# Dagstuhl Seminar Report

## Educational Programming Languages and Systems

Youyou Cong (Tokyo Institute of Technology)

# Seminar Overview

Participants background:

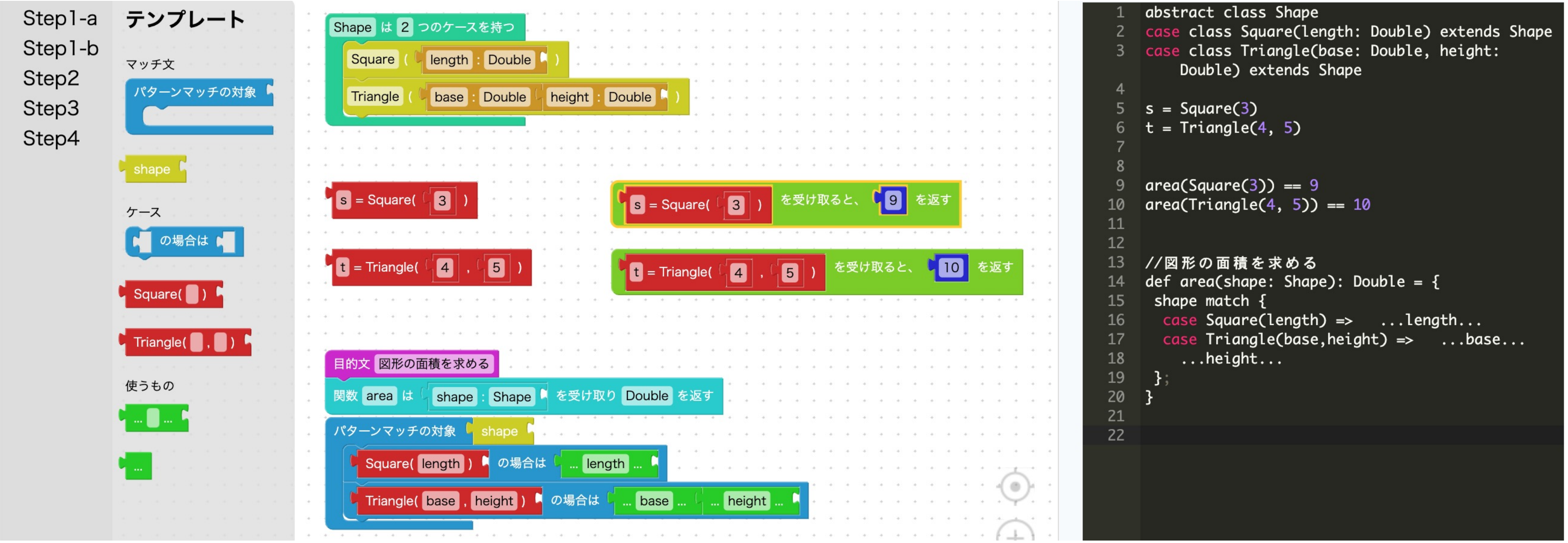
- CS/PL
- Cognitive science

Structure:

- 10-min talks (Mon-Wed)
- Breakout (Thu & Fri)

# Program Design by Blocks (Youyou Cong)

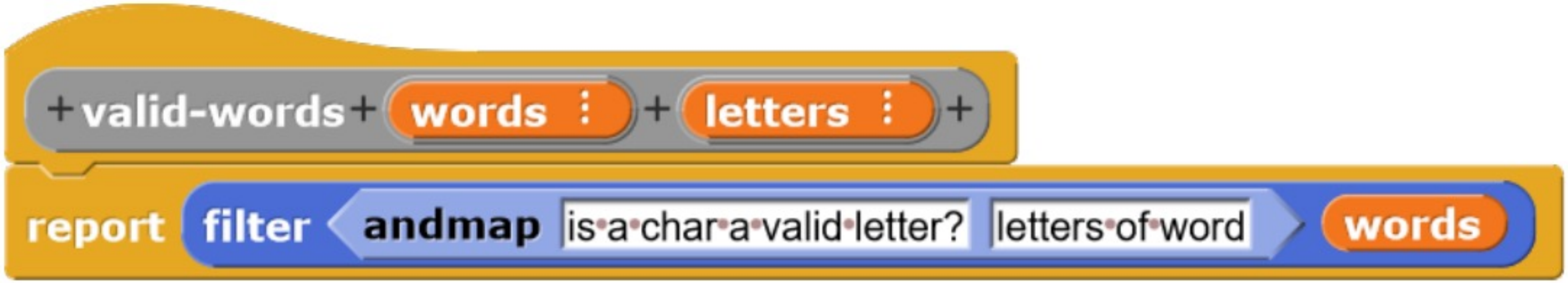
- Design in blocks, code in text
- Received positive feedback from students


