

若年層を対象とした情報活用能力の学習到達段階を評価するための ルーブリックの提案

齋藤大輔¹ 海江田祥汰¹ 鷺崎弘宜¹ 深澤良彰¹
大宮秀利² 佐藤衣津美² 小野寺美咲²

概要：若年層を対象とした情報活用能力を体系的に評価することが重要である。情報活用能力の評価方法として、評価基準や規準が多く提案されている。課題として、これら多くの評価基準や規準は評価項目が段階的でないため、評価者の違いによって差が生じてしまい、主観的な評価となってしまう。本研究では、この課題を解決するため、情報活用能力を評価するためのルーブリックを提案する。提案するルーブリックは既存の情報活用能力等を評価するための基準や規準の評価項目等を参考にし、評価の観点と5段階の学習到達目標を設定した。本論文では、提案したルーブリックについて、設計方法や既存指標との対応等の詳細を示す。本ルーブリックを用いることで若年層における情報活用能力の客観的な評価を容易にでき、体系化を図ることを期待できる。また、今後、小学校プログラミング教育の手引きA分類の小学校5年生の算数「正多角形」、6年生の理科「身の回りの電気の性質や働き」の授業等に適用し妥当性と有効性を検証する。

キーワード：情報活用能力, プログラミング教育, ルーブリック

Proposed Rubric for Evaluating Young People's Information Utilization Ability

DAISUKE SAITO^{†1} SHOTA KAIEDA^{†1}
HIRONORI WASHIZAKI^{†1} YOSHIAKI FUKAZAWA^{†1}
HIDETOSHI OMIYA^{†2} IZUMI SATO^{†2}
MISAKI ONODERA^{†2}

Abstract: Systematically evaluating young people's ability to utilize information is important, and various evaluation standards and criteria have been proposed for this purpose. However, many such standards and criteria do not include graded evaluation items. Therefore, evaluation outcomes may differ owing to differences in evaluators, and such differences may result in subjective evaluations. In this research, to address this problem, we propose a rubric to evaluate information utilization ability. The proposed rubric refers to existing evaluation standards and criteria and sets evaluation perspectives and five-step learning goals. In this paper, the design methodology of the proposed rubric and its correspondence to an existing index are shown. The proposed rubric provides a way to easily and systematically evaluate the information utilization ability of young people. In the future, to verify validity and effectiveness, we will apply the proposed rubric to fifth grade "regular polygons" and sixth grade science "personal electric properties and work" categories in elementary school programming education guidebook A.

Keywords: Information Utilization Ability, Programming Education, Rubric

1. はじめに

若年層を対象とした情報活用能力を評価することは、重要である。若年層の情報活用能力を育成するための代表的な例としてプログラミング教育が挙げられる。加えて、コンピュータを活用した創造性やコミュニケーション力を養う教育や、コンピュータを正しく活用するための情報倫理教育などが挙げられる。これらの教育における評価基準や規準は多く提案されている。課題として評価項目が段階的ではないため、評価者の違いなどによって、評価に差が生じてしまう。また、教育領域によって評価基準や規準が異なってしまう、評価の一貫性が保てない。本研究では、前

述した課題を解決するため、若年層を対象としたプログラミング教育やコンピュータサイエンス教育に焦点を当てた、情報活用能力を客観的に評価するためのルーブリックを提案する。本研究における研究課題(RQ)は以下の通りである。

- RQ1: 多くの評価基準・規準の評価項目を共通化、段階化にできるか？
- RQ2: 情報活用能力について多くの評価観点を含む統一化された評価指標を作成できるか？

RQ1を解決することによって、評価項目が共通化、段階化され客観的な評価が可能になる。加えて、新たなルーブリックの提案において、評価項目の選定は非常に重要である。本研究では、多くの基準や規準を参照し、評価項目を共通化、段階化にできるか検討する。この共通化、段階化によって、評価項目が統一化され客観的な評価が可能なルー

¹ 早稲田大学

Waseda University

² 富士電機 IT ソリューションズ株式会社

Fuji Electric IT Solutions Co.,Ltd.

ブリックになる。また、RQ2 を解決することによって、多くの情報活用能力を評価可能にすることができる。何故ならば、情報活用能力は1つや2つではない。このため、多くの評価観点を洗い出し、それらを整理することが重要である。

本論文は、本章で研究の概要と課題を示した。2章では本研究の背景を示す。3章では、提案したループブリックについて示し、4章で提案したループブリックについて考察する。5章では本研究のまとめと今後の研究について示す。

2. 背景

2.1 日本のプログラミング教育分類

情報活用能力の一つとしてプログラミングの活用があげられる。児童を対象としたプログラミング教育も実施されている。文部科学省は小学校段階での学習指導要領にてプログラミング教育は「学習の基盤となる資質・能力」と位置付けている[1]。加えて、インターネットを通じて配布されている「プログラミング教育の手引(第二版)」では、情報活用能力」の育成や情報手段 (ICT) を「適切に活用した学習活動の充実」を進めるうえでプログラミング教育を適切に位置づける必要があるとしている[2]。このことから、プログラミング教育が情報活用能力の育成に使用されることは明らかである。また、同手引上で、プログラミング教育として、6分類に分割している(表 1)。これらの分類で実施されるプログラミング教育の内容には差があり、育成される能力にも差が生じる。故に、各分類について、それぞれ評価する必要がある。しかし、都度、各分類に対応した評価を作成することは手間が掛かる。また、評価の一貫性が保てなくなってしまう、主観的評価となってしまう。故に、6分類で実施されるであろう、プログラミング教育の評価について、統一化された評価基準や規準で評価することが重要である。

