

Squeak eToys による待ち行列シミュレーションプログラムの作成

相澤正人, 中平勝子, 福村好美

長岡技術科学大学 〒 940-2188 新潟県長岡市上富岡町 1603 番地の 1

E-mail : {aizawa, katsuko, fukumura}@oberon.nagaokaut.ac.jp

概要

Squeak eToys はビジュアルなプログラミング言語であり, 小学生の情報教育, ゲームの作成などに利用されている. 本稿では, シミュレーションにおける Squeak eToys の利用可能性について評価することを目的として, 待ち行列のシミュレーションプログラムに応用した結果を報告する. 対象とする待ち行列はランダム到着, ランダムサービス時間で, Squeak eToys の特徴である「タイル」のみを使用したプログラミングを試みた.

A Simulation Program for Queuing Model by Using Squeak eToys

Masato Aizawa, Katsuko T.Nakahira, Yoshimi Fukumura

Nagaoka University of Technology

1603-1 Kamitomioka-machi, Nagaoka-shi Niigata 940-2188 Japan

Abstract

Squeak eToys is a visual programming language and it is used for information education in elementary school and games. This paper shows the results of simulation program for the queuing model by using Squeak eToys. The queuing model is random arrival interval time and random service time, and the program is constructed from only instruction tiles that is a feature of Squeak eToys.

1 はじめに

情報社会において, e ラーニングの導入など情報化が進んでいる. 高校でも ICT (Information and Communications Technology: 情報通信技術) 基礎力向上のため, 平成 15 年度より教科「情報」が導入されており, 習得目標や内容に応じて「情報 A」「情報 B」「情報 C」の 3 種類に分類されている.

文部科学省では, 高等学校学習指導要領において, 「情報 B」の習得目標を「コンピュータにおける情報の表し方や処理の仕組み, 情報社会を支

える情報技術の役割や影響を理解させ, 問題解決においてコンピュータを効果的に活用するための科学的な考え方や方法を習得させる。」と掲げている [1]. さらに, 具体的なカリキュラムとして「問題のモデル化とコンピュータを活用した解決」が設けられており, その一つとして「モデル化とシミュレーション」を取り上げ, 「身のまわりの現象や社会現象などを通して, モデル化とシミュレーションの考え方や方法を理解させ, 実際の問題解決に活用できるようにする。」という習得目標を掲げている. 我々の身の回りには様々な現象 (物理現象・社会現象) が存在するが, それらの

中には、実際に試すことが難しいものや、偶然に左右されるために確定的な結果とならないものがある。このような場合、シミュレーションなどによって模擬実験を行うことにより、問題を解決する方法が有効である。

本稿では、シミュレーションによって問題解決を図ることが可能な事例の例として「待ち行列」を取り上げた。「待ち行列」は、スーパーのレジや駅の窓口などで見られる現象であり「待ち時間の長さ」が問題になることが多い。この「待ち行列」をモデル化し、「待ち時間の長さ」を短くするという問題解決をシミュレーションによって模擬実験することは「情報 B」の習得目標達成に貢献できる。さらに発展させることで、通信ネットワークのサーバにおけるパケット処理最適化にも応用が可能となる [2]。

「待ち行列」のシミュレーションプログラムに関しては、Java や VisualBasic などのプログラミング言語で作成した例がある [3,4]。しかし、これらのプログラミング言語では、「待ち行列」のような複雑なプログラム作成を行う以前に、プログラムの記述方法やアルゴリズムの考え方などを十分勉強する必要があるため、初心者がいきなりプログラム作成に取り組むことは難しい。また、前述した作成例では、結果を視覚的に表示しているものの、客が動いて窓口へ到着するビジュアルな様子を表現できていない。

このため「待ち行列」を、タイルスク립ティングシステムを採用した Squeak eToys というプログラミング言語を用いてビジュアルに表現できるシミュレーションプログラムを作成した。他のプログラミング言語のように複雑なプログラムコードを作成することなく、Squeak eToys は、タイルスク립ティング機能により、プログラミング未経験者であっても比較的容易にプログラムを作成できる。さらに、画面上の部品（部品のことを Squeak eToys ではモーフと呼ぶ）を容易に動かすことができるため、ビジュアルな表現を行うプログラム作成に向いている。

