

## 教員養成課程学生における熟達者のプログラミング的思考過程

柏原 健人\*・宮本 直樹\*\*

(2020年10月21日受理)

Computational Thinking Process of Experts in Pre-service Teachers

Kento KASHIWABARA and Naoki MIYAMOTO

キーワード：プログラミング的思考，熟達者，教員養成課程

本研究では、教員養成課程学生における熟達者のプログラミング的思考過程を明らかにすることを目的とした。具体的に、分析方法として発話プロトコルを「動きを分解」「組合せを考える」「試行錯誤」「自己モニタリング」「モデルチェック」といった5つのカテゴリーに分け、それらの推移をみることでプログラミング的思考熟達者のプログラミング的思考過程を看取った。その結果、以下に示す4点が明らかとなった。第1点は、プログラミング的思考過程では、「動きを分解」することが重要であること、第2点は、つまづきが小さい時の思考過程は、論理的な改善が中心であること、第3点は、つまづきが大きい時の思考過程は、試行錯誤的な改善が多くみられ、論理的な改善と相互に関係しあうこと、第4点は、「試行錯誤」から「動きを分解」する思考過程が重要であることである。また、熟達者のプログラミング的思考過程は、論理的な改善だけではなく、試行錯誤的な改善と関わりながら構成されていた。

### はじめに

小学校学習指導要領では、必要な資質・能力の育成の1つとして情報活用能力を育むことを掲げ、「児童がプログラミングを体験しながら、コンピュータに意図した処理を行わせるために必要な論理的思考力を身に付けるための学習活動」（文部科学省，2018a：22）を計画的に実施することとしている。また、小学校学習指導要領解説総則編では、学習活動としてプログラミングに取り組むねらいは、プログラミング言語を覚えたり、プログラミングの技能を習得したりといったことではなく、子供たちが将来どのような職業に就くとしても時代を超えて普遍的に求められる「プログラミング的思考」を育むこととしている（文部科学省，2018b：85）。これらのことから、今後さらに、プログラミング教育が盛んになっていくことが予想される。立田（2017）は、小学校に

---

\*日立市立仲町小学校      \*\*茨城大学教育学部

においてプログラミング教育を本格的に行う際、小学校教員の負担が増加することを危惧し、「現在小学校教員養成課程を持つ大学で、プログラミング教育を導入することが急務である」と述べている。山下(2018)は、2020年度から全面実施される小学校学習指導要領を受け、「小学校教員養成課程の大学生は、プログラミング教育を実施する能力を在学中に修得する必要がある」としている。つまり、教員養成課程を持つ大学において、プログラミング教育の導入が喫緊の課題となる。さらに加えて、新地・安影(2018)は、教員養成課程3年生のプログラミング的思考を機器動作に関わる説明記述分析から考察した中で、「自分が意図する一連の活動を実現するために、どのような動きの組合せが必要であり、一つ一つの動きに対応した記号を、どのように組み合わせたらいいのか、記号の組合せをどのように改善していけば、より意図した活動に近づくのか、といったことを論理的に考えていく力、すなわちプログラミング的思考を把握するための問や分析方法の提案が急がれる」と述べている。以上のことから、教員養成課程学生のプログラミング的思考の実態や思考様式、また、それらを分析・評価する方法が確立されていないことにも課題があるといえる。附言すれば、先行研究では、プログラミング的思考を分析・評価する際に必要なプログラミング的思考の過程が明らかにされておらず、また、研究の知見もない。したがって、教員養成課程学生のプログラミング的思考過程について調査を行うことは急務である。

一方で、新地・安影(2018)は、教員養成課程3年生109名に対して、質問紙調査により2つの質問の回答を求めた。1つは、「学生個々の情報機器の利用状況」、もう1つは、「機器の動作についての自由記述」である。「学生個々の情報機器の利用状況」においては、「ア. パソコン、イ. スマートフォン、ウ. タブレット型端末、エ. その他」について、「5. 1日2時間以上、4. 1日1回以上、3. 週に数回程度、2. 月に数回程度、1. 全く利用しない」の5件法により回答を求めた。「機器の動作についての自由記述」においては、「動作する仕組みが分かるように、『自動ドア』の動きについて文で説明してください。また、図や絵で補足しても構いません。『自動ドア』とはコンビニエンスストアにあるようなものを想定してください」という問題に取り組みさせた。「学生個々の情報機器の利用状況」からは、日常生活の中で、学生がどの程度情報機器を利用しているかについて、「機器の動作についての自由記述」からは、学生がどの程度のプログラミング的思考を持っているのかについて調査をした。本研究に関わりがある「機器の動作についての自由記述」のプログラミング的思考調査について詳しく述べると、まず、プログラミング的思考のレベル分けである。「機器の動作についての自由記述」の結果からプログラミング的思考をレベル分けする基準として、6段階の評価基準を利用した。評価基準を作成するにあたっては、「順次的な動作としてのドアの開閉に関わる表記及び開及び閉に関わる計測および制御に関わる表記の有無」を基にプログラミング的思考のレベルを分類している。その6段階の説明記述に対する評価基準は、「レベル6: 感知→開く→感知(時間経過)→閉じる(閉鎖中の停止動作など、開閉操作中の制御も含む表記)、レベル5: 感知→開く→感知(時間経過)→閉じる(閉じることにしても感知、または時間経過のカウントという動作(ステップ)を表記、レベル4: 感知→開く→閉じる、レベル3: 感知→開く、レベル2: 感知にかかわる表記はなく、開閉のみの表記(開のみ、閉のみの表記も含む)、レベル1: 順次にかかわる表記なし」である。その結果、6段階の評価基準を用い、学生のプログラミング的思考を分類すると学生109名中、レベル1が1名、レベル2が3名、レベル3が64名、レベル4が9名、レベル5が29名、レベル6が3名となった。これらの

結果から、「ドアの開動作及び閉動作がセンサー等による計測（感知）結果に従い制御されていることを説明できた学生は、レベル6及び5の計32名（29.4%）」「閉動作に関わる感知について言及のなかった学生がレベル3, 2の計67名（61.5%）」と大きく2つに分けて考察している。以上より、プログラミング的思考を段階的に分けることができるといえる。しかし、この先行研究では、「『センサーの感知』によって『自動ドア』が開き、『時間経過』によって『自動ドア』が閉じる」という「自動ドア」の仕組みをすでに知識として持っている学生もプログラミング的思考のレベルが高いと分類されてしまう。そのため、プログラミング的思考（自分が意図する一連の活動を実現するために、どのような動きの組合せが必要であり、一つ一つの動きに対応した記号を、どのように組み合わせたらいいのか、記号の組合せをどのように改善していけば、より意図した活動に近づくのか、といったことを論理的に考えていく力）よりも、「自動ドア」の仕組みを知っているという知識が大きく左右してしまうことが課題として挙げられる。したがって、プログラミング的思考が段階的に分けられることを踏まえながら、知識にあまり左右されることのない調査問題を作成する必要もある。

したがって、知識にあまり左右されることのない調査問題を作成し、そして、プログラミング的思考が低い教員養成課程学生では、研究の知見が得られないため、プログラミング的思考過程が表出する可能性が高い教員養成課程学生、つまり、熟達者を抽出し、プログラミング的思考過程を調査する必要がある。

## 研究の目的と方法

本研究では、教員養成課程学生における熟達者のプログラミング的思考過程を明らかにすることを目的とする。本研究の方法としては、まず、事前の調査問題により教員養成課程学生のプログラミング的思考熟達者ペアを1組抽出する。そして、その抽出したプログラミング的思考熟達者ペアにScratch<sup>1)</sup>を用いて実際のプログラミングを行ってもらう。プログラミングを行っている際の発話を記録し、プロトコル分析をすることで、プログラミング的思考過程を検討する。なお、Scratch教材を本研究において使用した理由は、プログラミング経験がほとんどない学生でもプログラミングに取り組みやすい教材であったためである（山下，2017）。

## 本研究におけるプログラミング的思考

文部科学省（2016）は、プログラミング的思考について「自分が意図する一連の活動を実現するために、どのような動きの組合せが必要であり、一つ一つの動きに対応した記号を、どのように組み合わせたらいいのか、記号の組合せをどのように改善していけば、より意図した活動に近づくのか、といったことを論理的に考えていく力」と説明している。また、プログラミング教育の中で育成する資質・能力のうち、「思考力・判断力・表現力等」において「プログラミング的思考」を育むこととしている。鹿江（2017）は、小学校教員養成系学生を対象としたプログラミング教育の実践の中で、「プログラミングを実施するときには、実行する前に期待される具体物の動きを再

度確認させること、及び予想に反した動きをしたときにはプログラムに戻り、各プログラムを実施していく過程をタブレットで観察しながら問題点を発見することにより、『試行錯誤』的に修正するのではなく論理的に考え修正することの重要性」を強調している。このことから、鹿江

