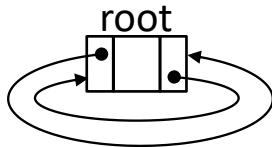


H30 3 年プログラミング演習Ⅱ課題

2018 年 7 月 6 日 米村作成

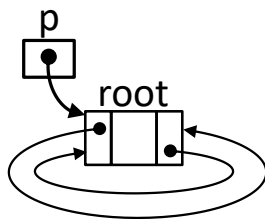
課題 1 (この時間はこれができるだけで十分です！)

さて、サンプルプログラム `0706makeDualList.c` は、水曜に勉強した双方向リストを実現するプログラム的一种ですが、循環リストにもなっています(水曜とは少しだけデータ構造が異なりますね！)
`main()` の `for` 文の前までで初期状態が作成され、図としては次のようになりますね～

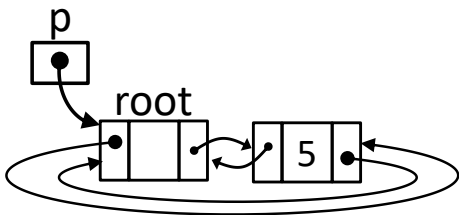


この `root` の真ん中の箱は `root.value` なのですが、値は入れません！ので、この状態はリストが空の状態となります！

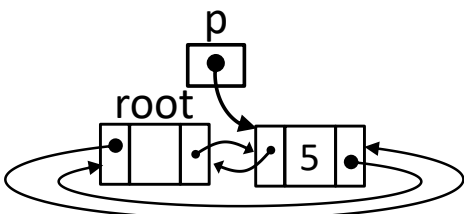
`for` 文以降はメニューになっていて、`insertValueToList` では、任意の場所に `CELL` を挿入することができます！例えば、メニューで「2: 挿入する」を選んで、その後、`data` として 5 を入力し、`basyo` として 0 を入力すると、関数にリスト全体が渡され、`p` は `root` を指す状態になってから(次の図のようになりますね～)、



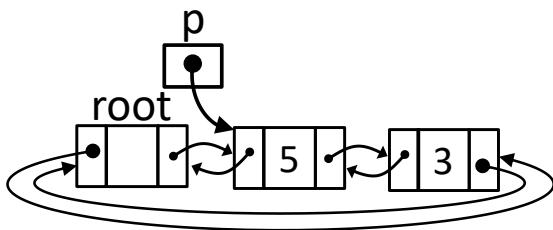
`p` が挿入される場所の手前まで移動し(場所 0 では実際には移動しませんが)、`CELL` が挿入されます！挿入後の図は次のようになりますね～



この状態からさらに、「2: 挿入する」を選んで、`data` として 3 を入力し、`basyo` として、1 を入力すると、まず関数にリスト全体が渡されて、`p` の位置は上の図のようになります、そこから、`p` が一回移動して、下の図のようになります、



CELL が挿入されますね～（下の図のようになります！）



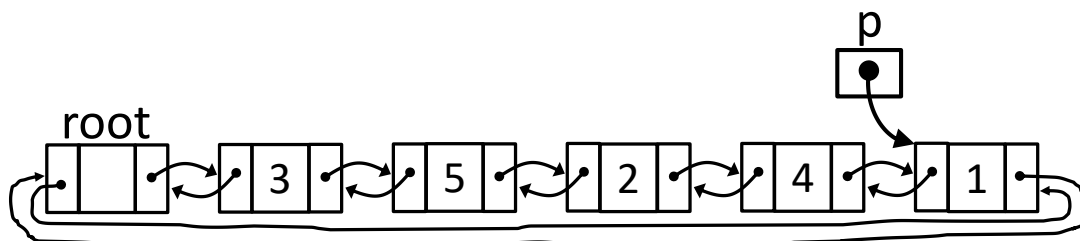
メニューにて「1: リストの内容を出力する」を選ぶと、**PrintList()**が実行され、**next** を辿って、5, 3 と出力され、さらに **prev** を辿って、3, 5 と出力されます！

ということで、任意の場所の **CELL** を削除する **deleteValueToList()**を作ってみましょ～
余裕があれば、**ClearList()**も作ってみてくださいーい！

課題 2（こちらは、課題 1 では物足りない！ってな場合に取り組める系でっす！）

サンプルプログラム **0706makeDualListAndSort.c** は、**mySort()**が実行される前までに、双方向リストを作成しま～す！具体的には、下の図の状態になります！**printList()**により、内容が表示されますね！（両方辿るように作ってありまーす！）

この状態から、**mySort()**が実行されると、**value** を入れ替えてソートされて、さらに、**printList()**が実行され、ソート後の結果が出力されます！



さて、この **mySort()**ですが、値のみを入れ替えたソートになっており、リストの特性を考えると、面白くな～～～い！！

というわけで、きちんとつなぎ変える形のソートに修正してみましょう！