Proposal of Multi View Programming

Eigo Ito, Takayuki Fujimoto
Dept. of Information System
Toyo University
2100 Kujirai, Kawagoe, Saitama, Japan, 350-8585
e-mail@e-ito.jp, me@fujimotokyo.com

ABSTRACT

Recently, many programming languages are used. There have different features. We should use different languages for purposes. It provides options to our developments. The options make us need to have enough programming skills and knowledge. Programming beginners do not fit present situation. We propose a system for programming learning on smartphone. This system shows blocks and statements by interfaces like smartphones. The best example is multi view scripting. Exist programming languages use some factors. However, each file has only a view. Programming beginners do not learn many at one time. They no need to check algorithm widely. We seem the beginners learn outline of algorithm at first. The beginners should learn flowcharts if the beginners will formal learning. But many beginners want to learn programming for make any application. Proposed system will reduce difficulties to accomplish the aimed application..

KEYWORDS

smartphone; programming; learning; education; Multi View Programming

1 INTRODUCTION

Today, we use many programming languages. The languages are mainly high-level programming languages. These have some artifices for easy scripting. Programming beginners often use sample codes. Programming beginners should learn the artifices at first. However, programming languages have different artifices. The beginners focus to learn the artifices and they cannot spend for learning algorithm enough.

Other problem of programming is a developing environment. Programming languages are made for computers when it is borne. Generally, computers have character user interface and keyboards. The users are accustomed to input characters. In other words, they are used to the scripting environment. This is the biggest deference between the users and today's beginners. The beginners use smartphones than computers. Smartphone do not have keyboards. In addition, Smartphones are based on GUI. Users of smartphones need not to use inputting by keyboards. Then the beginners are used to the scripting.

Developing environments cause other problems. Current computers have high functionality. It contributes pliable uses. However, this feature make lets when we change the environments of the computers. Variety troubles cause in the computers. Programming beginners will be disrupted by them absence knowledge. In addition, completed environments are made for multi uses. It is good for long-term uses, but the beginners have small goals with short-term. Today's hugeness environments hide the goals.

Programming languages have different features. However, the description methods are similar between the languages. Many rules are shared. For example, C uses semicolons as statement terminators. Java and PHP use semicolon for the same ways. Good points should be inherited but it does not be good for programming beginners. Old programming languages have good information on books and web. However,

bad information is on books and web similarly. The beginners cannot pick out information.
We seem to that programming beginners should learn programming with single purpose system. The system focuses on learning and constructing algorithm. Recently, the beginners are used to smartphone than computers. We propose a system for programming learning on smartphones with multi view.

2 MULTI VIEW PROGRAMMING

2.1 User Interface of Smartphones

Smartphones use a touch display with few mechanical buttons. This touch display has enough performances as size, colors and resolutions. Then smartphone are made for graphical uses. It does not have character keys. We can use smartphones without input characters.
The displays of smartphones do not fit text reading. To solve the problem, smartphones have multi view interfaces. Today, many book applications and news applications includes multi view interface. This multi view interfaces probably have a label of hierarchy.

2.2 Multi View Blocking

We seem to that programming beginners should use the accustomed devices. Today, it is smartphones. Smartphone have smaller display than computers. Then smartphones cannot show source code enough.
Many applications on smartphones use multi view. Multi view reduce words form each view. The applications are used multi view for paging. However, programming source code should be looked global image. We propose a system show blocks by multi view. Generally, the most important information of a block is terms. This system shows only the terms. If we want to check the details, we touch the statement and move to next view. This function makes clearly global images.

2.3 Cell Statement

Multi view system must have function of view controller. One way is to put view control buttons. This way shows view action clearly, but it needs to spend spaces for the buttons. On smartphones, display should use spaces efficiently. Other way is to change statements to touchable cells. Programming source code is constructed by statements. These statements have evincive sign for terminating as semicolons. Therefore, we can divide to statements from source codes. For the multi view programming, divided statements form to touchable cells. If users want to check the details of block, they have to touch the cell of statement.

2.4 Closed Scripting

Multi view blocking and cell statement are ways of how show code shape. These artifices are focused on effective space uses. For this purpose, each statement and view should be with concision. When users are allowed open inputting, programming source code has possibilities for the code gets complex. Moreover, open inputting needs the users must have knowledge of the imperatives and syntax. It is hard to learn. Then proposed system has closed inputting function. This function is followed a purpose which is defined at the begging of to start programming.