(2017) は、「プログラミング的思考（自分が意図する一連の活動を実現するために、どのような動きの組合せが必要であり、一つ一つの動きに対応した記号を、どのように組み合わせたらいいのか、記号の組合せをどのように改善していけば、より意図した活動に近づくのか、といったことを論理的に考えていく力）」について「試行錯誤」的に記号の組合せを改善していくのではなく、論理的に考えることで改善していく力と捉えていることがわかる。文部科学省（2016）における「プログラミング的思考」の定義と鹿江（2017）の「プログラミング的思考」の捉え方を比較すると、どちらにおいても「論理的に考える」ことが重要となる。以上より、本研究では、「プログラミング的思考」を「自分が意図する一連の活動にするために、動きを分解し、その組合せを知識の有無に関わらず論理的に考え、改善していく力」とする。

## プログラミング的思考過程の調査

### プログラミング的思考熟達者の抽出方法及びその結果

まず、プログラミング的思考熟達者を抽出する方法を述べる。先に取り挙げた新地・安影（2018）の先行研究から、プログラミング的思考を段階的なレベルに分類し、大きく「プログラミング的思考熟達者」と「プログラミング的思考初学者」に分けた。しかし、課題点として、説明記述を求めた問題「動作する仕組みが分かるように、『自動ドア』の動きについて文で説明してください。また、図や絵で補足しても構いません。『自動ドア』とはコンビニエンスストアにあるようなものを想定してください」が、プログラミング的思考を適切に評価できるものであったかの検討が必要である。本研究では、先述したように「プログラミング的思考」を「自分が意図する一連の活動にするために、動きを分解し、その組合せを知識の有無に関わらず論理的に考え、改善していく力」とするため、「知識」に大きく左右されることのない調査問題を作成し、プログラミング的思考熟達者を抽出する必要がある。新地・安影（2018）が用いた問題では、「自動ドア」の仕組みを知っているかどうかという「知識」に大きく左右されてしまう可能性が大きいためそのまま参考にはできない。よって、「知識」の有無とは関係のない、よりプログラミングに近い調査問題を作成する。調査問題の作成にあたっては、先行研究に前例がないため、本研究における「プログラミング的思考」の定義をもとに作成していくこととする。本研究における「プログラミング的思考」の定義から、「知識の有無に関わらないこと」を留意し、プログラムの動作する仕組みを知っているか、そうでないかに関わらず、思考することができる調査問題にした。そのため、調査問題において使用するプログラムはあらかじめすべて用意しておき、それらの組合せを考えていく内容とした。また、「実際のプログラミングに近い」調査問題にするため、Scratch を参考に、ブロック型のプログラムを組み合わせしていく問題にした。これら2点を踏まえ、調査問題の設問を、「画面上の犬がキーボードの→を押したときには右に、←を押したときには左に、↑を押したときには上に、↓を押したときには下に動くようなプログラムを組もうと思います。どのよう

に下の記号を組み合わせればよいでしょうか。考えたプログラムを書いてください。また、プログラムは何回使用しても良いこととします」とした(図1)。なお、調査は、教員養成課程学生9名を対象に行った。

次に、調査問題から「プログラミング的思考熟達者」を分類する基準について述べる。本研究では、調査問題に対し、正しいプログラミングができていた者及びほぼ正しいプログラミングができていた者は「プログラミング的思考熟達者」とし、それ以外の回答は、プログラミング的思考初学者として分類を行った。すると、3つのパターンに分けることができた。「①キーボードを押したら、犬が動作するパターン」「②①に加え、『ずっと繰り返す』または『いっぺんに』のどちらかを使用するパターン」「③①に加え、『ずっと繰り返す』及び『いっぺんに』のどちらも使用するパターン」である。パターン①(図2)の回答をした学生は2名、パターン②(図3)の回答をした学生は5名、パターン③(図4)の回答をした学生は2名であった。付け加えておくと、パターン③は、調査問題に対し、正しいプログラミングができていた回答である。したがって、パターン③の回答ができた学生2名をプログラミング的思考熟達者として抽出した。

本研究で使用したプログラミング教材と調査問題により抽出したプログラミング的思考熟達者のプログラミングから得られたデータの分析方法を述べる。まず、本研究で使用したプログラミング教材について記述していく。前述した通りScratchをプログラミング教材として用い、プログラミング的思考熟達者2名がプログラミングを行った。プログラミングの内容は、「自動ドアを作成する」こととした。プログラミングは、人のモデルだけを提示しておくようにし、その他のプログラミングについて、プログラミング的思考熟達者2

### プログラミングに関する調査

卒業研究において、プログラミングに関する調査をしています。成績には一切関係ありませんのでご協力をお願いします。なお、今回記入していただいた内容は、卒業研究のためのみ使用し、外部に情報漏れることはありませんのでご安心ください。

選修 \_\_\_\_\_ 学籍番号 \_\_\_\_\_ 氏名 \_\_\_\_\_

画面上の犬がキーボードの→を押したときは右に、←を押したときは左に、↑を押したときは上に、↓を押したときは下に動くようなプログラムを組もうと思います。どのように下の記号を組み合わせればよいでしょうか。考えたプログラムを書いてください。また、プログラムは何回使用しても良いこととします。

犬を右に進める。	キーボードの→を押したら		
犬を左に進める。	キーボードの←を押したら	ずっと繰り返す	いっぺんに
犬を上に進める。	キーボードの↑を押したら		
犬を下に進める。	キーボードの↓を押したら	<small>※コの字の中にあるものを繰り返し実行できません。コの字の中にはプログラムをいくつか入れても構いません。</small>	<small>※コの字の中にあるものを縦行して実行できません。コの字の中にはプログラムをいくつか入れても構いません。</small>

図1 プログラミング的思考熟達者抽出のための調査問題

キーボードの→を押したら	
犬を右に進める。	
キーボードの←を押したら	
犬を左に進める。	
キーボードの↑を押したら	
犬を上に進める。	
キーボードの↓を押したら	
犬を下に進める。	

図2 ①のパターン

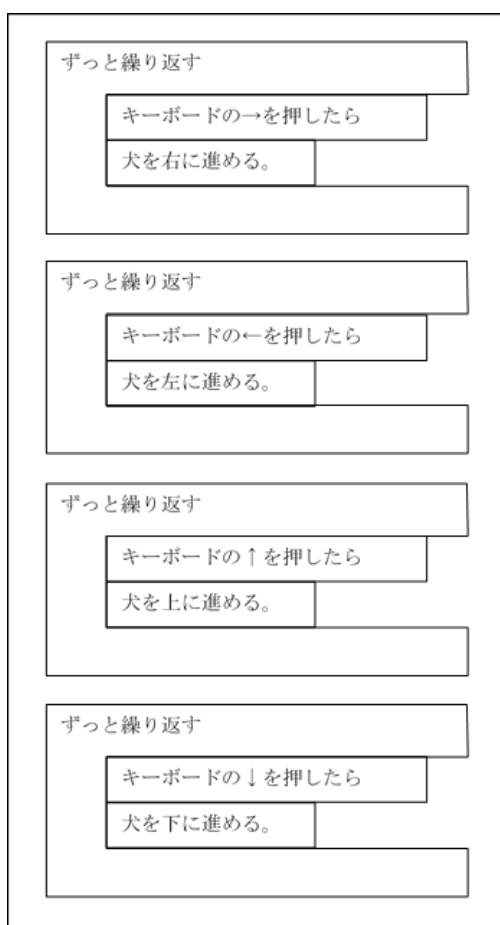


図3 ②のパターン

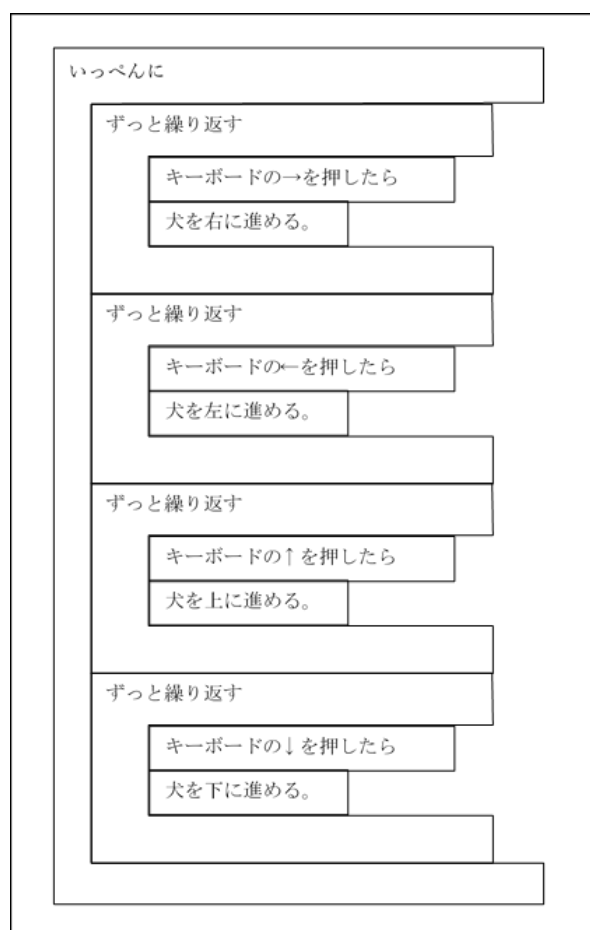


図4 ③のパターン

名が考える形で行った。プログラミングを行った時間は1時間であり、そしてプログラミング的思考熟達者の発話プロトコルをデータとして記録した。

### 本研究におけるプログラミング教材と分析方法

次に、プログラミング的思考熟達者のプログラミングから得られた発話プロトコル（資料）の分析方法を述べる。まず、得られた発話プロトコルを5つのカテゴリーに分類した。5つのカテゴリーは、「動きを分解」「組合せを考える」「試行錯誤」「自己モニタリング」「モデルチェック」である。「動きを分解」「組合せを考える」は本研究のプログラミング的思考の定義である「自分が意図する一連の活動にするために、動きを分解し、その組合せを知識の有無に関わらず論理的に考え、改善していく力」から抽出しており、プログラミング的思考過程の一部である。また、「試行錯誤」「自己モニタリング」「モデルチェック」については、プログラミング的思考の定義では

