示用関数 `print_array` の計算量は含めないとする。

(4)(3)の漸化式を解き、マージソートの最悪時間計算量 $T_m(n)$ を O 記法で示せ。ただし、$n = 2^k$ であるとして良い。

４．２分探索木

２分探索木 T に集合 $S=\{15,22,35,47,50,52,61\}$ を蓄えることを考える。２分探索木 T にデータ x を挿入する操作を `insert(x)` で表わし、２分探索木 T からデータ x を削除する操作を `delete(x)` と表す。ただし、節点 x に子が２つあるとき、`delete(x)` では、削除される頂点の場所には､左の部分木中の最大の要素で置き換わる。(この T は単なる２分探索木で AVL 木ではない。) このとき、以下の問いに答えよ。

(1) 集合 S を蓄える２分探索木 T で高さが最大となるもの図示せよ。(１種類示せばよい。)

(2) 集合 S を蓄える２分探索木 T で高さが最小となるもの図示せよ。(１種類示せばよい。)

(3)空の（要素数 ０ の） ２分探索木 T に次の順序で操作を行ったとき、２分探索木 T の状態の変化を図示せよ。

$$\mathrm{insert}(50)\to\mathrm{insert}(47)\to\mathrm{insert}(52)\to\mathrm{insert}(61)\to\mathrm{insert}(22)\to\mathrm{insert}(35)\to\mathrm{insert}(15)$$

(4)(3)の木にさらに次の順序で操作を行ったとき、２分探索木 T の状態の変化を図示せよ。

$$\mathrm{insert}(44)\to\mathrm{delete}(61)\to\mathrm{delete}(47)\to\mathrm{insert}(18)$$
$$\to\mathrm{delete}(22)\to\mathrm{insert}(39)\to\mathrm{delete}(50)\to\mathrm{delete}(44)$$

(5) 空の２分探索木 T に n 回を挿入を繰り返して２分探索木 T を作成するときの最悪時間計算量 $T_{BT}(n)$ を示せ。