

UML を用いたソフトウェア開発

～マルチタスクのプログラム化とデータ記憶における問題点～

情報システム工学科 3 年 73 番前田唯

1. はじめに

将来私たちが企業に就職して、行うであろうソフトウェア開発。大人数で一つのソフトウェアを開発するにあたり UML は意思疎通を図る為にとっても便利である。今回はその UML を用いてソフトウェアを作成することにより UML の特性を理解し、またソフトウェア開発を実感した。

2. 研究内容

LEGO ロボットを用いて迷路の探索走行と記憶走行を行った。まず、行き止まりが幾つもある迷路をしらみつぶしに進みゴールまでたどり着く。次に、そこからユーターンしてスタート地点まで戻る。ただし、帰りの分岐点では正しい方向に進み、一度も行き止まりに衝突せずにもとの地点に戻るようにする。これらを満たすプログラムを以下の4つの過程より作成した。

- ① UML の製作
- ② タスクを考える
- ③ プログラムの作成
- ④ 実装

3. UML

UML(Unified Modeling Language:統一モデリング言語)とは、ソフトウェアの設計とパターンを表現するための視覚的言語でありソフトウェア開発の用途に最も適している。一般的な用途としては「ソフトウェアの設計」、「ソフトウェアプロセスやビジネスプロセスの伝達」、「既存のシステム、プロセス、組織に関する文章化」などがある。

今回は以下の5つを作成した。

- ① ユースケース図:機能を用件レベルで表現
- ② クラス図:システムの静的な関係を表現
- ③ オブジェクト図:特定の時点でのクラスのインスタンス同士の関係を表現
- ④ シーケンス図:システムの要件の実現方法を動的に表現
- ⑤ ステートマシン図:要素の内部的な状態遷移を表現

4. タスク

システムが行う仕事の単位であり、プログラムの実行そのもののことも言う。今回は図1に示すタスクを考えた。

タスク名(周期/非周期)	優先度	周期[ms]
走行タスク(周期)	10	100
タッチセンサタスク(周期)	5	200
光センサタスク(非周期)	15	—

図 1 システムに必要なタスク

5. 考察と反省

最初に自ら設定した課題に対して要求を満たすソフトウェアを開発することができた。しかし、汎用性が小さく他者にわかりにくいプログラムとなってしまった。

当初はゴールから戻ってくるときに使用する分岐データの管理をポインタのリスト構造を用いて行おうと試みたがうまくデータが入力できてなかった。その理由として周期タスクは呼び出しごとにデータ領域に値が再設定されることが考えられる。また、光センサの間隔は迷路の仕様の関係上、仕様書よりも広くした。また、今回作成したソフトウェアは分岐が2方向のみ実現でき、3 方向以上の迷路に対応させるためには更に複雑なアルゴリズムを考える必要がある。

今回は簡単なソフトウェアの開発なのでプログラムも数百行にまとまり、UML も何とか作成できたが、今後さらに大規模なソフトウェアを開発するためには UML についてさらに学んで理解を深める必要があると思った。

6. 参考文献

- [1] Dan Pilone, Neil Pitman 著「UML 2.0 クイックリファレンツ」オライリージャパン
- [2] 前川守 著「オペレーティングシステム」岩波書店
- [3] 清水 兼多郎 著「オペレーティングシステム」岩波書店
- [4] 矢向研究室ホームページ
<http://www.comp.sd.keio.ac.jp/~yamaoka/research/research.html.ja.jis>