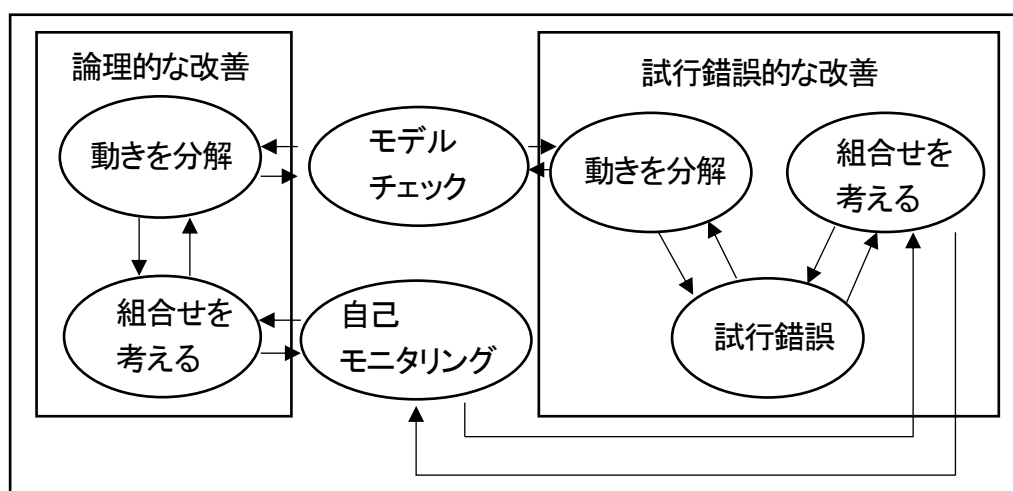


図5 「動きを分解」「組合せを考える」「試行錯誤」「自己モニタリング」「モデルチェック」の関係

ないものの、プログラミング的思考熟達者の発話プロトコルから多くみられた分類であるため、カテゴリーとして加えた。ここで、「自己モニタリング」及び「モデルチェック」について詳細を述べておく。「自己モニタリング」とは、プログラムした内容を確認、実行し、それまでのプログラミングについて振り返る場面であり、「モデルチェック」とは、作成されている人のプログラミングモデルを確認する場面である。加えて、5つのカテゴリーである「動きを分解」「組合せを考える」「試行錯誤」「自己モニタリング」「モデルチェック」の関係を述べる。プログラミング的思考熟達者の発話プロトコルからプログラミング的思考過程は、主に「動きを分解」し「組合せを考える」、そして、また「動きを分解」し「組合せを考える」といった論理的にプログラムを改善していく思考過程と、「試行錯誤」から「動きを分解」し、「組合せを考え」ながらプログラムを改善していく思考過程の2つに分けることができる。また、それら2つの思考過程に「自己モニタリング」と「モデルチェック」が関係してくるという形になっている(図5)。以上を踏まえ、プログラミング的思考熟達者の発話プロトコルを5つのカテゴリーに分類し、それらの推移をみることで、プログラミング的思考過程を読み取っていく。

### 教員養成課程学生のプログラミング的思考過程の分析結果及び考察

プログラミング的思考熟達者の発話プロトコルを5つのカテゴリーに分類し、それらの推移を分析した結果、まず、プログラミング的思考熟達者のすべての発話プロトコルにおける5つのカテゴリーの出現回数は、計63回であった。それらを出現回数の多い順に並べると、「動きを分解」が18回(28.6%)、「組合せを考える」が17回(27.0%)、「試行錯誤」が16回(25.4%)、「自己モニタリング」及び「モデルチェック」がそれぞれ6回(9.5%)であった。5つのカテゴリーを比較すると、「動きを分解」が18回(28.6%)と最も多く出現しているのに対

し、「自己モニタリング」及び「モデルチェック」はそれぞれ6回(9.5%)のみの出現であった(表1)。

表1 発話プロトコルにおける5つのカテゴリーの出現回数

カテゴリー	出現回数(割合)
動きを分解	18(28.6%)
組合せを考える	17(27.0%)
試行錯誤	16(25.4%)
自己モニタリング	6(9.5%)
モデルチェック	6(9.5%)
計	63(100.0%)

これより、プログラミング的思考熟達者は、主に「動きを分解」することを中心に思考をしているといえる。また、「動きを分解」する思考と比較すると、「モデルチェック」の出現回数が6回(9.5%)と少なかった。このことから、プログラミング的思考熟達者では、プログラムのモデルを参考にプログラムを組み立てていくという思考はあまりみられないことがわかる。つまり、プログラミング的思考熟達者は、「モデル」というプログラムの模範がなくても、自身の力で、必要な動きを分解し組み立てていくことができる。

次に、プログラミング的思考熟達者のすべての発話プロトコルにおける5つのカテゴリー間の推移をみていく。発話プロトコルにおける5つのカテゴリー間の推移は、計49回であった。推移回数の多い順に並べると、「動きを分解」からの推移は15回(30.6%)、「試行錯誤」からの推移は14回(28.6%)、「組合せを考える」からの推移は11回(22.4%)、「モデルチェック」からの推移は5回(10.2%)、「自己モニタリング」からの推移は4回(8.2%)であった。推移回数の多い「動きを分解」「試行錯誤」「組合せを考える」について詳しくみてみると、「動きを分解」からは「組合せを考える」への推移が一番多く、「動きを分解」からの15推移中8回(53.3%)であった。また、「試行錯誤」からは「組合せを考える」(14推移中6回(42.9%))、「組合せを考える」からは「試行錯誤」への推移(11推移中5回(45.5%))が

表2 発話プロトコルにおける5つのカテゴリー間の推移

→	動きを分解	組合せ考える	試行錯誤	自己モニタリング	モデルチェック	あるカテゴリーからの総推移回数
動きを分解		8	5	1	1	15(30.6%)
組合せを考える	2		5	2	2	11(22.4%)
試行錯誤	4	6		3	1	14(28.6%)
自己モニタリング	2	2	0		0	4(8.2%)
モデルチェック	3	1	1	0		5(10.2%)
あるカテゴリーへの総推移回数	11(22.4%)	17(34.7%)	11(22.4%)	6(12.2%)	4(8.2%)	



一番多い(表2)。

さらに詳しく、5つのカテゴリー間の推移をみていく。ここでは、発話プロトコルにおける5つのカテゴリー間の推移をみていく際に、2つの場面に分け、分析した。まず、2つの場面に分けた基準について記述する。プログラミング的思考熟達者の発話プロトコルを5つのカテゴリーに分類するにあたり、「A:動いた。まずはね。でも、何々にするだと瞬間移動になっちゃうから。B:徐々に動かせるんだ(資料 発話 N1217~N1218)」や「A:送り続けている、送って待つんだよね?送って待つんだ(資料 発話 N1438)」のような5つのカテゴリーに分類できない発話プロトコルが得られた。それらの発話プロトコルは、プログラムの改善を意識したものであり、それらの発話がみられた所で場面分けを行った。2つの場面はそれぞれ、最初の発話 N1001 から発話 N1218 まで(場面①)、発話 N1219 から発話 N1439 まで(場面②)である。次に、2つの場面の差異を述べる。2つの場面において、プログラミング的思考熟達者が「プログラミングをどうしたら良いかわからず」、思考が止まった秒数を計測すると、場面①は16秒、場面②は65秒であった。これより、場面①よりも場面②において、プログラミング的思考熟達者がつまずいていることがわかる。よって、2つの場面の差異を、つまずきの度合いの大小として考え、場面①を「つまずきが小さい時の思考の推移」、場面②を「つまずきが大きい時の思考の推移」として分析を行った。はじめに、「つまずきが小さい時の思考の推移」から分析結果を述べていく。つまずきが小さい時における5つのカテゴリー間の推移は、全部で26回であった。推移回数が多い順に並べると、「動きを分解」からの推移は9回(34.6%)、「組合せを考える」からの推移は6回(23.1%)、「試行錯誤」からの推移は5回(19.2%)、「モデルチェック」からの推移は5回(19.2%)、「自己モニタリング」からの推移は1回(3.8%)である。推移回数が一番多かった「動きを分解」からの推移を詳しくみると、「動きを分解」から「組合せを考える」への推移が最も多く、9推移中6回(66.7%)であった(表3)。つまずきが小さい時は、「動きを分解」から「組合せを考える」への推移が最も多かったことがわかる。