Step1-a
Step1-b
Step2
Step3
Step4
テンプレート
マッチ文
パターンマッチの対象
shape
ケース
の場合は
Square( )
Triangle( , )
使うもの
... ...
...
Shape は 2 つのケースを持つ
Square ( length : Double )
Triangle ( base : Double height : Double )
s = Square( 3 )
t = Triangle( 4 , 5 )
s = Square( 3 ) を受け取ると、 9 を返す
t = Triangle( 4 , 5 ) を受け取ると、 10 を返す
目的文 図形の面積を求める
関数 area は shape : Shape を受け取り Double を返す
パターンマッチの対象 shape
Square( length ) の場合は ... length ...
Triangle( base , height ) の場合は ... base ... ... height ...

```
abstract class Shape
case class Square(length: Double) extends Shape
case class Triangle(base: Double, height:
    Double) extends Shape

s = Square(3)
t = Triangle(4, 5)


area(Square(3)) == 9
area(Triangle(4, 5)) == 10


//図形の面積を求める
def area(shape: Shape): Double = {
 shape match {
  case Square(length) =>   ...length...
  case Triangle(base,height) =>   ...base...
    ...height...
 };
}

```

# Program Planning via Higher-Order Functions (Shriram Krishnamurthi)

- Higher-order functions as primitives for planning
- Used to observe how students understand/use HOFs

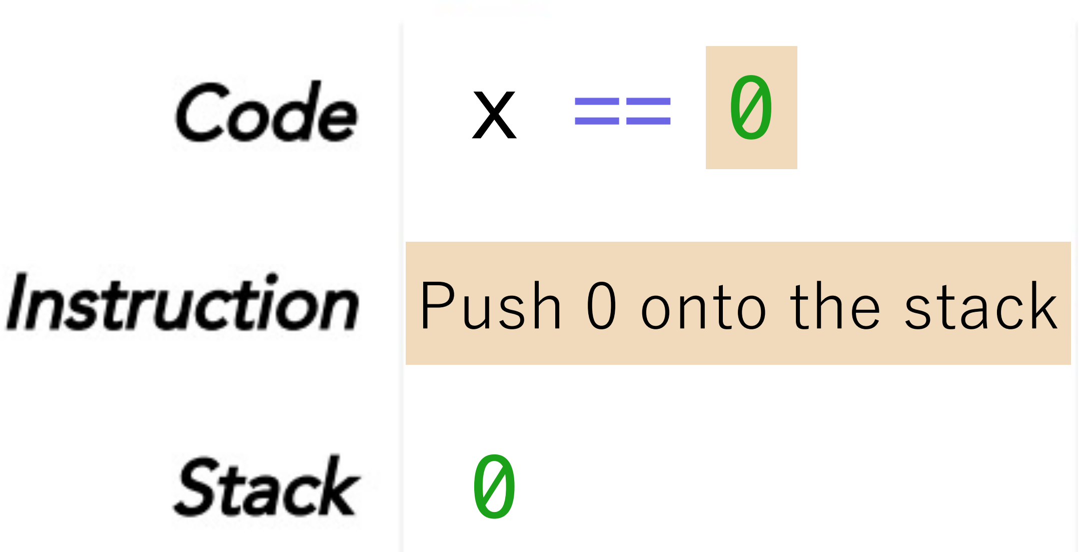


```
+valid-words+ words ⋮ + letters ⋮ +
report filter < andmap is a char a valid letter? letters of word > words
```

# PLTutor (Amy Ko)

- Semantic rules as causal relations

- Effective for learning tracing skills



| | |
|---|---|
| ***Code*** | x == 0 |
| ***Instruction*** | Push 0 onto the stack |
| ***Stack*** | 0 |