以下、第 2 章では Squeak eToys の概要を述べ、第 3 章ではプログラムの設計内容について述べる。第 4 章では、プログラムの画面構成と処理

内容について詳細に述べる。第 5 章は、プログラム全体の考察を行い、最後に第 6 章は、まとめを示す。

2 Squeak eToys の概要

プログラム作成には、日本語版の Squeak eToys (バージョン 3.7.1) を用いた [5]。Squeak eToys は、Smalltalk というプログラミング言語をベースに作られた Squeak を発展させたものである。以下に Squeak eToys の主な特徴を挙げる [6,7]。

1. 通常のプログラム言語が主としてソースコードを記述してプログラミングしていくのに対し、Squeak eToys では「タイル」と呼ばれる命令のパーツをマウスで配置することによってプログラミングを行う。そのため、未経験者であってもプログラム内容が比較的簡単にプログラミングできるといわれている。
2. 「タイル」を用いてのプログラミング環境であるタイルスク립ティング・システムにより、マウスのみを用いた簡単な操作でプログラミングできる。そのため、小学生などの低年齢層ユーザでも利用しやすい。
3. オブジェクト指向言語である「Squeak」がベースとなっており、タイルを用いたプログラミングだけでなく、ソースコードを記述してプログラミングを行うことも出来る。そのため、初心者から上級者まで広く利用可能である。
4. オープンソースであり、仕様が公開されている。また、フリーソフトでもあるため、無料で使用できる。Windows, Mac, UNIX 等の主要な OS にも対応しているため利用しやすい。Web ブラウザへのプラグインも用意されている。

3 プログラム設計

以下では、「待ち行列」シミュレーションプログラムの設計内容を述べる。

3.1 待ち行列モデル

本稿では、まず簡単な待ち行列のモデルを例にとってプログラムを作成した [8]。待ち行列の主な流れは (1) 客が窓口へ到着する (2) 窓口で何らかのサービスを受ける (3) サービスが終わると客は退去する、である。窓口の数や客の到着分布などは様々なパターンが考えられるが、本稿は、客の到着間隔および窓口のサービス時間はランダム、窓口の数は1つとした。

3.2 プログラムのフローチャート

「待ち行列」のシミュレーションを行う際には「客」と「窓口」という2つの主要なモーフが必要である。本プログラムでは、「客」モーフを発生・動作させるために、図1のフローチャートに沿って処理内容（以下、処理内容のことをメソッドと呼ぶ）を作成した。図1に新たな客が発生するまでのフローチャートを示す。

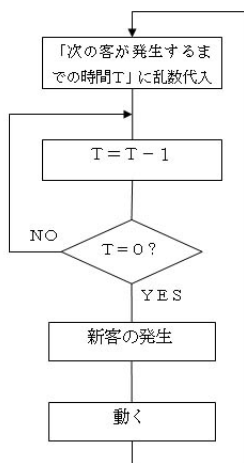


図1：新たな客が発生するまでのフローチャート

Squeak eToys では、各メソッドは基本的に繰り返し処理で実行される。そのため、繰り返し処理の制御がプログラムの作成上重要となる。Squeak

eToys では、繰り返し処理の制御に「チクタク」、「ノーマル」、「ポーズ」などの機能を利用する。プログラム開始から終了まで常に繰り返し処理を行う命令「チクタク」、他のメソッドから処理命令が来たときのみ処理を行う命令「ノーマル」、メソッド処理を一時停止する命令「ポーズ」を適切に与えることにより、プログラムを制御する。

本プログラムでは、後述する「次の客が発生するまで」タイマーの「時間をカウント」メソッドに「チクタク」命令を設定し、それ以外のメソッドには「ノーマル」命令を設定した。これは、「時間をカウント」メソッドが、新たな客を発生させる「新客の発生」メソッドへ「客発生」の命令を送るために、常に動いている必要があるからである。

プログラムを開始すると「次の客が発生するまで」タイマーの「時間をカウント」メソッドが処理を開始する。前述したように、この処理は「チクタク」命令で実行される。「次の客が発生するまでの時間」が0になったら、「時間をカウント」メソッドは「新客の発生」メソッドへ「客」モーフを発生させるよう命令を出す。そして、「時間をカウント」メソッドは自身に乱数を代入し、新たな「客」モーフ発生に向けて処理を開始する。



図2：「動く」メソッドのフローチャート

「新客の発生」メソッドにより発生した「客」モーフは「動く」という処理を開始し、窓口へ向かって動く。客は窓口へ到着すると、自身の動きを止めるために「動く」メソッドを「ポーズ」し、「窓口処理へ乱数代入」、「窓口処理の時間をカウ

