

和算・数学・プログラミングの関係性に関する一考察 A Study on the Relationship between Wasan, Mathematics, and Programming

内田 保雄[†]
Yasuo Uchida

井田 志乃[‡]
Shino Ida

西田 若葉[†]
Wakaba Nishida

1. はじめに

明治以前に日本において独自に発展した数学を「和算」と呼んでいる。和算は、特に江戸時代において飛躍的に進化を遂げたが、明治以降に西洋数学が導入されることにより衰退した。しかしながら、和算は日常の算法から高等数学に至るまでの幅広い領域を網羅しており、またその思考方法は今日的意義を有していると考えられる。とりわけ小学校段階における算数分野への活用については、さまざまな研究や教育実践がなされてきている。一方、プログラミング教育必修化の世界的な流れを受けて、日本においても2020年度から小学校段階からのプログラミング教育が必修化された。そこでは、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を通じて、問題発見・解決能力等の学習の基盤となる資質・能力の育成を目指すよう指摘している。しかしながら、プログラミング教育教材の量・質ともに十分でなく、その充実が焦眉の急となっている。そこで本研究では、おもに和算書「塵劫記」[1]に含まれる題材を中心として、和算と数学との関係に加えてプログラミング的思考との関係性について考察することにより、効果的なプログラミング教育実施の一助とすることを目的とする。そのために、塵劫記に含まれる題材の中からプログラミング教育の導入段階に適した題材を選択して教材を試作し、プログラミング授業で試用する。その結果を踏まえて、今後の本格的な教材開発や教育実践に向けた問題点や課題を分析・検討する。

2. 先行研究・関連研究

中央教育研究所による小学校算数における和算の活用についての研究では、多くの和算の問題中から学校の授業で活用可能なものを取り出し、その一つひとつをどの学年で活用可能かを検討している[2]。

算数・数学教育改革の視点から和算を取り入れた研究がある[3]。和算の舞台となった算術道場や、全国に広く分布し当時の数学レベルを今に伝える数々の算額には、情熱をもって算術を学び追求した、当時の人々の前向きな意欲が強く表れている。その熱意を取り入れ、「創造的なAL活動」を行うために、現代の算数・数学の授業の延長線上として実現可能と思われる具体的な方法を提案している。

本研究が対象としている塵劫記を算数教育に取り入れる意義を検討した研究がある[4]。塵劫記に取り上げられた問題は、数学的な思考力を働かせることによって解決できる問題が多い。考える楽しさが入り、帰納、類推、演繹の他に特殊化、一般化、発展などの数学的な思考力を培う題材がたくさんちりばめられている。小学校で興味を喚起する問題を解決し、考える楽しみを感得すれば、数学的な思考

力が養われるのではないかと指摘している。

「科学技術離れ」、「理数科離れ」が問題となったとき、その原因の一つとして、数学が日常生活からかけ離れているという高校生徒の認識があげられた。そこで、数学の有用性を実感させるため、自然から生じる事象を例に用いた数列、特に漸化式を扱った教材の提案を行った取り組みもあった[5]。

プログラミングの教育と数学的思考の関係について考察している研究がある[6]。そこでは、数学教育の立場から、プログラミングの教育が数学的な考え方の習得の場面として機能する可能性について検討している。そして、学校における普通教育の対象になり得るプログラミング教育は、変数概念の獲得、一般化の考え方、論理的思考力の養成、体系化等に貢献する可能性を指摘している。その中で、コンピュータプログラミングの教育が数学的思考や変数概念のような基礎的な概念発達を増進するという報告や数学教育としてのコンピュータプログラミングの指導の効果として、論理的思考力が養えることと数学を身近なものとして捉えられることを紹介している。

中学校段階で、近年その必要性が指摘されているプログラミング的思考の育成を、数学科において、どのように実践できるかに着目した研究がある[7]。第3学年の素因数分解の学習における発展的課題として、二進法と十進法の変換プログラムの作成を行う授業プランを提案し、トライアンドエラーを通して獲得されるプログラミング的思考と二進法に対する見方の深まりの関連について考察している。

プログラミング教育における論理的な思考について論じた研究がある[8]。まず、論理的な思考は、プログラミング教育に限らず、数学や算数は典型的な論理的思考を子どもたちに要求し、理科も自然科学を理解する上で、論理的な思考が必須であるとしている。そこで、大学生を対象にして、プログラミングの問題に対する正解率と、他教科の問題の正解率との相関を調べることによって、その関係構造を明らかにしようとした。その結果、プログラミング教育と他教科との相関では、数学・理科だけではなく、他教科との相関も見られ、総合的な論理的思考が必要とされることを確認している。すなわち、プログラミング的思考は、他教科・領域の論理的思考と関連するような総合的な論理的思考ではないかと推測している。

また海外に目を向けると、英国の学校における教育戦略の調査において、学習の文脈化、共同作業、コンピュータショナルシンキング、コードトレース、アンプラグドスタイルなどの有効性が示されている[9]。

3. 提案手法

塵劫記の中から、「盗人算」と「俵杉算」を題材として選択し、和算、数学、プログラミングによるそれぞれの解法を示し、その関係性を探る。

[†] 宮崎産業経営大学 Miyazaki Sangyo-keiei University

[‡] 宮崎公立大学 Miyazaki Municipal University

3.1 盗人算

塵劫記では、次のように説明されている。「橋の下で盗人たちが何やら話しているのが聞こえる。どうやら、盗んだ「きぬ」(絹)(反物のこと)を分けようとしているようである。ところが、8反ずつ分けると7反足らず、7反ずつ分けると8反余ってしまい、困っているようである。さて、盗人と反物の数はいくらか」というものである。なお、このような問題は一般的には過不足算と呼ばれている。

