

以下の 5 つの設問の中から、4 つを選んで解答せよ。

1 2 分探索木 (25)

電話番号と名前のデータベースを 2 分探索木で作る。データは、名前をキーとして検索するものとして、以下のようなプログラムを作成した。(同じ名前、同じ電話番号は無いものとする)

1. プログラム中の下線部 (A) ~ (E) に相応しい処理を埋めよ。(15)

2. このデータの中から、指定した電話番号を検索し出力する関数 `public static void find( PNTreeNode p )` を作成せよ。(10) `public static void find(PNTreeNode p, int ph)`

```
public class Main {
    public static PNTreeNode entry( PNTreeNode p , int ph , String nm ) {
        if ( p == null )
            return new PNTreeNode( ph , nm ) ;
        else if ( p.name.compareTo( nm ) > 0 )
            p.left = entry( _____ , ph , nm ) ;
        else
            p.right = _____ ;
        return p ;
    }
    public static void find( PNTreeNode p , int ph ) {
        //
    }
    public static void main(String[] args) {
        PNTreeNode top = null ;
        top = entry( top , 621111 , "nitfc" ) ;
        top = entry( top , 272925 , "tsaitoh" ) ;
        top = entry( top , 110 , "POLICE" ) ;
        find( top , 110 ) ;
    }
}

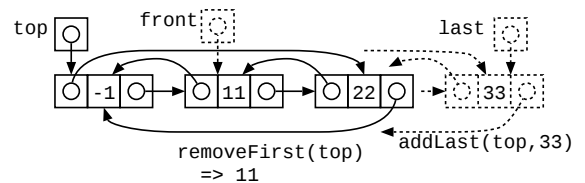
// str1.compareTo(str2) は文字列を比較し
// 同じなら 0 , str1 > str2 で正の値 ,
// str1 < str2 なら負の値を返す。

class PNTreeNode {
    int phone ;
    String name ;
    PNTreeNode left ;
    PNTreeNode right ;

    PNTreeNode( int ph, String nm ){
        _____ = ph ;
        _____ (D)3
        _____ = nm ;
        _____ (E)3
        this.left = null ;
        this.right = null ;
    }
}
```

2 Deque 問題 (25)

循環双方向リストを用いて、Deque を作りたい。
最初に、右図に示すようなイメージ図となるように、
main() 関数内の (A), (B), (C) に適切な命令を埋め
よ。また、先頭要素を取り出す removeFirst(), 末尾
に要素を加える addLast() の関数内を完成させよ。



```
public class Main {
    public static void addLast(
        ~~~~~(D)2 top , int x ) {
        DequeNode last = new DequeNode(
            ~~~~~(E)3 , x , top ) ;
        top.prev.next = last ;
        top.prev = ~~~~~(F)3 ;
    }
    public static int removeFirst( DequeNode top ) {
        DequeNode front = top.next ;
        int ans = front.data ;

        top.next = ~~~~~(G)4 ;
        front.next.prev = ~~~~~(H)4 ;
        return ans ;
    }
    public static void main(String[] args) {
        DequeNode top = new DequeNode( null , -1 ,
            new DequeNode( null , 11 ,
                new DequeNode( null , 22 , null ) ) ) ;
        top.next.prev = top ;
        top.next.next.prev = ~~~~~(A)3 ;
        top.prev = ~~~~~(B)3 ;
        top.next.next.next = ~~~~~(C)3 ;
        addLast( top , 33 ) ;
    }
}
```

```
class DequeNode {
    DequeNode prev ;
    int data ;
    DequeNode next ;
    DequeNode( DequeNode p ,
        int x ,
        DequeNode n ) {
        this.prev = p ;
        this.data = x ;
        this.next = n ;
    }
}
```

3 説明問題 (12+13)

1. java で `var stack = new LinkedList<int>()` ; のような式で使われている機能について説明せよ。
Integer
2. AVL 木について、その特徴や具体的な処理の一例などを図等を交えて説明せよ。

4 基本問題 (25)

```
public class Main {  
    public static void print_b( int[]array, int idx ) {  
        for( int i = idx ; i < array.length ; i++ )  
            System.out.print( array[ i ] + " " ) ;  
    }  
    public static void print_d( int[]array, int idx ) {  
        if ( idx < array.length ) {  
            System.out.print( array[ idx ] + " " ) ;  
            print_d( array , idx * 2 + 1 ) ;  
            print_d( array , idx * 2 + 2 ) ;  
        }  
    }  
    public static void main(String[] args) {  
        int[] a = { 21, 14, 39, 9, 18, 31, 41 } ;  
        print_b( a , 0 ) ; /* (A) */  
        System.out.println();  
        print_d( a , 0 ) ; /* (B) */  
        System.out.println();  
    }  
}
```

| 配列に保存された
| 2分ヒープのデータ
| を表示する
| (A) print_b()
| (B) print_d()
| のプログラムを
| 作成した。
| 各処理の実行結果と
| それぞれが何優先
| 探索にあたるか答えよ
|
| (A) _ _ _ _ _
| _ _ _ _ _ 優先探索
| (B) _ _ _ _ _
| _ _ _ _ _ 優先探索
| (18+4+3)

5 正規表現とバックス記法 (25)

1. 正規表現 $[A-Z]^+[0-9]^*$ が表現する文字列の集合の要素となるものに○、ならないものは×を答えよ。(2x4)
[A-Z] は、英字 1 文字を表す。[0-9] は数字 1 を表す。*は、直前の正規表現の 0 回以上の繰り返しを表す。+は直前の正規表現の 1 回以上の繰り返しを表す。
(a) 456789 _ _ _ , (b) ABC+99 _ _ _ , (c) ABC99* _ _ _ , (d) ABCDEF _ _ _
2. 次の規則から生成することができる式はどれか。
生成できるものに○、出来ないものに×を答えよ。(2x4)
<式> ::= <変数> | (<式> + <式>) | <式> * <式>
<変数> ::= A | B | C | D
(a) $A+(B+C)*D$ _ _ _ , (b) $(A+B)+(C+D)$ _ _ _ , (c) $(A+B)*(C+D)$ _ _ _ , (d) $(A*B)+(C*D)$ _ _ _
3. バイトコードインタプリタ方式について説明せよ。(9)