20171107 データ構造アルゴリズム課題

20181107 米村作成

1．二分探索木におけるノードの削除を理解する！
完成までの色々な状態のプログラムを用意しましたので、自分の進み具合と相談しながら使ってみてください！

講義の中で話してきた内容をそのまま書いたのが、myBTree-0.c です。
理解するのに使ってもよいでしょうし、答え合わせに使ってもよいですね！

上記 myBTree-0.c から、消したいノードが「左右とも部分木を持つ」際の削除の工程を全て空っぽにしたものが、myBTree-3.c です。ここから始めても良いでしょう！

上記 myBTree-3.c から、消したいノードが「左右とも部分木を持つ」際の削除の工程に必要な、左部分木から value が最大のノードを探す関数を完成させたものが、myBTree-2.c です。ここから始めても良いですし、答え合わせに使っても良いですね！

上記 myBTree-2.c から､消したいノードが｢左右とも部分木を持つ｣際の削除の工程をさらに進め、左部分木から value が最大のノードを見つけて、それを Tree からよけるところまで完成させたものが、myBTree-1.c です！ここから始めても良いですし、答え合わせに使っても良いですね！

上記 myBTree-1.c から､消したいノードが｢左右とも部分木を持つ｣際の削除の工程をさらに進め、左部分木から value が最大のノードを見つけて、それを Tree からよけて、その後、消したいノードと、よけたノードをつなぎ変えるところまで完成させたものが、って、これで完成ですので、myBTree-0.c に戻りましたねー！

2．ソート機能のある単語帳を作ってみる！（2 年生までの知識の確認問題ですね！）
■単語を複数登録できる
■ソート機能がある
が満たされていれば、どのようなものでもよいです！ので作ってみましょう！
が、作り方の一例としては（米村が決めた設定)、以下のようなパターンがありますねー
・単語帳は 2 次元配列により実現する
・最初にランダムな長さの文字列（正しい単語である必要はない）を複数登録する
・一度単語帳を表示
・ソート（難易度は色々ですね）
　難易度例：　先頭の文字のみ見てソート：難易度は比較的低い
　　文字列全体を見てソート：難易度は少しだけ上がる
・ソート後の単語帳を表示
といったところでしょうか！もちろん、自由に作ってもらって OK ですので～

以上です！