ント」、「消える」という3つのメソッドをスタートする。「窓口処理へ乱数代入」メソッドでは、窓口で行う処理時間を乱数で代入する。次に、「窓口処理の時間をカウント」メソッドで、窓口での処理時間をカウントさせる処理を開始する。そして「消える」メソッドでは「窓口処理の時間」の値を参照し、時間が0、すなわち処理が終了したら「客モーフ」自身を消すという命令を出す。これらのメソッドは各々の「客」モーフが持っているため、客は自律的に行動できる。図2に「客」モーフが持っている「動く」という処理内容のフローチャートを示す。

4 プログラム構成

以下では「待ち行列」シミュレーションプログラムの画面構成と、各モーフ内に含まれるメソッドについて述べる。

4.1 画面構成

図3にプログラムの実行画面と各モーフの構成を示す。

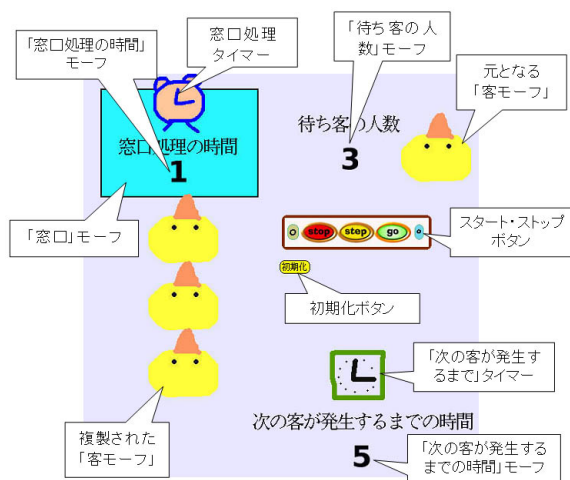


図3：プログラムの実行画面と各モーフの構成

「スタート・ストップボタン」がプログラム開始・停止の制御ボタンであり、これは Squeak eToys に内蔵されているモーフを利用した「スタート」に相当する「GO」ボタンをクリックすると、プログラムが開始される。客は「次の客が発生する

までの時間」が0になると指定された座標上に発生し、窓口へ向かって進む。窓口の前に客が並んでいた場合は動きを止めて行列を作る。並んでいる客の数は「待ち客の人数」に表示される。

客が窓口に着くと「窓口処理の時間」に乱数が代入され、その時間の間だけ窓口処理が行われる。「窓口処理の時間」が0になると客は自動的に消滅する。

4.2 「客」モーフ

客は、窓口での処理の対象であり、現実世界では、レジへやってくる「客」や、ネットワーク上を流れる「パケット」などが客に相当する。一つ一つの客は自律的に発生し行動することが要求される。発生後は窓口へ進み、先客がいたら待ち行列を作る。また、窓口での処理が終了したら客は消える。

4.2.1 「新客の発生」メソッド

本プログラムでは、客は時間が来たら自動的に発生し、自律的に行動することが要求される。そのためには、同じ性質を持つ「客」モーフを複数発生させる必要がある。Squeak では、同じモーフを複数発生させる場合、元となるモーフをコピーして複製を作るという手法が使われる。本プログラムでは同様に、元となる客モーフをコピーして新たな客を発生させている。この時、単純なコピーでは、特定のモーフに対して値を与えたり命令を出したりすることができない。

この問題を解決するために、本プログラムでは「プレイヤー型の変数」を利用した。プログラミングを行う際には、値を保存するために「変数」を作成することがある。その際、変数に代入する値がどのようなものが識別させるために、変数に対して「整数型」や「文字列型」などの型を与える。Squeak eToys には「プレイヤー型」の変数があり、プレイヤー型に指定した変数にモーフのコピーを代入することで、特定のモーフに対して値や命令を与えることができる。「新客の発生」メソッドでは、「新客」という名前のプレイヤー変数に「客」モー

