

タッチデバイス向け Java エディタ

早川 友啓[†]、小川 大輔[†]、紫合 治[†]

東京電機大学 情報環境学部[†]

1. はじめに

近年ではスマートフォンやタブレット等のタッチパネルを搭載した端末が急速に普及され始めた。また、業務等に情報化が進んでおりプログラミングをする機会が増え、学習する機会も増えてきた。

しかし、プログラミングツールとしてタッチ操作を前提に作られたツールはまだない。よって我々はタッチデバイスで操作する事を前提に文字入力を抑え、タッチ操作を使う java エディタの開発を行った。Scratch を代表とするブロック型エディタに着目し、ブロック型エディタ風に UI を仕上げた。

2. 関連研究

関連する研究の説明を以下に示す。

Scratch[1]とはブロック型言語であり、初学者向けのプログラミング言語である。日本語、英語などで構成されたブロックをつなげてプログラムを組む事ができる。初学者向けとして使われている事が多いが、複雑なプログラムを Scratch で書くこともできる。

また、保井等の研究[2]はブロック型言語をもちいた構造化支援システムである。ブロック型言語で作られたソースコードをテキスト型言語への移行を可能にした。日本語で構成されたブロックで構成されたプログラムを java ソースに変換するシステムのためブロック型とテキスト型を両方見る必要があり煩わしいと我々は考える。

3. システムの目的

本研究のシステムの目的は、タッチデバイスによる java プログラムの作成を支援するツールで、テキスト入力を抑え、文法エラーを無くし初学者に使いやすいツールを提供することである。また、初学者だけでなく java での開発者等の初学者以外の人にも従来の開発環境より使いやすくする事も目的である。インターネット検索ブラウザに見られるタブ機能のようにクラス、メソッドを分け、表示する事でわかりやすくした。

4. システムの概要

4.1 エディタの全体図

エディタの全体図を図 1 に示す。図 1 左側にボタン、左下に作成したクラス、メソッドを表示するコンボボックス、セーブ、ロード、テキスト記述型へ変換するアウトプットボタンがある。

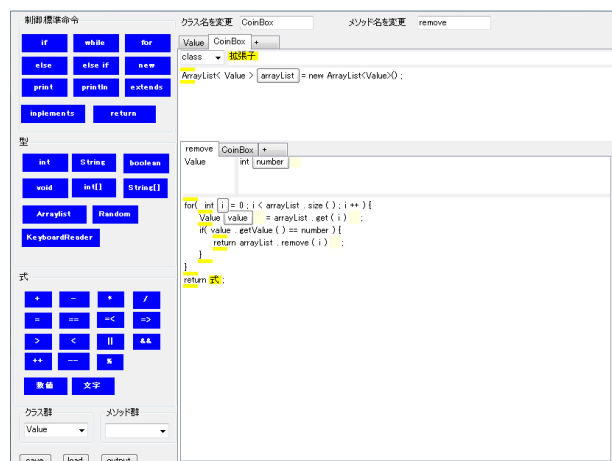


図 1 本エディタの全体図

図 1 右側がソースを作成する場であり、最上部に選択中のクラス名、メソッド名を変更するためのテキストボックスがある。クラス、メソッドはインターネットブラウザのタブのように、クラス名、メソッド名をタッチすれば表示する。

4.2 クラス生成



図 2 クラス生成の流れ

クラス生成にはクラスタブの+をタッチすることでクラス名を入力するテキストボックスが表示され、そこにクラス名を入力する事でクラスが生成される。メソッド生成も同様にできる。生成されたクラスは図 2 下部のように表示される。左上は class か interface の選択ができ、その右の枠内には extends 等の拡張子を入力する。拡張子の下枠内にフィールド宣言を入力する。

4.3 メソッド生成

図 3 の下半分はメソッド生成後の図である。

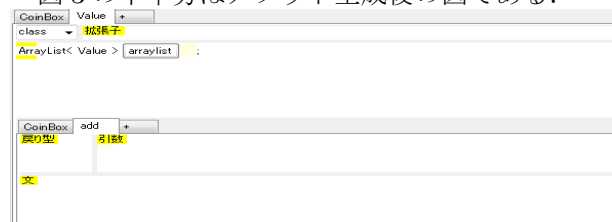


図 3 メソッド生成後のエディタ

メソッド枠の上部の枠に返却値の型と引数を入力する。図3の下部の枠にメソッド本体のプログラムを入力する。

5.プログラムの作成

プログラムの作成では、左側のボタンから使いたい型、制御、命令を選択してタッチした後、それを使うクラスのフィールド宣言や、メソッド本体の黄色い四角で示された場所をタッチする。

