プログラミング演習 第 １１ 回演習レポート

2064229 福場脩真

# 指定された課題番号と課題名

基本課題3「線形リストの削除」

# (2) フローチャートあるいは自然言語によるアルゴリズムの記述

# ダイアグラム 自動的に生成された説明

# アルゴリズムが「正しいこと」である説明あるいは証明

任意の整数（０以外）を入力すると、リストにその登録番号の選手があれば削除され、なければその旨のメッセージを出力した。その後、また入力を要求され、繰り返し処理ができた。0を入力すると、ループが正しく停止した。よってこのプログラムは正しい。

# ソース・プログラムの説明

Listクラスについて

Listクラスは線形リストのノードのクラスである。選手番号、名前、詳細をに入力するとインスタンスが形成され、次のポインタを示すnext関数は初期値Noneで自動的に定義される。

LinkedListクラスについて

線形リストを操作しやすくするために定義した。インスタンスを生成するとインスタンス変数topがListインスタンスとして自動的に生成される。リストにノードを挿入するのは、insetメソッド、リスト表示はshowメソッド、削除はdeleteメソッドである。

LinkedLlistのdeleteメソッドについて

まずインスタンス変数topの値をfront変数に代入。Front.nextをrearに代入。Frontの次が存在する間でwhileループを回し、中で引数numとrear.numが一致するか判定する。一致した場合直ちにループから抜け、front.nextをrear.nextに変更し、rearの値をNoneにする。

一致しなかった場合はfrontにrear、rearにrear.nextを代入し、frontとrearを次の値に更新。更新したあとにrearがNoneとなった場合はメソッドの処理を終了する。

メインの処理について

LinkedListのインスタンスlを生成。Zac\_Japan.txtを開き、1行ずつデータを取得し、insertメソッドを使って線形リストを作成していく。次に削除したい選手の番号を入力するnum変数を用意し、whileを使って0が入力されるまでnumを引数としてdeleteメソッドを実行する。

# (5) 考察

一番奥のデータ、手前のデータ、すべてのデータを削除できるように、frontとその次のノードを示すrearという２つの変数を準備して処理した。FrontにはLinkedListのインスタンスを生成したときに自動で作られるList(None,None,None)インスタンスが初期値として代入されているので、初期値の次にあたる手前のデータの削除が可能。奥のデータの削除については、front.nextであるrearがNoneで無い限りでループを回しているので、可能になっている。手前と奥のデータを削除できれば、同じ要領の繰り返しですべてのデータの削除もできる。

今回、一度Listクラスだけをつくって、機能については一つずつ関数を作成する方針で実装していた。しかし、この方法だと線形リストの生成と表示がうまくできなかったので、リストの各データを示すクラスとは別に、リストの先頭を指す変数を生成し、リストの操作ができるクラスが新たに作った。こうすることで、シンプルなコードで機能の実装が可能となったと考えられる。