

提出日 令和2年06月19日

アルゴリズムとデータ構造 第3回課題レポート

学籍番号 (A19117)

氏名 (永尾優磨)

【課題】

- NodeListインスタンスlistを作れ
- listに文字列"AAA", "BBB", "CCC", "DDD", "EEE", "FFF"を挿入せよ。
- 確認のため、リスト内のすべてのデータを表示せよ。
- データ内に文字列"CCC"があるかどうかを検索をかけよ。
- 検索結果が正しいかどうかを確認するため、検索に引っかかったノードのデータ部を表示させよ。
- 先頭のノード("FFF"のノード)を削除せよ。
- 確認のため、リスト内のすべてのノードを表示させよ。
- 残りの5つのノードも削除せよ。ただし、ループを使って効率的に行うこと。
- 確認のため、リスト内のすべてのデータを表示させよ。(何も表示されないはず)

【作成したプログラム】

```
public class NodeTest {  
  
    public static void main(String[] args) {  
        NodeTest tes = new NodeTest();  
        NodeList list = tes.new NodeList();  
        list.insertFirst("AAA");  
        list.insertFirst("BBB");  
        list.insertFirst("CCC");  
        list.insertFirst("DDD");  
        list.insertFirst("EEE");  
        list.insertFirst("FFF");  
        list.displayNodeList();  
        System.out.println(list.find("CCC"));  
        list.deleteFirst();  
        list.displayNodeList();  
        for (int z = 0; z < 5; z += 1) {  
            list.deleteFirst();  
        }  
        list.displayNodeList();  
    }  
  
    class Node {  
        // 属性  
        String data;
```

```

Node next;

// メソッド
// コンストラクタ
Node(String s) {
    data = s;
    next = null;
}

// ノードを表示するメソッド
public void displayNode() {
    System.out.print("{ " + data + " }");
}

}

class NodeList {
    Node first;

    NodeList() {
        first = null;
    }

    public boolean isEmpty() {
        if (first == null) {
            return true;
        } else {
            return false;
        }
    }

    public void insertFirst(String s) {
        Node newNode = new Node(s);
        newNode.next = first;
        first = newNode;
    }

    public Node deleteFirst() {
        Node temp = first;
        first = first.next;
        return temp;
    }

    public void displayNodeList() {
        System.out.print("リスト上のデータ -> ");
        Node current = first;
        while (current != null) {
            current.displayNode();

```

```

        current = current.next;
    }
    System.out.println();
}

public Node find(String key) {
    Node current = first;
    while (!current.data.equals(key)) {
        if (current.next == null) {
            return null;
        } else {
            current = current.next;
        }
    }
    return current;
}
}
}

```

プログラム1 最初に書いたプログラム

```

public class NodeTest2 {

    public static void main(String[] args) {
        NodeTest2 tes = new NodeTest2();
        NodeList list = tes.new NodeList();
        String a = "AAA";
        String b = "BBB";
        String c = "CCC";
        String d = "DDD";
        String e = "EEE";
        String f = "FFF";

        list.insertFirst("{}".format(a));
        list.insertFirst("{}".format(b));
        list.insertFirst("{}".format(c));
        list.insertFirst("{}".format(d));
        list.insertFirst("{}".format(e));
        list.insertFirst("{}".format(f));
        list.displayNodeList();
        System.out.println(list.find("{}".format(c)));
        list.deleteFirst();
        list.displayNodeList();
        for (int z = 0; z < 5; z += 1) {
            list.deleteFirst();
        }
        list.displayNodeList();
    }
}

```

```
}
以下省略(プログラム1と同じものが続くため)
```

プログラム2 最終的なプログラム

【プログラムの解説】

NodeTestの中に作成したNodeListを使えるように宣言した後に、NodeList内に記述したinsertFirstメソッドを使い、AAAなどの文字列を代入出来るようにした。

また、displayNodeListの最後をflushでは無く、printlnにした。そのため、改行が毎回追加されて見やすくなっている。

また、プログラム2では、冗長であったAAAなどの文字列を変数化したことにより、追加する処理や削除する処理が増えたときに簡単に記述量を減らせるようにした。

【結果】

```
IJava 575429d3\bin 第03回線形探索.NodeTest
リスト上のデータ -> {FFF}{EEE}{DDD}{CCC}{BBB}{AAA}
第03回線形探索.NodeTest$Node@6a5fc7f7
リスト上のデータ -> {EEE}{DDD}{CCC}{BBB}{AAA}
リスト上のデータ ->
```

図1 プログラムの出力結果

【考察】

最初は、書かれている通りに記述したと考えていたため、何が原因でプログラムを実行出来ないのかが不思議だった。参考文献2を参照した結果、最初に書いていたコードだと、class内の関数にアクセス出来ないことが判明した。そこで、`NodeTest tes = new NodeTest();`追記した結果、class内の関数にアクセスが出来た。そのため、プログラムを出力することが出来た。このことから、VSCodeに書かれているエラーを、そのままChromeで検索することで、問題を解決しようと考えた。

今回起こったエラーの内容は、Javaの文法をしっかりと理解していなかったことが根本の原因である。そのため、Javaの文法を理解するための5日で出来るJavaなどの入門書を読んでおこうと考えた。

【参考文献】

1. 「アルゴリズムとデータ解析の授業スライド、線形探索について」
2. 「No enclosing instance of type Hoge is accessible.」 <<https://qiita.com/watanabk/items/738988fac29e1e1d8d88>>
3. 増井敏克(2020)「Pythonで始めるアルゴリズム入門」翔泳社
4. 「参考文献の書き方」 <http://www7a.biglobe.ne.jp/nifongo/ron/ron_04.html>