フのコピーを代入している．図 4 に，プレイヤー変数指定時の画面を示す．

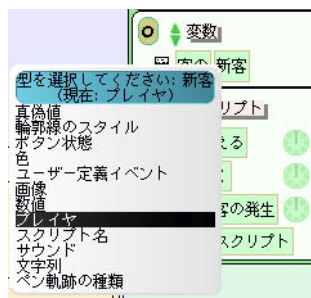


図 4：プレイヤー変数指定時の画面

「新客」変数に代入した「客」モーフに対して，発生場所の x 座標と y 座標を与え，後述する「動く」という処理をスタートさせる．また，「待ち客の人数」に 1 を加える処理も行っている．メソッドの内容を図 5 に示す．

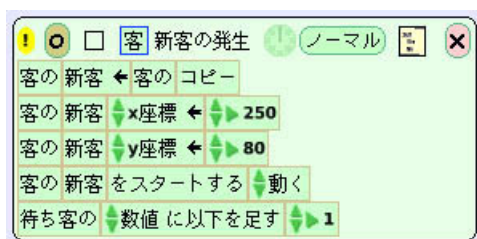


図 5：「新客の発生」メソッド

4.2.2 「動く」メソッド

「動く」メソッドでは，客が動いたり止まるときの処理を記述している．発生した客はまず，並んでいる客がいるかどうか（自分の鼻のオレンジ色が前の客の黄色い部分に触れているかどうか）という判断をし，触れていない場合に動くようにしている．こうすることにより，前の客が進んだときに新たに命令を出さなくても自動的に進むことが出来る．

次に，客が窓口に着いたと判断した（自分の鼻のオレンジ色が窓口の水色に触れた）時には，まず客の動きを止めるために「動く」メソッドを「ポーズ」する．そして，窓口処理へ乱数を代入し，後述する「窓口処理の時間をカウント」メソッドと「消える」メソッドをスタートさせている．メソッドの内容を図 6 に示す．



図 6：「動く」メソッド

4.2.3 「消える」メソッド

「消える」メソッドでは，窓口処理が終了したあとに客を消す処理を記述している．客は，窓口処理の時間が 0，すなわち処理時間が終了したと判断すると，自分自身を消去し，さらに「待ち客の人数」から 1 を引くという処理を行う．メソッドの内容を図 7 に示す．

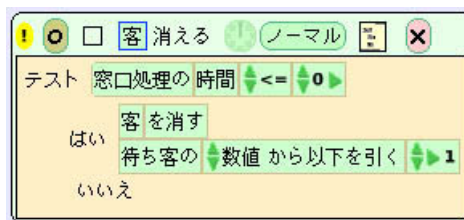


図 7：「消える」メソッド

4.3 「窓口」モーフ

窓口は，客がやってきて処理を受ける場所であり，現実世界では，買い物精算を受ける「レジ」や，ネットワーク機器の「ルータ」などが窓口に対応する．窓口では，客に対してある一定の時間，何らかの処理（サービス）を行う．サービス時間は客によって異なる．

4.4 「窓口処理」タイマー

「窓口処理」タイマーでは，客が窓口で受けるサービス時間を計測する．本プログラムでは，窓口でのサービス時間はランダムであるので，サービス時間決定のために Squeak eToys に内蔵され

ている乱数発生モーフを用いた。

4.4.1 「窓口処理へ乱数代入」メソッド

「窓口処理へ乱数代入」メソッドでは、窓口処理の「時間」という変数へ乱数を代入する処理を行っている。メソッドの内容を図8に示す。

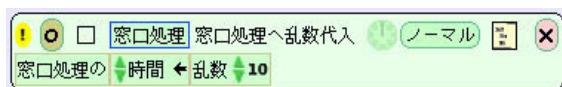


図8:「窓口処理へ乱数代入」メソッド

4.4.2 「窓口処理の時間をカウント」メソッド

「窓口処理の時間をカウント」メソッドでは、窓口での処理時間をカウントする処理が行われている。具体的には、窓口処理の時間が0になるまで、1ずつ減算するというものである。メソッドの内容を図9に示す。



図9:「窓口処理の時間をカウント」メソッド

4.5 「次の客が発生するまで」タイマー

「次の客が発生するまで」タイマーでは、新たな客が発生するまでの時間を計測する。本プログラムでは、客の到着時間はランダムであるので、発生時間決定のために Squeak eToys に内蔵されている乱数発生モーフを用いた。

4.5.1 「乱数を代入」メソッド

「乱数を代入」メソッドでは、次の客が発生するまでの「時間」という変数に乱数を代入する処理を行っている。メソッドの内容を図10に示す。



図10:「乱数を代入」メソッド

4.5.2 「時間をカウント」メソッド

「時間をカウント」メソッドでは、次の客が発生するまでの時間をカウントし、さらに時間が0、すなわち客が発生する時間になったら、「新客の発生」という処理を行っている。メソッドの内容を図11に示す。