表3 つまずきが小さい時における5つのカテゴリー間の推移


	動きを分解	組合せを考える	試行錯誤	自己モニタリング	モデルチェック	あるカテゴリーからの総推移回数
動きを分解		6	1	1	1	9(34.6%)
組合せを考える	2		2	0	2	6(23.1%)
試行錯誤	0	3		1	1	5(19.2%)
自己モニタリング	1	0	0		0	1(3.8%)
モデルチェック	3	1	1	0		5(19.2%)
あるカテゴリーへの総推移回数	6(23.1%)	10(38.5%)	4(15.4%)	2(7.7%)	4(15.4%)	

分析結果から、「動きを分解」から「組合せを考える」という思考過程は、5つのカテゴリー間の関係のうち、「論理的な改善」と捉えられる(図5)。また、プログラミング的思考熟達者において、つまずきが小さい時とは、彼らにとって、プログラミングの課題が容易である時と言い換

えられる。これらのことから、プログラミング的思考熟達者において、プログラミングの課題が容易である際には、「動きを分解」し、それらの「組合せを考える」といったサイクルを繰り返しながら、論理的にプログラムを改善していくといった思考過程でプログラミングをしている。また、「動きを分解」から「組合せを考える」という思考の推移を質的に詳しくみてみると、「A：触れていたなら、私自身がこっちに動くよっていう風に動かせればいいんじゃない？ってことは、メッセージなんじゃないかって思う（資料 発話 N1097）」という「センサーの動きを分解」することから、「A：人に触れたなら。B：ドアにメッセージを送るんだ（資料 発話 N1101 や N1102）」という「センサーの動きの組合せを考える」思考の推移や、「B：こっち（左ドア）は人を感知したらのプログラムを作ればいいのか（資料 発話 N1112）」という「左ドアの動きを分解」することから、「A：人を感知を受け取った時っていうのがあるから、人を感知を受け取った時、ずっと、開くみたいなこと（資料 発話 N1113）」という「左ドアの動きの組合せを考える」思考の推移がみられた。これらの思考の推移から、プログラミング的思考熟達者は、「動きを分解」から「組合せを考える」という論理的な改善を行う際、直前に分解した動きの組合せを考えるという思考過程をもつといえる。つまり、プログラミング的思考熟達者は、「動きを分解」から「組合せを考える」という論理的な改善において、組合せを考える思考の直前に分解した要素を基に並べていくといった認知構造をもっている。

さらに、つまずきが大きい時の5つのカテゴリ間の推移について述べる。つまずきが大きい時の5つのカテゴリ間の推移は、全部で23回であった。推移回数が多い順に並べると、「試行錯誤」からの推移は9回（39.1%）、「動きを分解」からの推移は6回（26.1%）、「組合せを考える」からの推移は5回（21.7%）、「自己モニタリング」からの推移は3回（13.0%）、「モデルチェック」からの推移は0回（0.0%）である。推移回数が一番多かった「試行錯誤」について詳しくみてみると、「試行錯誤」から「動きを分解」への推移が最も多く、9推移中4回（44.4%）であった（表4）。

表4 つまずきが大きい時における5つのカテゴリ間の推移

	動きを分解	組合せを考える	試行錯誤	自己モニタリング	モデルチェック	あるカテゴリからの総推移回数
動きを分解		2	4	0	0	6(26.1%)
組合せを考える	0		3	2	0	5(21.7%)
試行錯誤	4	3		2	0	9(39.1%)
自己モニタリング	1	2	0		0	3(13.0%)
モデルチェック	0	0	0	0		0(0.0%)
あるカテゴリへの総推移回数	5(21.7%)	7(30.4%)	7(30.4%)	4(17.4%)	0(0.0%)	

分析結果から、つまずきが大きい時は、プログラミング的思考熟達者といえども、「試行錯誤」からの推移が9回（39.1%）と多くみられた。本研究では、プログラミング的思考を「自分が意図する一連の活動にするために、動きを分解し、その組合せを知識の有無に関わらず論理的に

考え、改善していく力」と定義した。つまり、プログラミング的思考とは「論理的」な思考であり、「試行錯誤」的な思考ではないと仮定してきた。しかし、仮定に反し、つまずきが大きい時は、「試行錯誤」からの推移、つまり、「試行錯誤」をしながら、「動きを分解」し、「組合せを考え」たりといった思考過程がみられた。

以上より、プログラミング的思考は、つまずきが小さい時において、「動きを分解」し、「組合せを考える」といった論理的な思考過程、そして、つまずきが大きい時は、「試行錯誤」をしながら「動きを分解」し、「組合せを考え」たりといった「試行錯誤」的な思考過程が相互に関係しながら身に付いていくことが考えられる。教育の現場では、プログラミング的思考を育む際、「論理的」に考えることを重視し、「試行錯誤」的に考えることは排除しようという風潮がある。しかし、「試行錯誤」的な思考をする場面は、「論理的」な思考をする場面と同等に、プログラミング的思考を育むうえで重要な思考過程であると示唆できる。

最後に、プログラミング的思考熟達者が「試行錯誤」をしている時の思考過程について考察する。「試行錯誤」からの推移が多くみられた「つまずきが大きい時の思考の推移」をみると、「試行錯誤」から「動きを分解」への推移が多いことがわかる。「動きを分解」する思考は、「動きを分解」し、「組合せを考える」といった論理的な改善に推移しやすいものである。したがって、プログラミング的思考熟達者は、プログラミングの課題が難しく、つまずきが大きくなると、「試行錯誤」から「動きを分解」し、そして、「動きを分解」したことから「組合せを考える」といった思考過程をもっていることが考えられる。つまり、このような思考過程を繰り返しながら、プログラムを考え、プログラミング的思考がより高次なものへと成長していくのであろう。

## おわりに

本研究では、教員養成課程学生における熟達者のプログラミング的思考過程を明らかにすることを目的とした。具体的に、分析方法を、発話プロトコルを「動きを分解」「組合せを考える」「試行錯誤」「自己モニタリング」「モデルチェック」といった5つのカテゴリーに分け、それらの推移をみることでプログラミング的思考熟達者のプログラミング的思考過程を看取った。その結果、以下に示す4点が明らかとなった。第1点は、プログラミング的思考過程では、「動きを分解」することが重要であること、第2点は、つまずきが小さい時の思考過程は、論理的な改善が中心であること、第3点は、つまずきが大きい時の思考過程は、試行錯誤的な改善が多くみられ、論理的な改善と相互に関係し合うこと、第4点は、「試行錯誤」から「動きを分解」する思考過程が重要であること、である。また、熟達者のプログラミング的思考過程は、論理的な改善だけではなく、試行錯誤的な改善と関わりながら構成されていた。これらを踏まえると、プログラミング的思考を教員養成課程の学生に身に付けさせていく際には、「試行錯誤」的な思考がみられた場合に、「試行錯誤」をしながら、プログラムに必要な動きを考えさせること、つまり、「動きを分解」させることを指導していく必要があると示唆できる。

今後の課題は、本研究で得られた知見を活かした指導方略の開発である。これは、別稿に譲りたい。

## 註

1)本研究で使用するプログラミング教材 Scratch-についての詳細は、次の「Scratch について」を参照されたい。MIT メディアラボ. Scratch について, <https://scratch.mit.edu/about> (2020年3月23日取得)。

## 附記

本論文は、第一著者が執筆した茨城大学卒業論文（2019年度）を大幅に加筆、修正したものである。

## 引用文献

- 鹿江宏明. 2017. 「小学校教員養成におけるプログラミング教育に関する研究 (1)ー教育版 LEGO MINDSTORM を用いた授業実践例ー」 『比治山大学紀要』 24, 111-119.
- 文部科学省. 2016. 『小学校段階におけるプログラミング教育の在り方について（議論の取りまとめ）』 [http://www.mext.go.jp/b\\_menu/shingi/chousa/shotou/122/attach/1372525.html](http://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/attach/1372525.html) (2019年11月20日取得)。
- 文部科学省. 2018a. 『小学校学習指導要領』（東洋館出版）。
- 文部科学省. 2018b. 『小学校学習指導要領解説総則編』（東洋館出版）。
- 新地辰朗・安影亜紀. 2018. 「機器動作に関わる説明記述分析から考察する教員養成3年生のプログラミング的思考」 『日本科学教育学会研究会研究報告』 33 (2), 39-42.
- 立田ルミ. 2017. 「小学校におけるプログラミング教育の導入と問題点」 『情報学研究』 6, 89-92.
- 山下祐一郎. 2017. 「Scratch を利用した教員養成課程におけるプログラミング教育指導力の育成」 『教職研究』 189-196.

