

6つの設問の中から、5つを選んで回答せよ。

1 2分木 (20)

名前と年齢を要素とする2分木を処理するプログラムを作った。

```

struct NameAgeTree {
    char name[ 10 ] ;
    int age ;
    struct NameAgeTree* left ;
    struct NameAgeTree* right ;
} ;

struct NameAgeTree* tcons( char n[] , int a ,
                           struct NameAgeTree* l,
                           struct NameAgeTree* r )
{
    struct NameAgeTree* nw ;
    nw = (struct NameAgeTree*)
        malloc( sizeof(struct NameAgeTree) ) ;
    if ( nw != NULL ) {
        strcpy( nw->name , n ) ;
        ~~~~~(A)
        nw->age = a ;
        nw->left = l ;
        nw->right = r ;
    }
    return nw ;
}

int main() {
    char str[ 10 ] ;
    int age ;
    struct NameAgeTree* top = NULL ;
    while( scanf( "%s%d" , str , &age ) == 2 ) {
        struct NameAgeTree* n = tcons( str , age , NULL , NULL ) ;
        struct NameAgeTree** ppt = &top ;
        while( *ppt != NULL ) {
            ~~~~(C)
            int ans = comp_NameAge( *ppt , n ) ;
            if ( ans == 0 ) ~~~~(D)
                break ;
            else if ( ans > 0 )
                ppt = &((*ppt)->left) ;
            else ~~~~~(E)
                ppt = &((*ppt)->right) ;
        }
        if ( *ppt == NULL )
            *ppt = n ;
        ~~~~(F)
    }
    print( top ) ;
    return 0 ;
}

void print( struct NameAgeTree* p ) {
    if ( p != NULL ) {
        print( p->left ) ;
        printf( "%s %d\n" , p->name , p->age ) ;
        print( p->right ) ;
    }
}

int comp_NameAge( struct NameAgeTree* a ,
                  struct NameAgeTree* b )
{
    int scmp = strcmp( a->name , b->name ) ;
    if ( scmp > 0 ) // 大小で
        return 1 ; // a>b:1, a=b:0, a<b:-1
    else if ( scmp < 0 )
        return -1 ;
    else if ( a->age > b->age )
        return 1 ; ~~~~~(B)
    else if ( a->age < b->age )
        return -1 ;
    else
        return 0 ;
}
    
```

```

+-----+
| 入力として与える | 左に示したデータを
| def 12           | 与えた時の完成した
| abc 23           | データのイメージ図
| ghi 34           | を示せ。(×8)
+-----+
    
```

下線部の型を答えよ (2 × 6)

- (A) _____
- (B) _____
- (C) _____
- (D) _____
- (E) _____
- (F) _____

2 2分木の処理速度 (20)

設問1のような2分木で検索する場合、データ件数を N としたとき、検索時間のオーダーは、 $O(\log N)$ となることを分かりやすく説明せよ。

3 2分木で再帰 (20)

設問1のような2分木で、指定した年齢の人だけ、名前と年齢をすべて表示したい。
年齢を引数として、結果を標準出力に出力する関数print_age(...)を作成せよ。

4 ハッシュ法

名前と電話番号のデータベースをチェーン法によってプログラムを作成した。下線部にふさわしい処理を記入せよ。

```
#define HASH_SIZE 100
struct NPList {
    int phone ;
    char name[ 10 ] ;
    struct NPList* next ;
} ;

struct NPList* table[ HASH_SIZE ] ;

int hash_func( int ph ) {
    return ph % HASH_SIZE ;
}

void print_find( int ph )
{
    struct NPList* p ;
    int index = hash_func( ph ) ;
    for( p = table[ index ] ; p != NULL ; p = p->next ) {
        if ( _____ ) {
            printf( "%d -> %s\n" , p->phone , p->name ) ;
            return ;
        }
    }
}

int main() { char str[ 10 ] ; int tel ;
    while( scanf( "%d %s" , &tel , str ) == 2 ) // 入力をハッシュに登録
        entry( tel , str ) ;
    print_find( 272925 ) ; // 272925 を検索し表示
    return 0 ;
}
```

```
void entry( int ph , char* nm ) {
    int index = _____ ; (A)
    struct NPList* n ;
    n = (struct NPList*)
        malloc( _____ ) ; (B)
    if ( _____ ) { (C)
        _____ (D)
        strcpy( n->name , nm ) ;
        _____ (E)
        table[ index ] = n ;
    }
}
```

5 再帰のトレース (20)

```
int array[3] = { 34 , 12 , 56 } ; // 左に示すプログラムにおいて、
void print_hash( int i , int size ) // 下線部を実行する順序を答えよ。
{ if ( i < size ) { // 式 (B), (D) については、引数にどのような値が
    ~~~~~(A) // 渡されるのか、明記すること。
    print_hash( i * 2 + 1 , size ) ;
    ~~~~~(B)
    printf( "%d," , array[ i ] ) ;
    ~~~~~(C)
    print_hash( i * 2 + 2 , size ) ;
    ~~~~~(D)
}
}
int main() {
    print_hash( 0 , 3 ) ;
} ~~~~~(E)
```

6 説明問題 (20)

ハッシュ衝突とはどういう現象か説明し、その対策について、設問 4 で示したような
チェイン法以外の方式 を説明せよ。

(データの登録する時や検索する時について、具体的な例を交えながら説明せよ)