3.1.1 和算による解法

8反と7反を加えると15になる、これが盗人の人数であるとしている。

この問題を、面積図を用いて解くと図1のようになる。

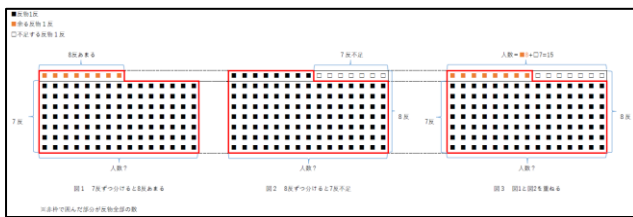


図1 面積図

3.1.2 数学による解法

盗人の人数を x として方程式を立てると次のようになる。

$$8x - 7 = 7x + 8$$

これを解くと

$$x = 15$$

と求まる。したがって、 $8 \times 15 - 7$ より、きぬの数は 113 反となる。

あるいはまた、きぬの数を y として

$$(y-8)/7 = (y+7)/8$$

を解くと

$$y = 113$$

と求まる。

3.1.3 プログラミングによる解法

最初に盗人の数を 0 人と考える。普通は 0 人ということはあるが、コンピュータの世界では 1 からではなく 0 から始まることが多いので 0 とした。なお、この例では 1 から始めてもかまわない。

8反ずつ分けると7反たりないので、反数を求めるには

$$0 \text{ 人} \times 8 \text{ 反} - 7 \text{ 反} = -7 \text{ 反}$$

となる。

同時に、7反ずつ分けると8反余ってしまうので、反数を求めるには

$$0 \text{ 人} \times 7 \text{ 反} + 8 \text{ 反} = +8 \text{ 反}$$

となり、2つの条件で反数が同じではないので、0人は正しくないことになる。

次に1人増やして、計算すると

$$1 \text{ 人} \times 8 \text{ 反} - 7 \text{ 反} = 1 \text{ 反}$$

$$1 \text{ 人} \times 7 \text{ 反} + 8 \text{ 反} = 15 \text{ 反}$$

と今度も同じではなく、問題の条件が成り立っていない。そこで、同じように順番に盗人の数を 15 人まで増やしたとすると、

$$15 \text{ 人} \times 8 \text{ 反} - 7 \text{ 反} = 113 \text{ 反}$$

$$15 \text{ 人} \times 7 \text{ 反} + 8 \text{ 反} = 113 \text{ 反}$$

となり、同時に条件を満足する。

したがって、盗人の人数は 15 人、また、きぬの数は 113 反ということになる。

答えが求まったので、これ以上人数を増やして処理を続ける必要がないので、繰り返し処理を終了する。

3.2 関係性について

この盗人算においては、和算では図形による解法がベースになっていると想定される。また、数学の場合には代数の考え方により、不明な値を変数として仮想して、式という形で表現して解を探るものである。プログラミングによる解法の場合には、所与の条件下においてあらゆる組み合わせを調べることにより解を求めるものである。いずれにしても、解としての変数の内容の求め方は異なるが、所与の条件に合致する、変数に入れるべき値を探しているという点では本質的に同一と言える。

プログラミング的思考との関係は、その定義「プログラミング的思考とは、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」に含まれる要素との対応を考慮する。すなわち、①「自分が意図する一連の活動」は解の導出、②「動きの組合せ」は、「繰り返し条件の設定」である。このとき、③「一つ一つの動きに対応した記号」は、本稿では示していないが、プログラムの実装(コードの記述)に相当する。

4. おわりに

本研究は、おもに塵劫記に含まれる題材を中心として、和算と数学との関係に加えてプログラミング的思考との関係性についても考察した。それぞれの解法の原理は異なるが、和算や数学による解法を併せて理解することは、プログラミングによる解法における境界条件の設定に参考になると考えられる。

参考文献

- [1] 吉田光由、「塵劫記」、岩波書店(1977).
- [2] 中央教育研究所、「小学校算数における和算の活用」, 研究報告, no.74 (2011).
- [3] 鈴木将史、「和算流による算数・数学教育改革の試み」、創価大学教育学論集、第72号、pp.71-87 (2020).
- [4] 橋本由美子、「和算を算数教育に取り入れる意義とその教材化—『塵劫記』等に焦点をあてて—」、浦和論叢、43、pp.87-101 (2010).
- [5] 坪井健司、愛木豊彦、「スーパーサイエンスハイスクール講座「自然の中の数列」実践報告」、岐阜数学教育研究、Vol.2, pp.116-127 (2003).
- [6] 栗原秀幸、「プログラミングの教育と数学の思考」、数学教育学会研究紀要、vol. 29、No.3・4 (1988).
- [7] 高山琢磨、「数学教育におけるプログラミング的思考に関する一考察」、日本科学教育学会年会論文集、42、p. 517-518 (2018).
- [8] 赤堀侃司、「プログラミング教育における論理的思考とは何か」、学習情報研究論文誌、第261巻、第4号、pp.56-61 (2018).
- [9] Sentance, S., & Csizmadia, A., "Teachers' perspectives on successful strategies for teaching Computing in school", In IFIP TC3 Working Conference 2015: A New Culture of Learning: Computing and Next Generations, pp.1-10 (2015).