

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)  
下線 (a), (b) は後の設問で答えてください。

- プログラム中の下線部 (A), (B), (C) に相応しい処理を埋めよ。
- 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 void find(int) を作成せよ。  
↓名前を出力

```
void entry( int ph , const char nm[] ) {
    struct PNTree** tail = &top ;
    while( (*tail) != NULL ) {
        int cmp = strcmp( (*tail)->name , nm ) ;
        if ( cmp == 0 ) ~~~~~(a)
            break ;
        else if ( cmp > 0 )
            tail = &( (*tail)->left ) ;
        else ~~~~~(b)
            tail = ~~~~~ ;
    }
    if ( (*tail) == NULL ) {
        *tail = (struct PNTree*)malloc( ~~~~~ ) ;
        if ( (*tail) != ~~~~~ ) {
            (*tail)->phone = ph ;
            strcpy( (*tail)->name , nm ) ;
            (*tail)->left = (*tail)->right = NULL ;
        }
    }
}

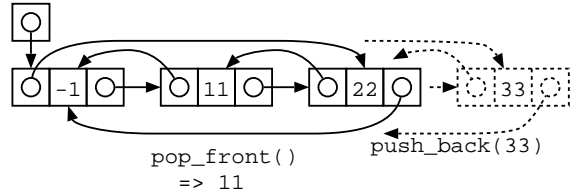
void find( int ph ) { // 以下を記述せよ (16)
```

```
struct PNTree {
    int phone ;
    char name[ 10 ] ;
    struct PNTree* left ;
    struct PNTree* right ;
} ;
struct PNTree* top = NULL ;
// strcmp(A,B) は文字列比較関数
// A=B なら 0, A>B で 1, A<B で -1 を返す。
```

```
int main() {
    entry( 272925 , "tsaitoh" ) ;
    entry( 110 , "POLICE" ) ;
    entry( 621111 , "nitfc" ) ;
    find( 110 ) ;
}
```

## 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 pop\_front(), 末尾に要素を加える push\_back() の関数内を完成させよ。  
下線 (c), (d), (e) は後の設問で答えてください。



```
struct Deque {
    struct Deque* prev ;
    int data ;
    struct Deque* next ;
};

void push_back( struct Deque* p , int x ) {
    p->prev->next = dq_cons( x , p );
    p->prev = p->prev->next ;
}

struct Deque* dq_cons( struct Deque* p , int x , struct Deque* n ) {
    struct Deque* ans = (struct Deque*)malloc( sizeof( struct Deque ) );
    if ( ans != NULL ) {
        ans->prev = p ;
        ans->data = x ;
        ans->next = n ;
    }
    return ans ;
}

int pop_front( struct Deque* p ) {
    struct Deque* del = p->next ;
    int ans = del->data ;
    p->next = del->next ;
    del->next->prev = p ;
    return ans ;
}

int main() {
    struct Deque* top
        = dq_cons( NULL , -1 ,
                  dq_cons( NULL , 11 ,
                          dq_cons( NULL , 22 , NULL ) ) ) ;
    top->next->prev = top ;
    top->next->next->prev = top ;
    top->prev = top ;
    top->next->next->next = top ;
    printf( "%d\n" , pop_front( top ) ) ; // 11 を取り出す
    push_back( top , 33 ) ;
}
```

## 3 説明問題 (12+13)

- C 言語のソースが機械語に変換されるまでにどのような処理が行われるか、その処理の概要もあわせて説明せよ。
- AVL 木について、その特徴や具体的な処理の一例などを交えて説明せよ。

## 4 基本問題 (25)

```
int a[] = { 21, 14, 39, 9, 18, 31, 41 } ;
void print_d( int a[], int x, int sz ) {
    if ( x < sz ) {
        print_d( a , 2*x + 1 , sz ) ;
        printf( "%d " , a[ x ] ) ;
        print_d( a , 2*x + 2 , sz ) ;
    }
}
void print_w( int a[], int x, int sz ) {
    for( int i = x ; i < sz ; i++ )
        printf( "%d " , a[ i ] ) ;
}
int main() {
    print_w( a , 0 , 7 ) ; /*A*/
    print_d( a , 0 , 7 ) ; /*B*/
    return 0 ;
}
```

実行結果を答えよ。  
また、表示する際の木をなぞる処理はそれぞれ何優先探索と呼ばれるか？  
(A) \_\_\_\_\_ (7) \_\_\_\_\_ 優先探索 (2)  
(B) \_\_\_\_\_ (14) \_\_\_\_\_ 優先探索 (2)

## 5 説明問題 (25)

1. 前の設問中の (a) ~ (e) について型を答えよ。

(a) \_\_\_\_\_ (b) \_\_\_\_\_ (c) \_\_\_\_\_  
(d) \_\_\_\_\_ (e) \_\_\_\_\_ (2x5)

2. 右図に示すような位数 2 の B 木において、一般的な要素の追加処理で、13 と 50 を加えた場合、どのように B 木に要素が追加されるのか、図の中に具体的な値を書き込んで、処理の概要を説明せよ。(5+10)

