**日報**

2023年08月16日

氏名：小松大二

|  |
| --- |
| ■目標  ・Problem12に取り組む  ＊carクラスを実装しレースに参加させる  ・Problem13に取り掛かる |
| ■作業内容  ・Problem12に取り組む  ＊carクラスを実装しレースに参加させる  →完了  ・Problem13に取り掛かる  ＊ TopWrapperクラスを作成し、入力文字列の上部をラッピングするクラスを作成する。  そしてそれを、 oneLineWrap()/multiLineWrap()に組み込み動かす  →完了 |
| * 習得したこと   ・BottomWrapperクラスのfor内で、各行の最も長い文字数をもとにラッピング文字数を決める処理と、この処理がTopWrapperクラスにおいても必要であることを理解して、TopWrapperクラス内に実装することができる  ・ System.lineSeparator()で改行ができることを理解する  ・retBuilderがラッピング後の文字列を構築するためのものであることを理解し、maxSizeが更新されればラッピング文字がretBuilderに追加されていく処理を書くことができる。また、retBuilderに元の文字列と改行を追加することで、文字列の上にラッピング文字を表示させる形を実装することができる |
| * 感想   今日はProblem12のcarクラスを実装しレースに参加させる問題から取り掛かりました。前回の日報での回答で、気にしなくていいと頂いたので、新しいボートをレースに参加させたときと同様の方法で、車をレースに参加させることができました。  次にProblem13に取り組みました。TopWrapperクラスを作成する問題で、StringWrapperインターフェースを実装してTopWrapperクラスを作成することは、今までの問題でやってきたことと同じなので苦戦せずにできたのですが、その中身をどう書いたらいいか分からずとても苦戦しました。文字列の上部にラッピング文字を追加したいので、それとは逆の、下部にラッピング文字を追加している BottomWrapperクラスを参考にすれば書けるのではないかと考え作業を進めました。文字列の上部にラッピング文字を追加するには、文字列の文字数に合わせてラッピング文字が追加されるようにしなければならないので、SideWrapperクラスにはなくてBottomWrapperクラスにはある、最も長い文字数を取得する処理が必要であると考え、この処理をTopWrapperクラスに追加しました。これにより、BottomWrapperクラスと同様に、最も長い文字数をもとにラッピング文字数を決め、表示させる処理は書けたのですが、文字列の上にこれを表示する方法が中々思いつかず、また苦戦しました。しかし、考えてる中で、retBuilderに着目して、retBuilderにラッピング文字が追加されているのならば、ここに改行を追加し、最後に元の文字列を追加すれば、文字列の上にラッピング文字が書かれている形を表示できるのではないかと思いつきました。実装してみた結果、無事に文字列の上下左右がラッピングされている状態を確認できたので嬉しかったです。  前回の日報でどんなif文を書いたかを教えてほしいということでしたので記述させていただきます。1週間ほど前の記憶ですので定かではありませんが、isRace=falseなどを書いてみた記憶があります。そしてそれを書いてみた後で、これは書いても書かなくても結果は一緒だなと気づき、結局if文を使ってjudgeメソッドが動かないようにするやり方は分からなかったのでreturnを使いました。 |