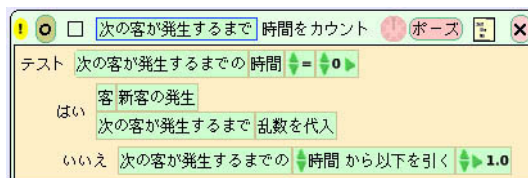


図11:「時間をカウント」メソッド

4.6 「待ち客の人数」モーフ

窓口と並んで待っている客の人数を表示するモーフ。「新客の発生」メソッドと「消える」メソッドが、このモーフに対して加算・減算を行い、数値で表示する。

4.7 「窓口処理の時間」モーフ

「窓口処理」タイマーの数値を表示するモーフ。「窓口処理」タイマーの数値を参照して表示する。

4.8 「次の客が発生するまでの時間」モーフ

「次の客が発生するまで」タイマーの数値を表示するモーフ。「次の客が発生するまで」タイマーの数値を参照して表示する。

4.9 「初期化」ボタン

本プログラムでは、表示されている数字（待ち客の人数など）や客の位置などを初期化する「初期化」ボタンを設けた。具体的な処理として、ボタンが押された時に「次の客が発生するまでの時間」、「窓口処理の時間」、「待ち客の数値」、「窓口処理の時間表示」、「新客発生までの時間表示」を初期状態にする。メソッドの内容を図12に示す。



図 12 : 「初期化する」メソッド

5 プログラム全体の考察

本稿では「待ち行列」シミュレーションプログラムの作成にあたり、タイルスク립ティング機能を持つ Squeak eToys を使用した。

「待ち行列」は様々な種類が存在するため、待ち行列をシミュレーションする際には、客の到着分布や窓口数などを変更する機能が必要である。また、待ち時間の計算などの演算処理は、タイルを用いたプログラミングでは難しい点があるため、より複雑なプログラムを作成する際には、コードを記述してのスクリプト環境の利用や、Squeak eToys 以外のプログラミング言語を用いる必要がある。

今回は「待ち行列」の概念を理解させるための学習教材としての利用を想定した。このような教材を作成するという観点からは、タイルを用いたプログラミングのみで実現が可能である。また、前述したように、全くの初心者であっても比較的簡単に本プログラムを作成できたことから、指導者側が教材として提供するだけでなく、学習者自身でプログラムを作成することも可能となり、授業での活用範囲が広がると考えられる。

6 まとめ

本稿では、高等学校の教科「情報 B」における「モデル化とシミュレーション」の習得目標達成のために、Squeak eToys を用いて「待ち行列」シミュレーションプログラムの作成を行った。Squeak eToys の特徴であるアニメーション機能

を用いて「客」の動きを表現した。結果として、基本的な「待ち行列」シミュレーションプログラムを作成できた。また、作成したプログラムを元に、Squeak eToys の「待ち行列」シミュレーションプログラムへの利用可能性を考察し、複雑な数式処理を必要としないプログラムであれば、標準のタイルのみを用いた環境でもプログラム作成が可能である。

7 謝辞

プログラム開発に当たって技術的なアドバイスを下さった有限会社エム・アイ・エルの五百部敦志氏に、この場を借りて厚く御礼申し上げます。

参考文献

- [1] 文部科学省 高等学校指導要領
http://www.mext.go.jp/b_menu/shuppan/sonota/990301/03122603/011.htm
- [2] 滝根哲哉, 伊藤大雄, 西尾章治郎「ネットワーク設計理論」岩波書店 (2004)
- [3] 瀧花喜彦「待ち行列理論の多角的視野からの考察」多摩大学 (1998)
- [4] 待ち行列シミュレーション
<http://homepage3.nifty.com/jobus/>
- [5] スクイークランド
<http://squeakland.jp/>
- [6] 梅沢真史「自由自在 Squeak プログラミング」ソフト・リサーチ・センター (2004)
- [7] BJ・アレン＝コンノキム・ローズ「子どもの思考力を高める「スクイーク」理数力をみるみるあげる魔法の授業」WAVE 出版 (2005)
- [8] 西田俊夫「待ち行列の理論と応用」朝倉書店 (1976)