

## 1 動作トレース (20)

```
int foo( int x , int y ) {  
    if ( x == 0 )  
        return 0 ;  
    else if ( x % 2 == 1 )  
        return foo( x / 2 , y * 2 ) + y ;  
    else  
        return foo( x / 2 , y * 2 ) ;  
}  
void main() {  
    printf( "%d\n" , foo( 6 , 10 ) ) ;  
    printf( "%d\n" , foo( 42 , 100 ) ) ;  
}
```

以下のプログラムの  
実行結果を答えよ。

-----  
-----

## 2 プログラム作成 (20)

1件あたり、名前, 国語, 算数, 理科の成績のデータを、動的メモリ上の配列に全件読み込み、その後で名前と平均点を出力するプログラムを作成せよ。入力と期待する出力の例を以下に示す。

処理に必要な構造体の (1) 宣言、(2) 入力処理、(3) 平均出力の処理を記述せよ。(5+8+7)

入力		出力
3	←データ件数	tohru 71.00
tohru	58 82 73	tomoko 78.33
tomoko	83 68 84	mitsuki 81.67
mitsuki	64 89 92	

### 3 プログラム穴埋め (20)

昇順に並べられた整数配列の指定した範囲の中から、2分探索法にて目的の値を探す関数 `find`(配列, 先頭, 末尾, 探す値) の下線部を埋めてプログラムを完成せよ。10 件の配列から探す場合には、先頭=0, 末尾=10 を渡すものとする。

```
int find( int array[] , int L , int R , int key ) {
    if ( L == R ) {           // 見つかったら 1, 見つからなかったら 0
        _____;
    } else {
        int M = _____ ;
        if ( _____ )
            return 1 ;
        else if ( array[ M ] > key )
            return find( array , _____ ) ;
        else
            return _____ ;
    }
}

void main() {
    if ( find( a , 0 , 10 , 10 ) )
        printf( "Find.\n" );
}

| int a[ 10 ] = {
|     7, 10, 12, 15, 22,
|     31, 48, 55, 72, 89
| } ;
```

### 4 オーダ問題 (20)

1. データ件数  $N$  でソートプログラムの処理時間が  $O(N \log N)$  で示されるとする。  $N$  が 1000 件で 3[msec] の時間がかかった場合、  $N$  が 100000 件での処理時間を予想せよ。(10)
2. 前設問の `find()` で、データ件数  $N$  における処理時間  $T_{find}(N)$  にふさわしい再帰方程式と、その処理時間をオーダ記法で示せ。(5x2)

## 5 データ構造のイメージ (20)

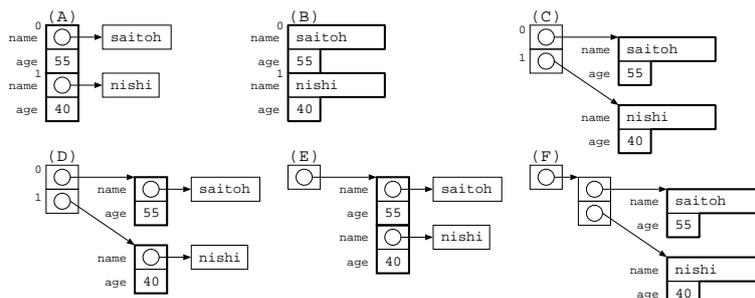
以下のプログラムで、構造体にて名前と年齢の情報を保存している。このデータ構造 (1), (2), (3) のイメージにふさわしい図を (A)~(F) から選べ。また、各構造体の a[1], b[1], c[1] に保存されている nishi 40 を表示するのにふさわしい printf() の式を答えよ。

```

struct NameAge1 {
    char name[ 20 ] ;
    int age ;
};
struct NameAge2 {
    char* name ;
    int age ;
};
void setNameAge1( struct NameAge1* p, char s[], int a ) {
    strcpy( p->name , s ) ;
    p->age = a ;
};
void setNameAge2( struct NameAge2* p, char s[], int a ) {
    p->name = strdup( s ) ;
    p->age = a ;
};
void main() {
    struct NameAge1 a[2] ; /* (1) */ (2)_____ (3)
    setNameAge1( &a[0] , "saitoh" , 55 ) ;
    setNameAge1( &a[1] , "nishi" , 40 ) ; (3)_____ (3)
    struct NameAge1* b[2] ; /* (2) */
    b[0] = (struct NameAge1*)malloc( sizeof( struct NameAge1 ) ) ;
    setNameAge1( b[0] , "saitoh" , 55 ) ;
    b[1] = (struct NameAge1*)malloc( sizeof( struct NameAge1 ) ) ;
    setNameAge1( b[1] , "nishi" , 40 ) ;
    struct NameAge2* c ; /* (3) */
    c = (struct NameAge2*)malloc( sizeof( struct NameAge2 ) * 2 ) ;
    setNameAge2( &c[0] , "saitoh" , 55 ) ;
    setNameAge2( &c[1] , "nishi" , 40 ) ;

    printf("%s%d\n", _____ ) ; // a[1]
    printf("%s%d\n", _____ ) ; // b[1]
    printf("%s%d\n", _____ ) ; // c[1]
}

```



## 6 説明問題 (20)

- 極めて大量のメモリを使用するプログラムがあった場合、並行動作するプログラムにどのような影響が発生するか、具体的に理由を挙げて説明せよ。(10)
- プログラムを極めて短時間で開発することによって発生する問題を2つあげ、問題点について具体的に説明せよ。(10)