# Hedy (Felienne Hermans)

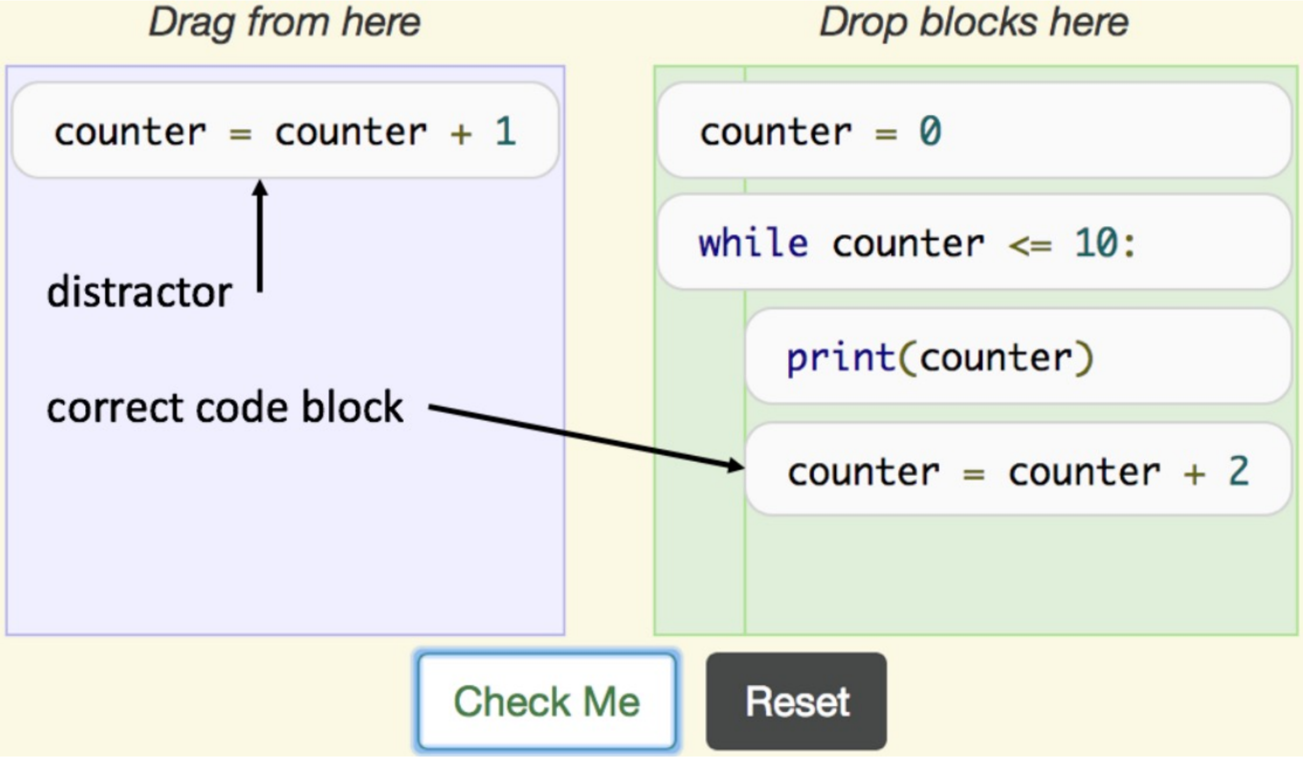
- Gradual learning via language levels

- Keywords in non-English languages

```
// level 1
print hello world

// level 4
print 'hello world'

// Japanese mode
かけ hello world
```

# Adaptive Parsons Problems (Barbara Ericson)

- Coding by dragging code fragments
- Support intra/inter-problem adaption



```
counter = counter + 1
```

```
counter = 0
while counter <= 10:
    print(counter)
    counter = counter + 2
```

# Evening Panels

1. Teaching at scale
2. Evaluation
3. AI in education

# Brainstorming Session

- What studies should we do together?
- What have we learned from building, deploying, and maintaining tools?

# Non-academic Activities

# What I liked about (this) Dagstuhl

- Small but diverse
- Not too packed, nor sparse
- Friendly and encouraging
- Good COVID policy

# Links

- Program design by blocks
- Plan composition via HOF
- PLTutor
- Hedy
- Adaptive Parsons problems
- Amy Ko’s blog post