3 PROTOTYPE SYSTEM

We construct a prototype system on iPhone as smartphone. This system is based on above-cite mechanisms: Multi View Programming. This system has 5 types views:

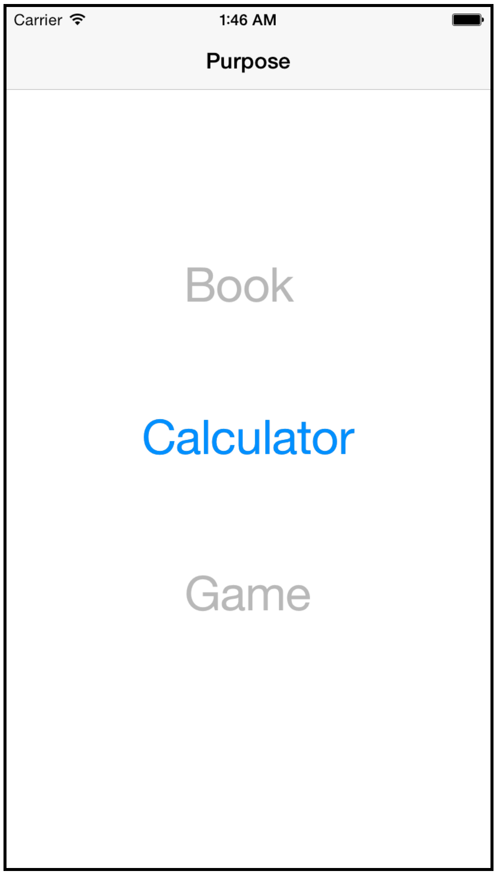
3.1 Project Manager

Project manager view is for management projects. One projects has only one file of programming source code.

3.2 User Interface of Smartphones

Before editing, users must select a category of purpose. Closed statements are followed this selection. Source code let be simple by this selection. On this paper, users are allowed selecting “Calculate.”

Figure 1 Purpose Selector

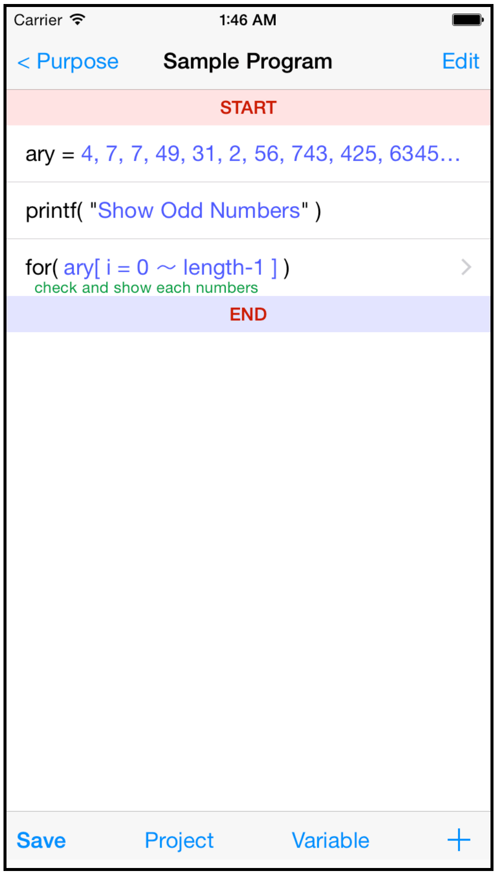


3.3 Source Code Editor

Source code editor view is the most mainly view. This view has cell statements. If users touch a block statement, this system moves to next view. It is source code editor, too. On the other hand, when users touch a non-block statement, this system moves to “Statement Editor.” Figure 2 shows followed code as C language.

```
//Sample Program
int aryNum = 12;
float ary[aryNum] = {4, 7, 7, 49, 31,
2, 56, 743, 425, 63450, 24, 0};
int i;
int isOdd;
int tmp;

printf("Show Odd Numbers\n");

//check and show each numbers
for(i=0; i<aryNum; i++) {
      ary[i] += 1;

      //check a number
      tmp = ary[i] % 2;
      if(tmp == 0) isOdd = 1;
      else isOdd = 0;

      //show a number
      if(isOdd){
           printf("ary[%d] = %d\n", i,
ary[i]);
      }
}
```