資料

ここでは、プログラミング的思考熟達者におけるすべての発話プロトコルを掲載する。なお、A及びBが教員養成課程学生のプログラミング熟達者、Rが研究者（第一著者）である。

番号	
N1001	A: じゃ、やりましょう。
N1002	B: まずは、
N1003	A: とりあえずこいつ（エバリーウォーキング）を見てみよう。どうなってるか
N1004	B: これを押すと何するの？
N1005	A: そうだね、座標を169、85にする。あ、そしたらここに移動されるよ。
N1006	B: 右向き→、これか、これが押されたら、10ずつ変える
N1007	A: ああ、なるほどね、10ずつ動くことになってるのね。左向き←が押されたら、戻る。あじゃあ、そうだね。
N1008	B: なるほどね、これで、やっと（人か）動くようになったんだ。こんな感じで
N1009	A: 前後に動けるやつになったんだね
N1010	B: これを、あつ、長押しすると、そのまま行くんだ
N1011	A: だから、ずっと長押しじゃない？
N1012	B: そっか
N1013	A: あ、どうなんだろね、一応そういうことで考えよう。
N1014	B: これさ、このプログラミングとき、これを（人の絵を指して）一致させなきゃいけないんだよね。例えば今からさ、ドア作ったらさ、ドアとプログラミングが一致するようにしなきゃいけないっていうのは、どうしたらいいんだろ。
N1015	A: ああ、そういうこと。多分、このプログラミングの、このプログラムって、この絵に対するプログラムってことなんじゃない？
N1016	B: あ、なるほどね。
N1017	A: だから、ドアの絵のプログラムを、追加して、絵に対するプログラムをまた付けば
N1018	B: ああ。
N1019	A: そう反映させられるんじゃないかな
N1020	B: ドアとドアのプログラミング作って、人でプログラミングを付けるとか
N1021	A: 一回やってみよう。そしたら、まず、絵を作ろう。
N1022	B: どうやって？
N1023	A: 描く、選ぶ
N1024	B: 描くんじゃない？ ドア無、ゆら、（もともとある絵の中）
N1025	A: そっか、ドアは描くのか、んん、そうだね
N1026	B: こっからスタートなんだね
N1027	A: 四角、（ドアを描きながら）これを、左ドア、右ドアのほうか、どっちか？でもそうだね、開く、それぞれが開かなくちゃいけないから。
N1028	B: そのドアに向かって、こころめに入っていくんではよ。
N1029	A: うん、なるほどね、これさ、これを、あ、ちょっと待てよ、あ、判定がめっちゃちゃんピア。
N1030	B: そうそうそう。
N1031	A: そうだ、確かはね、じゃあ、一応同じ大きさのドアを..
N1032	B: 人が動いてプログラミングはできてるんだよね
N1033	A: そうだね、そうだね、そしたら、ドアの右にこえて、ドアの右にこえてとセンサーにつてじゃない？
N1034	B: そうだね
N1035	A: 考えたほうがいいのかは、後は何あるのかな？そのくらいだよ
N1036	B: センサーっていうのが、あれだよ、入ってくる時に動く、この範囲から外に出たら、閉まるっていう2段階
N1037	A: そうだね、そうだね
N1038	B: それをドアと一致させなくちゃいけないんだよね
N1039	A: 確か、何があるのかな、それにつては何か、センサーの絵はさきかきだよ
N1040	B: いい、描くんじゃない？
N1041	A: センサーを描こう、描く、描く、じゃあ直線で、適当に作っちゃうよ
N1042	B: この範囲でってことは、このセンサーの範囲でこの延長線上？こっから下？
N1043	A: いい、多分、プログラミングするときに、実装がやってみないと分からないけど、あれじゃないかな、ようは、人がこの線に触れたときとか..
N1044	B: あつ、触れたときか
N1045	A: ってことなんじゃない？センサーって... ま、そういうのがあるかかわかんないけど。
N1046	B: ああ、とりあえず、動くようにしてみるか
N1047	A: ちょっとこれ、これさ、これを、あ、ちょっと待てよ、あ、判定がめっちゃちゃんピア。
N1048	B: もうちょっとこっこのほうがよくなる？（人か）出たときこれだと..
N1049	A: あ、そっか、でもこれ、当たっちゃう、傾きすぎたな、これ
N1050	B: か、上にセンサーを待てていくか
N1051	A: ああ、なるほどね、上ならいけそうだね
N1052	B: （傾かしながら）もっとこっぴにしよう、これちょっと出るくらいじゃない？
N1053	A: いい、いい、それでいいよ、これで？つてきたから、これで、多分まあ。
N1054	B: ああ、それぞれのね、（それぞれプログラミングが組めることに気付く）
N1055	A: こういう感じだよ、だから、えっと、どうするか、ドアから考える？
N1056	B: 左ドアから、やろう。
N1057	A: でもさ、開く条件ってさ..
N1058	B: センサー、じゃあ、
N1059	A: センサーが当たった時？これとこれが（センサーとドアを指して）、なんか、ちょっと待てよ。なんかさ、この時みたいな（プログラミングを探す）、もしなんとかなら..
N1060	B: もし..（その先どうするか考える）
N1061	A: センサーが触れたとき、それって分かるのかな？
N1062	B: 触れさせなきゃいけないね
N1063	A: ちょっと待てよ、メッセージを受け取った時ってなんだ？メッセージを送る？
N1064	B: メッセージで、自動ドアからのメッセージってこと？あ、違う、センサーからのメッセージってこと？
N1065	A: え、そうできるんじゃあ、やっつて
N1066	B: じゃ、ここに（メッセージを送る）センサーが入るのかな？メッセージ1つってところ
N1067	A: いい、えっと、うーん、1回センサーでやってみよう、センサーの役目って、人と当たった時だよ、人と当たった時？
N1068	B: ようは、人をセンサーの範囲で感知したとき
N1069	A: （そしたら）当たんなきゃ無理だよ
N1070	B: 当てなきゃ
N1071	A: あ、ちょっと待てよ、これを、こうじゃない？（←触れたときのプログラミングを指して）こいつの名前ってなんだ？そうだよ、こいつだよ
N1072	B: あー
N1073	A: だから、このエバリーウォーキングに触れたときってのがあから、このセンサーは、エバリーウォーキングに触れた時
N1074	B: 触れた時、自動ドアを開かせる、動かせるんだよね
N1075	A: そうだね、そうだね
N1076	B: をしたらやっぱり、何々動かすっていうプログラミングがあるか..。これってどうなるんだっけ？この人って？
N1077	A: 見よう見よう
N1078	B: もし何々されたとき
N1079	A: あ、なるほどね、もし何々された時、じゃあ、もしこいつ（エバリーウォーキング）に触れたなら何をすればいいんじゃない？
N1080	B: そうだね
N1081	A: これが合ってるんじゃない？
N1082	B: ああ、そういうことだよ
N1083	A: 今こいつは（センサー）こいつ（エバリーウォーキング）に触れたならこうできるよって
N1084	B: そうだね
N1085	A: じゃあ、こうできる部分、イベントかな？
N1086	B: こっちはあ、これじゃない？
N1087	A: 調べるか？これは、調べる？
N1088	B: えっ、だってこの人形さ、もし、これが押されたら、あ、動きた動き。
N1089	A: 動きか、触れたなら、動かすか？あ、はい、はい
N1090	B: これでもいんじゃない？X座標を何々にするってことは、そこまで動かすってことじゃない？
N1091	A: それってセンサーの話じゃない？
N1092	B: あ、そっか
N1093	A: だから、うん？
N1094	B: ドアを動かすってやつが欲しいよね
N1095	A: うん、そっか、ドアが、ドアのほうに、もしセンサーが感知したら..そんなことが可能なのか？ どうしていいかわかんず8秒待たせ
N1096	B: もし何々に触れたら、ドアを移動するんだ
N1097	A: ドアのほうでやっつてあげないんじゃないかな、ドアがもし、センサーがこれに（エバリーウォーキング）触れていたら、私自身がこっぴに動くっていう風に動かせればいんじゃない？ってことは、メッセージなんじゃないかって思う。
N1098	B: ああ
N1099	A: 仮に、今センサーの話をしてるじゃん、センサーでもし、
N1100	B: あ、つながら
N1101	A: 人に触れたなら、
N1102	B: ドアにメッセージを送るんだ
N1103	A: そうだね、ドアにどうい、今こいつ（センサー）が発するんだよね
N1104	B: あー
N1105	A: メッセージ1を、メッセージ1..じゃあ、人を感知とかいする？1回
N1106	B: うん、え、人を感知ってこいつで触れた時点で感知してんじゃないかな？
N1107	A: あ、だからその、触れたのはこいつ（センサー）自身が人の存在を受容したけど、それを受容したよって発する。
N1108	B: ああ、なるほど
N1109	A: 発信を全体に送るみたいなことじゃない？
N1110	B: そうい、これ
N1111	A: それで、多分こっちは（ドア）