5.1 宣言文

左側の int のボタンをタッチし宣言文の黄色い四角をタッチした後の int 型変数宣言の作成を図4に示す。

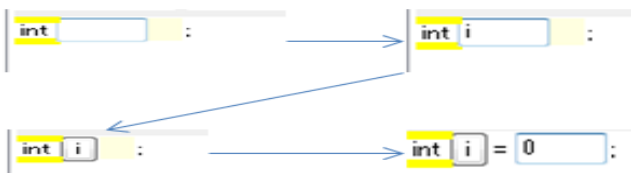


図4 int 宣言の作成方法

宣言文の黄色い四角をタッチすると型名 int と変数名を入力するテキストボックスが表示される。変数名を入力し終わると変数名のボタンが表示され、以後このボタンを使う事が出来る。ボタン”i”と”;”の間の空白をタッチする事で初期値の設定ができる。

5.2 実行文

for ボタンをタッチし、文の黄色い四角をタッチした後の for 文作成を図5に示す。

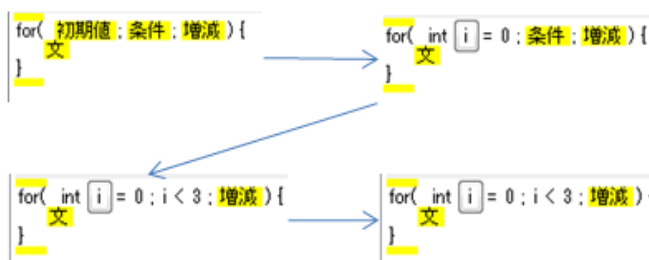


図5 for 文の作成方法

初期値の設定は int 型変数の宣言と同じである。条件を入力するには、不等号のボタンをタッチした後に条件の黄色い四角をタッチすると図6のように不等号の左右に黄色い四角が表示され式を入力できるようになる。ここに for 文内で作成した変数”i”のボタンをタッチし条件の左に入力し、条件の右に数値3を入力すれば図5のようになる。

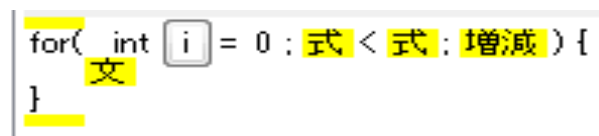


図6 条件に不等号を入れた後の図

増減に i++ と入力するには ++ を入力してから i を入力する。このようにして for 文が完成する。

6.システムの機能

システムの機能として生成したクラス、メソッド

の使用と、クラスのメソッド予測機能がある。生成したクラス、メソッドを使用するには、エディタ内左下にあるクラス群、メソッド群のリストを選択する。クラス群、メソッド群を図7に示す。



図7 クラス群メソッド群

メソッド予測機能を使うには、選択したクラス Value をインスタンス化する。図8にインスタンス化の方法を示す。

```
Value value = new Value();
```

図8 Value をインスタンス化

次にインスタンス化したクラスのボタンを選択し文の黄色い四角に挿入した場合の図を図9に示す。

```
value ;
```

図9 選択したクラスのボタンを文に挿入した図 この時に”value”と”;”の間の空白をタッチするとメソッド予測機能が表示される。この時の図を図10に示す。

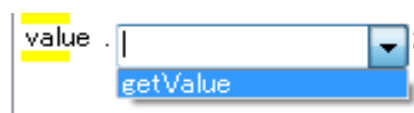


図10 メソッド予測機能

このようにして他クラスのメソッドを楽に入力できる。

7.おわりに

タッチ操作向けの java エディタを開発した。これによってキーボード操作が少なくて済むだけでなく、エラーを含まないプログラムの作成が可能になった。今後の課題として、本システムの有効性を確かめるために学生への実験をする必要がある。加えて、タッチ操作の使いにくい点を改善していく必要がある。さらに、選択したボタンがどこに挿入できるか、挿入場所に対してどのボタンが使えるかを教えてくれるガイド機能の強化を検討したい。

参考文献

[1] MIT メディアラボ Scratch

<https://scratch.mit.edu/>

[2] 保井 元, 松澤 芳昭, 酒井三四郎: ブロックエディタ方式によるプログラミング構造化支援システム, 情報処理学会 コンピュータと教育研究会 vol.2012-CE-113 No.11 (2012)