Figure 2 Source Code Editor



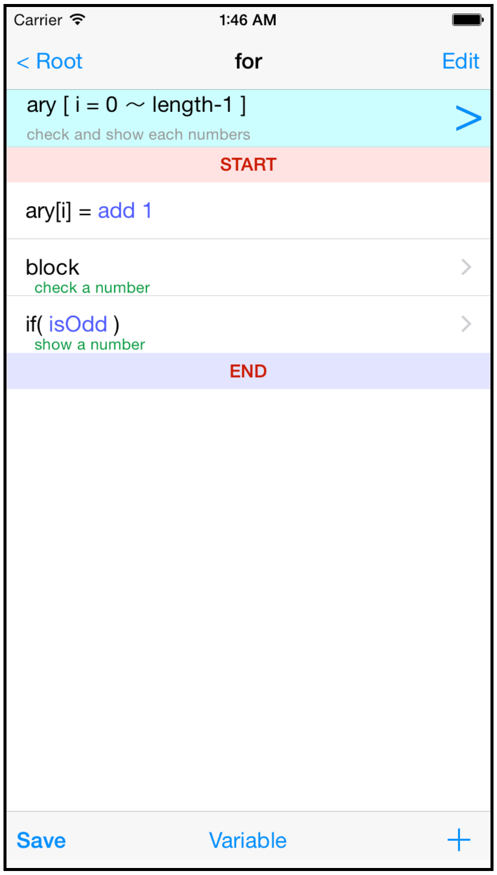
ary = 4, 7, 7, 49, 31, 2, 56, 743, 425, 6345…

printf("Show Odd Numbers")

for(ary[i = 0 ~ length-1])
check and show each numbers

Souce Code Editor has multi view blocking. If touch up the for statement of Figure 2, open new view as Figure 3.

Figure 3 For Statement (Multi View Blocking)



ary [i = 0 ~ length-1]
check and show each numbers

ary[i] = add 1

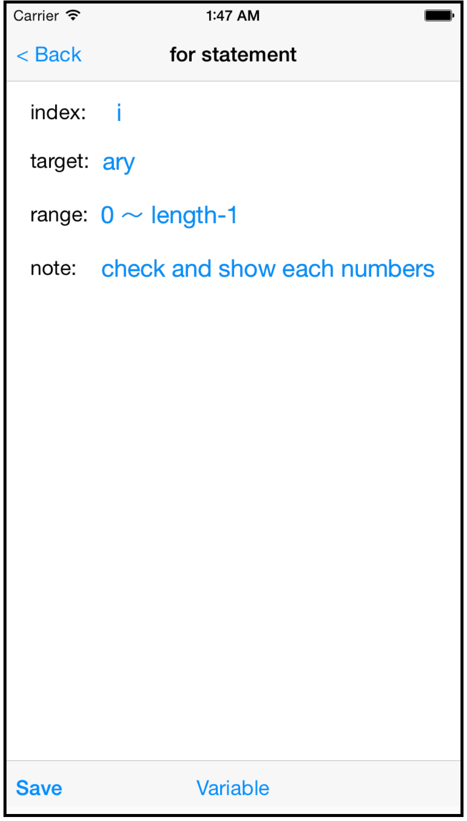
block
check a number

if(isOdd)
show a number

3.4 Statement Editor

Each statement is constructed by closed inputting. At first, users select an imperative. If the imperative needs some arguments, users select the arguments from fixed variables and values. The users cannot use computational expressions.
This prototype system has statement of “if”, “for”, “printf”, “scanf”. In addition, it has especially statements: “block” and variable setting. block statements do not operate. However, block statements create new view with a note. Variable setting statement include operators as like “=”, “+=”, “-=”, “*=” and “/=”. Those operators are shown “=” in this system.

Figure 4 Statement Editor (for statement)



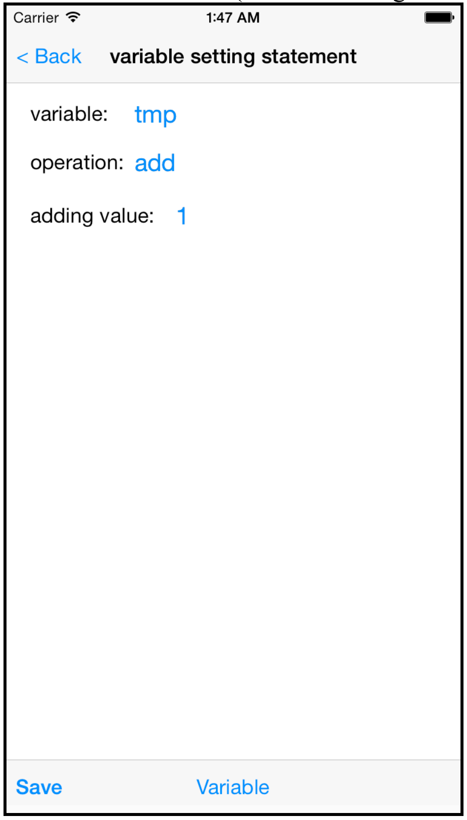
index: i

target: ary

range: 0 ~ length-1

note: check and show each numbers

Figure 5 Statement Editor (variable setting statement)