N1112	B: こっちは人を感知したらのアプログラミングを待たないのか
N1113	A: そうそうそう。人を感知を受け取った時っていうのがあから、人を感知を受け取った時、ずっと、開くみたいなこと？
N1114	B: そしたら、動きだ
N1115	A: これは、左ドアだから、えっと、X座標...
N1116	B: 何々ずつ変えるのかな？なんかさ、今X座標が10だったとするじゃん、で、こっち(右)に動くじゃんまあ、70とか
N1117	A: うんうん
N1118	B: すると、70にするだったら、これがいきなりバツって移り変わるってこと？
N1119	A: 一回やろう
N1120	B: やってみよう
N1121	A: じゃあ、こいつの(ドア)今のX座標が分かるんだよね。あ、これかな
N1122	B: そうだね
N1123	A: 107、じゃあ107を、じゃあ107だから40くらいにしてみる？
N1124	B: うん
N1125	A: 人を感知を受け取った時、こいつを(ドア)40にする。一回やってみよう
N1126	B: 何にも動かない
N1127	A: これ当たってはいらね？(人がセンサーに)ちょっと待てよ、もし...、こいつを押さなきゃダメなのか？(スタートしたら)
N1128	B: ああ、なるほど
N1129	A: 最初って、人を感知、あ、ダメだ
N1130	B: まずこれ動してる？
N1131	A: そういうこと？これって、触れたってことじゃなくて？
N1132	B: ああ、そうなるか
N1133	A: もうちょい下げてみる？
N1134	B: これを伸ばせばいいんじゃない？
N1135	A: ああ、ちょっと待てよ、こいつの(センサー)伸ばして...
N1136	B: これを(A)をセンサーにしてやってみよう、これって、どこにいても
N1137	A: そうだ、スタート地点は合わせるってことだ
N1138	B: じゃあ、それもしなきゃいけんじゃない？
N1139	A: 最初ってこと？スタートを受け取った時
N1140	B: これすら(ドア)、スタートを押されたときにしないと
N1141	A: でも、これはさ
N1142	B: でも人を感知っていう信号が
N1143	A: うん、を受け取ったらってことだから、こいつの(ドア)最初の位置が107、64だから、これちょっとここに(ドアのプログラミング)ぶち込もう。ああ、いやこれ、所定の位置に戻して、リセットするよってこと
N1144	B: でしかも、これは(ドア)人が入るとなると、こうなって、斜めから人が入っていくじゃん
N1145	A: そうだね
N1146	B: だから、最後までね
N1147	A: ああ、そうそう、最前面にするってあのかね？
N1148	B: なんか、これあったよ、なんだっけ、この人形タッチしたら、この人形のモデルで
N1149	A: ああ、了解了解、あ、じゃあこれの紫から見た目だ。確かに見た目だね
N1150	B: そういうこと
N1151	A: で、かつ、これを最前面で移動すると、なるほどね、したら、えっと、ずっと、ずっとX座標を40にするって、ちょっと待てよ、人を感知を送る。
N1152	B: 誰か？
N1153	A: 誰かあのかね？
N1154	B: そうだね、誰か送ってないもんね、これ
N1155	A: 誰か、誰か、誰か。(そういうプログラムを探しながら)
N1156	B: 多分これ、調べてるところだよ...、無い、イベントかな？いや、動きじゃない？
N1157	A: それか、思っただ、なんか、これはちゃんと閉まってんじゃない、こまごまという風に、(プログラムブロックの最後)スタートとしまりがあろうと思うの、ただ、こいつが(ドアのプログラム)何かまだ付いて足さなきゃいけないのかもしれない
N1158	B: ああ、人間さん最後何で終わってる？
N1159	A: ずっとで終わってるんだ、ずっと、送り続けるのかな？これを
N1160	B: そうだね、それだ
N1161	A: 一瞬、ビューって送っちゃって終わりになってる説あるよね、これいゝのかね？
N1162	B: いちばん外側の人じゃないか？
N1163	A: おお、どうでこうしてこう。(いちばん外側をずっとをプログラムする)
N1164	B: 動かしてみよう
N1165	A: やってみよう
N1166	B: ああ、まだ誰か送ってないのか
N1167	A: 誰か送ってあのかね？
N1168	B: 誰か送ってあのかね？
N1169	A: ああ、自分が思ったのは、これが(人を感知を送る)スイッチ的な役割だと思ったの
N1170	B: 感知かな
N1171	A: これボタンで触れた瞬間に押されるみたいで、スイッチ押されたまじのなイメージかな
N1172	B: あっ、分かった
N1173	A: で、スイッチが押されるから動くってあのかね？と思ったんだけど
N1174	B: そしたら、もし人を感知するを受け取ったならじゃない？
N1175	A: ヒトと待てよ
N1176	B: ほら、このドアは
N1177	A: でもそれじゃなかったっけ？
N1178	B: なってない
N1179	A: ああ、いや、そう受け取った時って、受け取ってると思うんだよね、もうちょい考えてみる？
N1180	B: これはこれで動くんだ。(人のプログラム)
N1181	A: あんまり
N1182	B: 別にそれは(センサー)動かしなから、動かしなから、えっ、全部スタートが押されたときでつけるの？ どうしていゝのかから8秒は懸
N1183	A: ちょっと待てよ、もう一度改めて見てみよう。それか、ちょっと待てよ。このX座標を40にするが、おつ(ここで、プログラムをクリックしたことによりドアが動く)ちょっと待てよ、なぜ？(動いた理由がわからない)
N1184	B: どうして動いたんだ？
N1185	A: マイクを動かして動かしたんだ？こいつは？止める。(プログラムを)(ここでスタートをクリックする)あれ？
N1186	B: 動いたやつね、(人が)(スタートを押したときに所定の座標に戻るようにプログラムしてため)
N1187	A: あ、これは無理やり動かしているが、(ドアを)
N1188	B: でも、動かしてもスタート押したら元の位置に戻る
N1189	A: ああ、そうだね
N1190	B: あ、でも感知を受け取ってない。(ドアのイベントプログラムを見て)
N1191	A: ああ、ああああ、ちょっと待てよ、動いた
N1192	B: これ、センサーの問題じゃない？(感知を送るのか)
N1193	A: 一回、こいつは(ドア)スタートを押されたとき、ちゃんと戻るようにしよう
N1194	B: ああ、定位位置に来るよ
N1195	A: そうだね、最初スタート地点で、それでもう1個人を感知したとき
N1196	B: 確か、何となく、自動ドアだから、くっついていゝんだよね
N1197	A: じゃあ、ここ定位位置しようか、121
N1198	B: 61に合わせればいゝのか
N1199	A: どっちも61にすればいゝのか
N1200	B: そうだね
N1201	A: で、最前面で移動する
N1202	B: これ、左ドア？
N1203	A: これはドア
N1204	B: ん？最前面になってるよ。(スタートを押してないため)
N1205	A: 誰か送れば、いや、これは、(スタートを押す)
N1206	B: あ、動いた、こうなっちゃった。(思ってた座標が40じゃなかった、(ドアのY座標))
N1207	A: あ、そうだった半角だ半角だ、半角じゃないと、61
N1208	B: マイナスも半角じゃなきゃいけんじゃない？
N1209	A: なるほどね、(動かし始めるが動かない)0ってことは感知してないってことでしょ、これを、これ、マイナスじゃあいゝのか？ハイフンとか
N1210	B: そういうこと、この小さいやつか
N1211	A: で、一回押してみ
N1212	B: ああ、なった
N1213	A: で、ごめん、まって、今1個思ったことがある。言ってもいい？これが全角だ。(ドアを動かす方のプログラミング)
N1214	B: ああ、そもそも
N1215	A: これをちゃんと(半角で)40にすれば動く説ない？で、実行で、ほら、ほら。(動く)
N1216	B: 来た
N1217	A: 動いた、まじだね、でも、何々にするだと瞬間移動になっちゃうから
N1218	B: 徐々に動かせるんだ
N1219	A: 動かすで、でもこいつは止めておけば、ずっと動いてくれると思うの、ん？ちょっと待てよ
N1220	B: じゃあ、X座標を何々ずつ変えるってこと
N1221	A: ああ、最高、ああ、いっちゃった、いっちゃった。(ドアが徐々にとこまでも動く)いやね、じゃあ、これを5とかにするか
N1222	B: 全角全角
N1223	A: ああ、危ない危ない、(動かし)
N1224	B: そしたら、止めなきゃいけない、どこで止める？
N1225	A: おっけー、おっけー、ちょっと待てよ
N1226	B: 止めるっていう動かないのか？
N1227	A: 行く、えっ、行く？
N1228	B: いや、なんかイベントであんじゃない？もしどこまで行ったらとか
N1229	A: ああ
N1230	B: あ、制動か
N1231	A: ああ、これずっとじゃなくて、ああ、でも、ちょっと待てよ、いや、えっとね、どうする
N1232	B: どうしていゝのかから8秒は懸
N1233	B: おおもってこれだね
N1234	A: これを(ドア)ここに動かすわけじゃないのか
N1235	B: ああ、なるほどね
N1236	A: 1秒で、秒とかにするとかやこいから
N1237	B: これだと、バツってなっちゃうよね

