

アルゴリズムとデータ構造

記号表と2分探索木（その2）

演習2-1（必須課題）

- ST_BST2.cおよび関連するプログラムをコンパイルし、動作を確認せよ。
- ランダム及び、昇順の20個の要素を記号表に挿入した場合の実行結果を確認せよ。

この課題は提出不要

演習2-2（加点課題）

- ST_BST2.cを修正して、各節点を根とする部分木に含まれる節点の総数を保持するカウント機能、および、木の形状を表示する機能を実装せよ。
- 提出
 - ランダム及び、昇順の20個の要素を記号表に挿入した場合の実行結果をMoodleのテキストフォームから提出すること。実行結果には以下を含めること。
 - 木の形状、カウント
 - ソースコード（ST_BST2.c）を添付ファイルとして提出すること。

演習2-2: 補足

- 以下の関数の追加によって、木の形状を表示することができる。

```
void STshow(link h, int l){  
    int i=0;  
    if(h != NULL) {  
        STshow(h->r, l+1);  
        for(i=0; i<l; i++) printf("\t");  
        printf(" (%d,%d)\n",key(h->item), h->N);  
        STshow(h->l, l+1);  
    }  
}  
void STshowAll(){ STshow(head, 0); }
```

演習2-2: 表示の例

16649 30886 36915 38335 41421 47793 60492 85386 89383 92777

10 keys 10 distinct keys