variable: tmp

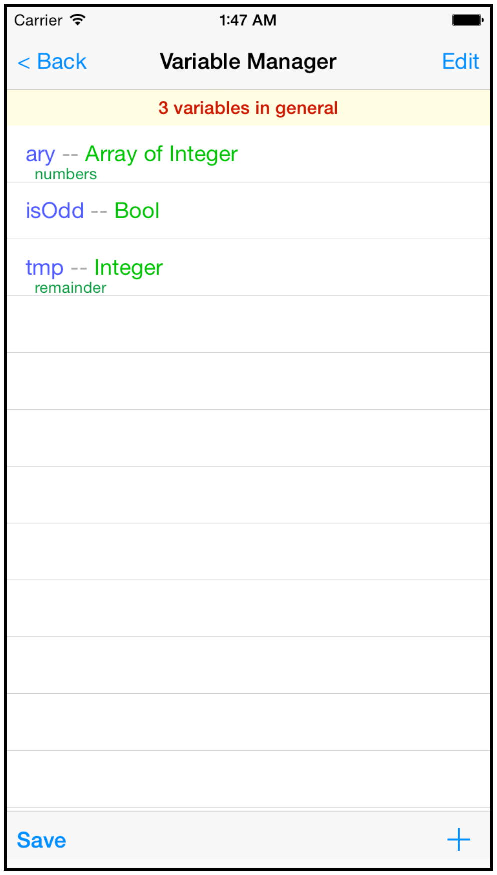
operation: add

adding value: 1

3.5 Variable Manager

This system allows using alphabets and numbers for naming to variables. Variable need not to be set data types. Each variable has a name and a value.
Prototype system uses only one source code file. This system should be simple. This system has no subroutines.

Figure 6 Variable Manager



Carrier 1:47 AM

< Back Variable Manager Edit

3 variables in general

ary -- Array of Integer
numbers

isOdd -- Bool

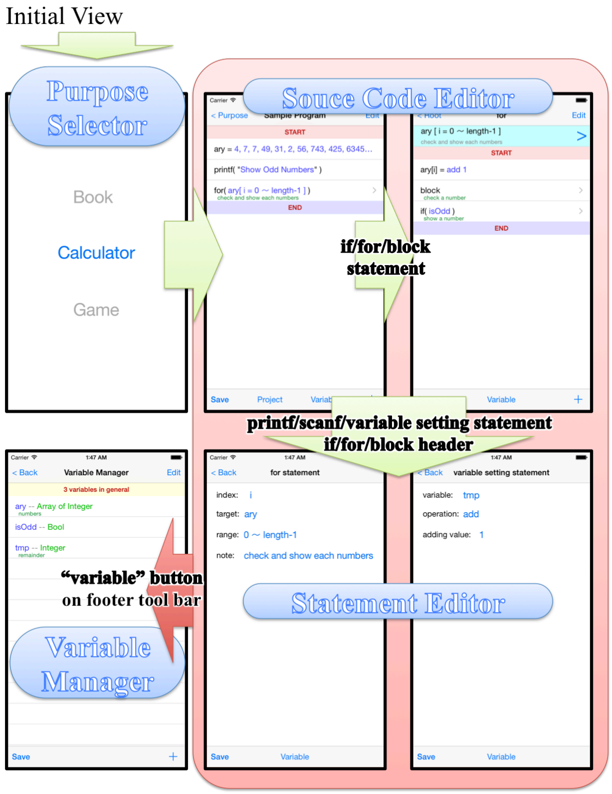
tmp -- Integer
remainder

Save +

3.6 Connections of Views

Purpose Editor view is the initial view. After the selection of purpose, a Source Code Editor view appears. In Source Code Editor views, when users touch up if/for/block statement cell, other Source Code Editor view appears. The other hands, when users touch up printf/scanf/variable setting statement cell, Statement Editor view appears. When users touch up headers of additional Source Code Editor Views, they move to Statement Editor views. Variable Manager view is a optional view. Source Code Editor views and Statement Editor views can move to Variable Manager view via “variable” buttons.

Figure 7 Connection of Views



4 CONSIDERATION

Proposed system is a multi-view system for programming learning. Multi-view makes understandability. Programing statements and blocks are written simply. In addition, the multi-view programming system has GUI like basic book application on smartphones. Users need not learn GUI operating. This system is good for programming beginners.

5 FUTURE TASKS

Prroposed system is a prototype. It has only 6 sorts of statements. We would try to experiment about this restriction is suitable or not.
In addition, this prototype system does not have outline of a project. We seem to need to show the outlines. We would research how show the outline in this system.

REFERENCES

[1] Tetsuo Ogino, and Takeshi Fujioka, “System Design and Analysis for Action Streams in MASUME: Runtime Enviroment for Educational Programming,” SIG Technical Reports, 2014-CE-124(7), 2014, pp.1-8.

[2] Kaduki Kawamura, “The trial of the programming training education by lecture less,” SIG Technical Reports, 2015-CE-128(21), 2015, pp.1-7.