

## 1 オーダ (必須:30 点)

1. 以下の一般式で示される処理時間をオーダ記法で答えよ。

$$T(N) = T_\alpha + T_\beta \times N^{1.2} + T_\gamma \times N \log N$$

2. ハノイの塔を解くプログラムのオーダは、ディスクの枚数を  $N$  としたとき、 $O(2^N)$  で示される。ディスクが 10 枚の処理時間が、1 秒であった時、13 枚の処理時間は、何秒かかるか予測せよ。
3. あるプログラムの処理時間のオーダが、 $O(\sqrt{N})$  であった。プログラムのデータ件数が 100 件の時の処理時間が 50 msec であった。このとき、プログラムのデータ件数が 10000 件の時の処理時間を予測せよ。

## 2 再帰 (必須:30 点)

このプログラムの下線部 (C) を呼び出した時の以下の設問に答えよ。

```
void msort( int array[] , int size )
{ int i , m , t ;
  if ( size > 1 ) {
    m = size - 1 ;
    for( i = 0 ; i < size - 1 ; i++ )
      if ( array[ i ] > array[ m ] )
        ~~~~~(A)
        m = i ;
        t = array[ m ] ;
        ~~~~~(B)
        array[ m ] = array[ size - 1 ] ;
        array[ size - 1 ] = t ;
        msort( array , size - 1 ) ;
  }
}
```

答えよ。  
}

```
int a[ 3 ] = { 12 , 8 , 20 } ;
```

```
void main() {
  msort( a , 3 ) ;
} ~~~~~(C)
```

設問 1

下線部 (A), (B) の式を実行した回数を答えよ。

(A) \_\_\_\_\_ (B) \_\_\_\_\_

設問 2

(C) の処理が終わった直後の配列 a[] の内容を

\_\_\_\_\_ , \_\_\_\_\_ , \_\_\_\_\_

1. このプログラムの処理時間の再帰方程式を示せ。(データ件数を  $N$  とする)

2. プログラムのオーダを示せ。(一般式を導出する時の証明は不要)

### 3 アルゴリズムの選択 (選択:20 点)

1 つの処理を 2 通りのアルゴリズム A,B で作ったら、それぞれの処理時間のオーダーは  $O(\sqrt{N})$ 、 $O(N)$  であった。また実際のデータを使って処理時間を測ったら、 $T_A(100) = 100[\text{msec}]$ ,  $T_B(100) = 50[\text{msec}]$  であった。

この場合、データ件数に応じて、最適なアルゴリズムを選択するには、どうすれば良いか説明せよ。

### 4 説明問題 (選択:20 点)

以下の 2 つの内容を説明せよ。

1. 2 つのプログラム A,B が並列動作しているコンピュータで、プログラム A がメモリを大量に無駄遣いした場合、プログラム B はどのような影響を受けるか、理由と共に説明せよ。
2. 処理時間のオーダーが、 $O(N \log N)$  で示されるような、処理の例をあげよ。

## 5 ヒープメモリ (選択:20 点)

以下のプログラムを実行し、下右図のようなイメージでデータを格納したい。プログラム中の下線部を適切に埋めよ。

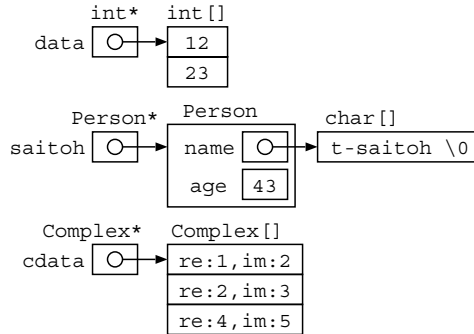
```

struct Person {
    char* name ;
    int age ;
} ;

struct Complex {
    double re ;
    double im ;
} ;

char my_name[ 20 ] = "t-saitoh" ;

```



```

void main() {
    int* data ;
    struct Person* saitoh ;
    struct Complex* cdata ;
    FILE* fp ;
    int size ;

    if ( (data = (int*)malloc( _____ )) != NULL ) {
        data[ 0 ] = 12 ;
    }

    if ( (saitoh = (struct Person*)malloc( sizeof( struct Person ) )) != NULL ) {
        if ( (saitoh->name = _____ ) != NULL ) {
            strcpy( saitoh->name , my_name ) ;
        }

        if ( (fp = fopen( "data.txt" , "rt" )) _____ ) {
            fscanf( fp , "%d" , &size ) ;
            if ( (cdata = (struct Complex*)malloc( sizeof( struct Complex ) * size )) != NULL ) {
                int i ;
                double x , y ;
                for( i = 0 ; fscanf( fp , "%lf %lf" , &x , &y ) == 2 ; i++ ) {
                    _____(F)
                }
            }
        }
        fclose( fp ) ;
    }

    if ( data != NULL ) free( data ) ;

    if ( _____ ) free( saitoh->name ) ;
    if ( saitoh != NULL ) free( saitoh ) ;
    if ( cdata != NULL ) free( cdata ) ;
}

```

|| ファイル data.txt の内容  
|| 3  
|| 1.0 2.0  
|| 2.0 3.0  
|| 4.0 5.0