柏原・宮本：教員養成課程学生のプログラミング的思考過程

N1238	A: なっちゃうね
N1239	B: どのかの場所、どこかの場所へ行くって。
N1240	A: これさ。
N1241	B: どのかの場所に行かせるんじゃない？行くってことは、動いてってことだから。
N1242	A: そうだね、これを、多分、いいですか、これ多分、これにすると(セパレーターウォーキング) っていうところへ向かうってことよね、だから、だからさ、これをもう1個動かすわけじゃない？ていうのは、
N1243	B: 人を？
N1244	A: いや、人じゃなくて、無を。
N1245	B: ああ、それだったらマウスのポインターって何？
N1246	A: こいつじゃない？こいつ (マウスを指して)
N1247	B: ああ、そっか？
N1248	A: だから、こいつのその場所まで行ってってことだしよ。
N1249	B: でも、無を作っちゃったらさ、もし、これとちゃんと現実的に...
N1250	A: あ、でもそっか？
N1251	B: 自動ドア作るってなったら、無を設定しなきゃいけないんだよ、それは無理だよな。
N1252	A: できてるわけでもないね、でも、現実では、じゃあ、違うな。
N1253	B: どこまで動いていくのがあるんだよ、何々？
N1254	A: 止めておく、止めておく、止めておく、この動き(ドア)がずっとじゃがメってことなの結局、ずっとだったら無にしちゃうから。
N1255	B: そっか？
N1256	A: まで繰り返す。違うな。
N1257	B: 何々まで待つ、どういこと？ここに入るの何何の？X座標まで待つって出来るのかな？
N1258	A: マウスの座標でしょ？
N1259	B: ここで止めるってどうするんだ？片方ができれば(ドアの)もう片方もすぐできるんだよな。
N1260	A: うん、うん、うん、うん、反対に動かすだけだよな。
N1261	B: うん。
N1262	A: マウスが押された...、ちょっと待てよ。
N1263	B: 今、これ、止めておけばいい、居なくなった時の感知さえ出来ればいい。
N1264	A: 止めるのはあるよ。
N1265	B: おお、それ、すべてを止める？すべてを止めたなら、
N1266	A: そういこと？
N1267	B: センサーだね、動きでなくの？止まるって、
N1268	A: 動きで止まる...
N1269	B: あっ、これ(ドアの座標)最終目標がこれじゃん、たぶん。
N1270	A: うん、うん。
N1271	B: 例は、0で(ドアの座標)を入れてその下に何動かすってすれば、...、何動かすってそこにするとおんなじかな？
N1272	A: そういこと？なるほどね、じゃあ、これを、ありだよ、10回繰り返す10回繰り返す...、ずっと、これをさるとかして試してみよう？
N1273	B: で、この中にこいつを入れるんじゃない？
N1274	A: あ、違う違う違う、えっ？
N1275	B: ちょっと待って、この座標をここに、ここ。
N1276	A: しつらこれって、ヒュンって動いてからジュンジュンって感じじゃないかな？
N1277	B: ああ、なるほどね。
N1278	A: イメージ
N1279	B: やってみるか？
N1280	A: ま、ま、やってみよう、1回実行しよう。
N1281	B: ああ。
N1282	A: 飛んじゃうんだよ。
N1283	B: そうだね。
N1284	A: だから、一回それは試してみよう。
N1285	B: これで。
N1286	A: ずっとだから。
N1287	B: あっ、秒でいいんじゃない？
N1288	A: そういこと？
N1289	B: 1秒で座標まで行ったら別に速度っていうか何回か繰り返すっていいのかな？
N1290	A: うんうん。
N1291	B: これちょっと入れてみていい？これ、どのかの場所からこちのまうか？(座標のまう)
N1292	A: そうだね、こっちで目的になる場所。
N1293	B: そうだね、ここが0か？
N1294	A: でも、0でもちようどいぶんじゃない？(座標を調整して)いぶんじゃない？
N1295	B: ちょっと待って、1秒だと早いね、3秒とかにしよう。
N1296	A: おけおけおけ、ちょっとやってみよう。
N1297	B: (動いて)来たよ、動いた動いた、3秒じゃ遅かったね。
N1298	A: ちょっとね、ま、2秒ぐらいでちょうどいいんじゃない？
N1299	B: 来たんじゃない？
N1300	A: あるよ、あるある。
N1301	B: 動いて、一回、(動き方が少しおかしい)感知が...、いぶんいぶんじゃん、ちょっと待って、今思ったのが感知ってのがやっぱり触れてなきゃいけないんだ、これに(センサー)。これもうちよ、何ぼ、そう、ほら、これさきさき。
N1302	A: これさ、なんでさ、これを、触れて1回繰り返すからじゃない？こいつ(ドア)が動いてるのって、
N1303	B: もう1回やるよ。
N1304	A: 触れてる間も動かすんじゃない？触れて、こいつが(人)触れなくなった瞬間動くじゃん。
N1305	B: ほんとは、
N1306	A: こいつで何だろう？何でだろう？
N1307	B: どうして？
N1308	A: ほらこれでもったってことは、
N1309	B: 今それになったんだ。
N1310	A: うん、だから、なんで？面白い面白い。
N1311	B: ハハハハハハ。
N1312	A: なんて？触れたなら、感知を送る、触れたなら、感知を送る。ずっと、連続して信号を受け続けるから今、今からスタートするぞ今からスタートするぞ繰り返しのかな？
N1313	B: ああ、そういうことね、それなら、
N1314	A: 面白いね、だから、えっとね、ちょっと待てよ。
N1315	B: 1回、こいつを(プログラム)まらね？これ、センサーをちゃんと伸ばして一番端も伸ばして、(ドア)右も動くようにしよう。
N1316	A: おっけ、ちょっと待てよ、ああ。
N1317	B: こころ(センサーの距離)をそろえなきゃ、
N1318	A: そしたら、ええっとね、(伸ばすのではなく、線をまっすぐに消すことでそろえる)
N1319	B: なるほどね。
N1320	A: これさ、(辺りすぎ)かな？どう思う？これとさ、すぐ当たっちゃいそうじゃない？
N1321	B: ああ、そうだね、もうちょっと、狭くする？まうか？
N1322	A: これさ、どうもいぶんじゃない？(センサーの線を1本減らす)ちょっと下のこと？
N1323	B: これなら多分、先が触れてるよ。
N1324	A: ああ、このぐらいにしておけか？(センサーの高さ)
N1325	B: で、右のドアも動くようにしよう。
N1326	A: うん。
N1327	B: 左ってどうやって作ったの？
N1328	A: まず、押されたとき所定の位置に戻すよ。ええっと、X座標を63だったよね？
N1329	B: うん、今、入ってるやつじゃないの？
N1330	A: ああ、そうか？で、最初だよ、こっちは、最初、最初。
N1331	B: で、次のイベントか？
N1332	A: そうだ。
N1333	B: 何々を受け取った時でしょ、これだ。
N1334	A: 人を感知を受け取った時、ずっと。
N1335	B: ずっとだった？(確認)
N1336	A: そうだね、ずっと。
N1337	B: ずっと、何々のところまで何秒で何々する。
N1338	A: うん。
N1339	B: これは、X座標をずらすの？
N1340	A: どのくらいにすればいいかな、こいつが、
N1341	B: 61じゃない？こいつが、
N1342	A: ほんとだ、61にしよう。ずらっちゃうから、で、こいつが今いる場所が、121から0、120ずつだった、でもそんな、
N1343	B: 120ずらしたらどっか行っちゃうよな。
N1344	A: うん、見えなくなっちゃうね。
N1345	B: じゃあ、(か)かやれよう、に40くらいずらす？
N1346	A: 40ずらしたら、81と、こっちは、279。
N1347	B: 299じゃない？
N1348	A: ああ、そうだ、ごめん、Y座標を、
N1349	B: 61にする、でやってみよう。
N1350	A: で、所定の位置、で、なせか触れてちょっと外れてから、(動く)まあまあまあま。
N1351	B: まあ、ちょっと除けなってるな。
N1352	A: 通れることは通れると、通れるには通れるから、この当たってる間が、
N1353	B: 判定だよな。
N1354	A: うん、でも光っているから感知はしてるよ、絶対。
N1355	B: これ、動き続けると(人が)動かさないんだ。
N1356	A: うん？ちょっと待てよ、わかった。
N1357	B: まって、外に行ったら動いたな。
N1358	A: だからこれはあれじゃない？ここがまだ当たってる範囲だから、これを、センサー送るのをさずってとじゃなくて、1回でいいんじゃないかな？
N1359	B: なるほどね。
N1360	A: 1回ってなんの？1回、1回、1回だけ、1回だけ送る。
N1361	B: これ1回繰り返すって、あ、でも繰り返すしちゃうか？
N1362	A: まあ、でも、1回繰り返すとかでいいんじゃないかな？

