

「オブジェクト指向プログラミング特論」

2016 年度期末レポート課題

締切:2016/8/4

1 期末レポートについて

期末レポート課題は、以下の通りです。期日 17 時までに、理工学部 7 号館 2 階のレポートボックスへ提出すること。

- Word や \LaTeX などを利用して電子的に組版し、A4 用紙に印刷して提出すること。
- レポート課題とその解法をわかりやすく説明すること
- 動作例を作成すること
- 単に、プログラムだけ提出した場合には、不合格とする
- プログラムでは、わかりやすい変数名とメソッド名を用いるとともに、適切にコメントを付し、可読性を高めること。
- 正しい日本語で作成すること。
- 他人のレポートを写したと判断した場合には、不合格とする。
- 書籍や Web ページ等を参考としている場合には、必ず出典を明示すること。

2 課題

2.1 Dijkstra 法

頂点の集合 V とそれらを結ぶ弧（向きがある）の集合 A によって定義されるグラフ $G = (V, A)$ を考える。各弧 $a \in A$ に長さ $l(a) > 0$ が定義されているとき、指定した始点 v_0 から各頂点までの最短距離を求めるアルゴリズムに Dijkstra 法がある。アルゴリズムは

- $U \subseteq V$: 始点からの経路を一旦は求めたが距離が確定していない頂点の集合
- $W \subseteq V$: 始点からの距離が確定した頂点の集合

- $p: V \rightarrow R$: 各頂点の始点からの距離を表す写像

を用いて、Algorithm 2.1 のようになる。最初に初期値を以下のように定める。

$$\begin{aligned}
 U &= \{v_0\} \\
 W &= \emptyset \\
 p(v_0) &= 0 \\
 p(v) &= +\infty \quad (\forall v \in V \setminus \{v_0\})
 \end{aligned} \tag{2.1}$$

```

while( $U \neq \emptyset$ ){
   $w = U$  のなかで  $p(w)$  が最小となる頂点  $w$ 
  //  $w$  は  $U$  から取り除くことに注意
  forall ( $a \in \delta^+ w$ ){ //  $w$  を始点とするすべての弧に対して
     $x = \partial^- a$  //  $a$  の終点
    if ( $x \notin W$ ){
      if ( $p(x) > p(w) + l(a)$ ){
         $p(x) \leftarrow p(w) + l(a)$ 
         $U \leftarrow U \cup \{x\}$ 
      }
    }
  }
   $W \leftarrow W \cup \{w\}$ 
}

```

Algorithm 2.1: Dijkstra アルゴリズム

2.2 クラス設計

Dijkstra 法を実装するために、以下に概要を示すクラス設計を行うこと。

- graph パッケージ
 - Node クラス: 頂点のクラス。始点からの距離を保持できる。
 - Arc クラス: 弧のクラス。長さを定数として保持している。
 - Graph クラス: グラフの抽象クラス。頂点のリスト、頂点からそこを始点とする弧への写像を保持している。頂点の登録、頂点と頂点の連結のメソッドを有している。また、グラフを実際に構成する抽象メソッドを有している。具体的なグラフは、このクラスの拡張として定義する。

- dijkstra パッケージ
 - Dijkstra クラス : 指定した Graph に対して、指定した頂点を始点として Dijkstra 法によって、始点から各頂点への最短距離を求める。
- problems パッケージ
 - 具体的なグラフのクラスを定義し、dijkstra 法によって最短経路を探索する。

2.3 実装と確認

上記のクラス設計に従って、グラフ及び Dijkstra 法を実装しなさい。頂点数が 5 程度、10 程度の二つのグラフを定義し、Dijkstra 法が正しく実行できることを示しなさい。

Algorithm 2.1 では、集合 U のうち始点からの距離が最も短いものを選ぶ必要がある。効率よく選ぶには、 U を二分木ヒープとして実装すると良い。この場合、 U に含まれている頂点 x の距離 $p(x)$ を更新する際には、注意すること。

3 レポート採点基準

C: プログラムを作成しているが、十分な説明が行われていない。または、適切にクラス構成が行われていない。

B: クラスが適切に設計され、プログラムに十分な工夫がある。

A: B に加え、クラス設計、処理の流れがレポートで適切に説明されている。

S: A の基準を満たし、特に顕著な工夫や記述がある。