**日報**

2023年08月18日

氏名：小松大二

|  |
| --- |
| ■目標  ・Problem13の問題をすすめる |
| ■作業内容  ・Problem13の問題をすすめる  ＊SideWrapperクラスにおいて最も長い文字数に合わせてラッピングする  →　完了  ＊AllWrapperクラスを作成し、入力された文字列の上下左右がラッピングされるようにする  →完了  ・Problem14に取り掛かる  ＊ファイルが開けず未着手  ・Problem07  ＊if文を使わずに全ての行を揃えることに挑戦　→失敗 |
| * 習得したこと   ・「最大文字数を取得するための処理」と「最大文字数に合わせて空白を追加する処理」が同じループ内に書かれていたせいで全ての行を揃えることができなかったことを理解する。そこから、最大文字数の取得を行う処理を、ループを分けて最初に書き、最大文字数を取得してからラッピング文字を設定することで、全ての行を揃える処理を書くことができる  ・Problem13内に書かれている StringWrapper bottom = new BottomWrapper();がクラスのインスタンスを生成して処理を行っていることを理解し、それと同じようにAllWrapperクラス内でSideWrapperクラスなどの各クラスのインスタンスを生成することで、文字列が上下左右にラッピングされ、かつ全ての行が揃っている状態のものを表示することができる |
| * 感想   今日は昨日の続きで、Problem13のSideWrapperの問題から取り掛かりました。  昨日記述したクラスで望んだ結果が得られなかったのは、恐らく、各行の最も長い文字数を取得する処理を書く位置が違ったことが原因だと思われます。昨日の処理だと、最も長い文字数を取得してラッピング文字を設定する前に、ラッピング文字を設定してしまっているため、昨日のように全ての行が揃わず表示されてしまったのだと思います。実際、この考えのもと、最も長い文字数を取得する処理を、空白を追加する処理と同じループ内に書かず、ループを分けて最初に書いたことで、望んだ結果を得ることができました。この結果に至るまでの考え方は間違っているかもしれませんが、佐藤さんにコメントしていただいたやり方と同じやり方を自分で実装して、動かすことができたのでとても嬉しかったです。  次にAllWrapperクラスを作成する問題に取り組みました。これは今までSideやTopで分けていた機能を一つにまとめたクラスなので、まずは単純に、各クラスに書いてあった処理をそのまま書くところから始めました。書いたものを実行してみた結果、文字列が余分に表示されたりラッピング文字がずれて表示されてしまったりしてしまい、望んだ結果は得られませんでした。  AllWrapperクラスがうまく呼び出されていないのか、それとも処理の書き方が間違っているのかもしれないなどを考えながらProblem13のプログラムを見ていると、 StringWrapper bottom = new BottomWrapper(); の文に目がとまりました。問題を解く前からすでに書いてあった文章なので、今まで特に意識していませんでしたが、この文はクラスのインスタンスを生成する文であることに改めて気付きました。クラスのインスタンスを生成することで、クラス内に定義されている処理が利用可能になり、wrapメソッドを呼び出しラッピング処理を行っているのならば、Problem13に書いてあるものと同じものをAllWrapperクラスに書けばいいのではないかと考えました。その考えのもと、Problem13内に書かれていたものを参考に、TopWrapperクラスなどの各クラスのインスタンスを生成する文をAllWrapperクラス内に書いたところ、無事に全ての文字列がラッピングされ、かつ全ての行が揃った状態のものを表示させることができました。今まで意識せずに使ってはいたのですが、今回のように、自分でクラスのインスタンスを生成し処理を行う文は書いたことがなかったので、AllWrapperクラスが動いたのを見て感動しました。今回でクラスのインスタンスを生成することへの理解が深まったので嬉しかったです。  続いてProblem14に取り掛かろうとしたのですが、課題を解くのに必要であろうファイルが見つけられなかったので、久しぶりにProblem07に取り組みました。Problem13にて、最大文字数の取得処理について書いてきて、今までよりは理解が深まったと思うので、サーモンに特定の操作をしなくても行を揃えることに挑戦してみました。Problem13のとき、最大文字数を取得する処理を書く位置が間違っているせいで、うまく動かなかったケースがあったので、それと同じことがProblem07でも起きているのではないかと考え、処理を書く位置を変えてみました。しかし、結果は変わらず行を揃えることはできませんでした。まずそもそも、自分で最大文字を取得する処理を書く前から、うにの行がほかの行と揃っていたので、他のところになにかあるのかもしれないと考え、Sushi.javaを見てみたり、Itamae.javaをまた読んだりしたのですが、Sytem.out.printlnで空白を追加していることぐらいしか、行に関する操作をしているところは見つけられませんでした。今日みたいに、急遽、先の問題に取り組めなくなった時などに、また挑戦してみたいと思います。 |