N1363	B: ずっと1回繰り返す?まあ、やってみよう。
N1364	A: ああ、でもだめだ。あんまり変わってない。なんでだろう。あんまり意味なかったな。 - どうしても何かからず10秒状態
N1365	B: ずっとじゃなくでいい。
N1366	A: そう、ずっとじゃなくでいい。
N1367	B: じゃあ、もうそれなしで(ずっと) やってみたら?
N1368	A: 確か。
N1369	B: これはどうだろう。(動かない) あ、ずっとじゃなくとスタートしなかった。あ、終わってないからな。
N1370	A: ああ、すべてを止める? (すべてを止めるを入れる) なんだらうねこれ。(全部の動かない) あ、止まっちゃってんのかこれ。じゃあ、このスクリプトを止める。だったらこいつだけ(センサー) 止まるんじゃないか? だめだ止まっちゃってんだ。止めちゃダメなんだこれは。やめましょう。だから、ずっとは必要だね。 - どうしても何かからず3秒状態
N1371	B: ずっとこれは感知を送り続けるんだよね?
N1372	A: うん。
N1373	B: 触れたら、開くんだけね。
N1374	A: なんか、何秒でどこまで行っていく風にしたら、止まるけど、触れている間感知しなくなったじゃん。(動かない) で、最初のこっちは移動するってやつだったら、止まらなかったけど、触れている間も動いてたよな。
N1375	B: これ、人間の動きとドアの動きは同期か? だってさ、ここにさ(センサー) ちよって触るとするじゃん。
N1376	A: うん。
N1377	B: で、すぐ戻るとするじゃん。でも1回センサーに触れたら自動ドアが開くよね?
N1378	A: うん。開く開く。だから、その開けるよって動作を1回だけしてくれればいいよ。
N1379	B: うん。
N1380	A: 思ったんだけど、いや、ちょっと待てよ、もし何かに触れたら、触れた... うわー。じっくりやりたくな。
N1381	B: いやでもなんか、詰まってきたね。
N1382	A: ハハハハハハ、悔しいな。 - どうしても何かからず9秒状態
N1383	B: 触れたら動いてほしいんだよね。でもこの感知は当たってんだよね。
N1384	A: うん。
N1385	B: ん? 触れたら、触れたら動か
N1386	A: 触れたら以外あんのかな? (条件) 無いんだよね。んー、触れたら感知を送る。 - どうしても何かからず14秒状態
N1387	B: んん、難しいな。
N1388	A: いや、成功したいよね。(座標を何かに) するは同期動作だもんね。するは同期動作で、何秒で行く...
N1389	B: 1個1個のね、現象を割ってほしいと無理なんだね。
N1390	A: んん、にする。動かす、の場所に行く、どっかに行っちゃったのかこれ。まっでまっで、最初の位置にこだったっけ? いやこれ、ずる。送って待つ、送って待つ?
N1391	B: 待つってどういうこと?
N1392	A: 動かさない、1回やってみよう。(動く) ああ。
N1393	B: 決まった。
N1394	A: これ決まったんじゃない?
N1395	B: 偶然当たったんだけど、どういうことなんだろう? これほ?
N1396	A: なんか思ったの。送ってさ、センサーが信号を送るよってさ送ってこれをドアが感知してさ、あ、感知したぞー、ドア開くよっていう風感知、開けるっていうムーブのこの時間軸があって、こ ういふ流れだと思うの。だけど、これがセンサーが感知している間、今送って1回待っている状態なんだと、待たないといけない。なんだから送る続けられる状態だと思うの、で、受けた受けた今から動かすぞ今 から動かすぞの連続で、止まっちゃってらんじやないか?
N1397	B: だから、動いてない止まっちゃうの?
N1398	A: そうそう、そんで、こいつが離れた瞬間に、動き出すっていう動作が始まったわけじゃん。それは、1時間このセンサーの連続が来てた信号が途切れたから、じゃ、今から(スタート) っていうことなのかな なって思っ
N1399	B: ちよっともう1回見して。
N1400	A: 送って待つ。
N1401	B: おお、いいじゃん。
N1402	A: 当たってると、動いて(ドアが) このまじけると。
N1403	B: あとは戻す動作必要だ。
N1404	A: そうだね、閉めればいいんだ。これで。
N1405	B: うん。
N1406	A: 閉めるから、待てよ、感知...
N1407	B: あ、もし、センサーの外に行ったらな。
N1408	A: そうだね、こいつが(1) 触れたら...
N1409	B: 距離から出たらってことでしょ?
N1410	A: 触れたら... あ、っていうことは、触れてない状態のとき、人を感知してない状態のときは、
N1411	B: あ、そうだ。
N1412	A: えっと、こいつら(ドア) は基本的に戻り続けている。閉まり続けているようにして、触れた時に開ける。っていう風にするよ。
N1413	B: 触れてない状態を維持するのよ。触れてない状態に戻るやつだから、もし... 何かあれば条件...
N1414	A: 触れてないならなってるあのか?
N1415	B: 閉鎖... あ、もし触れてない状態ならいい? これ。
N1416	A: あ。
N1417	B: これどうやって解除すればいいの?
N1418	A: もし、触れたら...
N1419	B: ああ、もし触れたらで、ここに入ると(1段目) 触れたらなんじゃない? ここ空白で、下に入ればもし触れてない状態ならなんじゃない?
N1420	A: ああ、もし触れたらだよ?
N1421	B: ああ、いいじゃん。
N1422	A: で、何とかがなければ。
N1423	B: あ、これは(人を感知を送る) ここに(1段目) 入るよね。でも、もし触れてない状態...
N1424	A: ああ、あ、えっと、そしたら、逆に触れてない状態を送っちゃえばいいんじゃない? もう1個作って。
N1425	B: ああ、なるほど。
N1426	A: 人未感知、で、これをずっとずっと、これでも動作は(前までの) するのよ、するする。
N1427	B: で、未感知の信号を送ってあげればいいんだ。
N1428	A: そうだ、未感知は基本的にこっちはっていう風にするよ... そうだね、そうだね。
N1429	B: もう1個人の人未感知を受け取った時をばねのよ。
N1430	A: いや、ええ、人を未感知を受け取った時、ずっと、
N1431	B: 何秒か(位置) に居るよ、それよりかは
N1432	A: 頭も頭も、めっちゃめっちゃいいえ。そうだ、そうだ。
N1433	B: この座標(位置) を打ち込めば...
N1434	A: 121の61。やってみよう。今人を感知している状態でしょう? (動いたまま戻らな)
N1435	B: 動かないぞ。
N1436	A: 行って戻ってってことは、あ、どっちも受け取ってる今。
N1437	B: そうか、何したらいやめかなくなってるんだ。センサーが
N1438	A: そうだ、感知してし、離れたら未感知もしてるし... じゃあこれ一掃こしちゃだめってこと? あ、まって、送り続ける、送って待つんだよね? 送って待つんだ、そうだ、送り続けたらさっさと うか。
N1439	B: そうだ、さっきと同じことした。
N1440	A: 送って待つ。え? (動かない)
N1441	B: やや、感知してない。
N1442	A: なんてだ? 落ちそうか? 人未感知を受け取った時...
N1443	B: 今、受け取ってるね。
N1444	A: 受け取ってるね、あ、今触れてないから人未感知を常に送り続けている状態だ。人未感知を受け取ってこいつは(ドア) ここに居るよという状態だ。
N1445	B: ああ、感知したらシステムをそっちに切り替えなきゃいけないんだよね。
N1446	A: そうだね。
N1447	B: えうするよ、センサーか、これ、ずっとなってるってことは
N1448	A: 今感知状態だね、これ、今感知、感知だけ。
N1449	B: これさ。
N1450	A: これさ、感知と未感知がどっちもなってる状態なのこれ? (違う) でもこれさ、感知してるよね、感知してるはずだよ未感知
N1451	B: 今未感知
N1452	A: なせだ?
N1453	B: これ最初で受け取ってるやつを繰り返すんじゃない?
N1454	A: ああ、分かるね。
N1455	B: もし最初こいつにしたら、(センサーに) 触れていたらもう感知をずっと受け取るっていうことだから... 未感知から感知へ行くための条件を考えなきゃいけない。
N1456	A: そうすると、もし何とかなら入れる?
N1457	B: これさ、同じに入ればいいんじゃない? センサーと、これと。もし感知を受け取ったらってやつあんなだけ? ここにはさっさと入るんだだけ?
N1458	A: ダイアマークじゃない? えええ。
N1459	B: ずっと未感知状態だもんね。今、スタートが未感知だもんね。
N1460	A: そうだね、1回。
N1461	B: これさ、あ、でもずっとにしないといけないよ、行き来したときにできないよ。
N1462	A: うん、人がまだセンサーに届いてない状態のときは、未感知状態なんだよね。要は、センサーから離れた状態は未感知状態だから、閉めようとする。開いている状態であったとしても閉めようとする状態が 未感知状態じゃん。
N1463	B: そうだね。
N1464	A: で、人がセンサーに当たったら、こいつは開こうとして欲しいから、感知状態になって欲しい。押されたときずっと... もし触れたら感知を送って待つ、触れてない状態は未感知を送って待つ。
N1465	B: これ、1回だけ、もうこれで終わってるんだよね、この時点でこのシステムは終わってる。
N1466	R: 1時間が経ちましたので、終わりにしてください。