表 1, プログラミング教育の 6 分類[2]

分類	概要
A	学習指導要領に例示されている単元等で実施されるもの
B	学習指導要領に例示されていないが、学習指導要領に示される各教科等の内容を指導する中で実施するもの
C	教育課程内で各教科とは別に実施するもの
D	クラブ活動など、特定の児童を対象として、教育課程内で実施するもの
E	学校を会場とするが、教育課程外のもの
F	学校外でのプログラミング学習機会

2.2 コンピュータサイエンスの基準

情報活用能力を養う一つとして諸外国で実施されてい

るコンピュータサイエンス教育が挙げられる。この、コンピュータサイエンス教育の評価基準として CSTA K-12 Computer Science Standards がある[3]。この基準は、K-12 におけるコンピュータサイエンスのカリキュラム設計のための学習目標を提供している。

加えて、ISTE Standards for Students がある[4]。この基準は、絶え間なく進化する技術環境において、学習が学生主導の流れとなるように設計されている。技術の利用、情報倫理、協力や創造性などの項目が設定されている。

これらの基準の課題は学習目標の設定が詳細に分類されていないことである。例えば、CSTA K-12 Computer Science Standards における学習目標として以下のように設定されている。

- シーケンスと単純なループを含むアルゴリズムまたはプログラムのエラーをデバッグ (識別および修正) ができる(1A-AP-14 プログラムの開発)[3]

この学習目標の中には、単純なアルゴリズムの理解やデバッグ方法の理解など多くの評価観点が含まれている。故に学習者がこの目標を達成したかの評価が困難である。また、評価者によって評価に差が生じてしまい、主観的な評価となってしまう。

したがって、本研究ではこれらの目標を分割し、段階化が可能か検討した。

2.3 国内における評価基準・規準

先行研究として多くの基準や規準がある。小田らが「デジタル・情報活用評価規準」の開発に取り組んでいる[5]。この研究で取り組まれている規準は4分野、6段階の評価規準の開発である。また、小林らが提案した小学校におけるプログラミング教育の評価規準もある[6]。この評価規準は抽象化や一般化等の評価項目を持っている。また各項目の評価は学年で段階分けされている。これら評価規準の課題は、評価段階のレベルが学年で段階分けされたことである。故に、学年における評価の段階でしか評価できず、学習者の到達段階を詳細に把握することができなくなってしまう。

加えて、西野らが提案したプログラミング教育評価用の記述式ループブリックがある[7]。このループブリックの課題はプログラミング学習環境の Scratch に依存していることである。故に、Scratch 以外の環境で評価しようとしたとき、ループブリックの変更が生じてしまう。

これらのことから、我々が提案するループブリックは、学年ごとの段階分けではなく、評価項目ごとの段階分けとし、学習環境に依存しない形で設計した。

2.4 学習の到達段階分類

学習に関する分類法として、ブルームのタキソノミーや SOLO タキソノミーなどがある。これらのタキソノミーは評価基準や規準を作成するときに有用である。

2.4.1 SOLO タキソノミー

SOLO タキソノミー (Structure of the Observed Learning

Outcome)[8]は、教育学における分類法で複雑さの観点から学習成果を分類する。いくつかの研究においてプログラミング教育に対して SOLO タキソノミーが適用され有効性が実証されている[9][10]。加えて、SOLO タキソノミーは、最終的に学習される内容に関する、事前の知識、動機、および思考方法を分類し、5 段階に分けている。分類される 5 段階は以下の通りである。

- Pre-structural：学習者は、学習内容について何も理解していない。
- Uni-structural：学習者は、学習内容について 1 つの側面を理解している。
- Multi-structural：学習者は、独立した状態で学習内容について複数の側面を理解している。
- Relational：学習者は、学習内容について複数の側面間の関係と構造の組み合わせを理解している。
- Extended abstract：学習者は、一般化とさまざまな視点から学習内容について理解し、新しいアイデアを作成できる。

本研究においてルーブリックの学習到達段階の設定の参考として利用した。

2.4.2 ブルームのタキソノミー

ブルームのタキソノミーは、教育学において有名な分類法である。この分類法は、知識、理解、応用、分析、統合、評価の 6 つの認知段階を定義している[11]。しかし、アンダーソンらによって記憶、理解、応用、分析、評価、創造[12]に改訂された。加えて、ブルームの改訂版タキソノミーは、プログラミング教育にも広く使用されている[13][14]。本研究では、ブルーム分類法を学習目標の設定する際の参考として使用した。

3. ルーブリックの提案

本研究では小学校から高校までの若年層を対象とした情報活用能力の学習到達段階を評価するためのルーブリックを提案する。提案したルーブリックはいくつかの評価基準や規準を参考として評価観点を決定した。また、本ルーブリックはプログラミング教育やコンピュータサイエンス教育に焦点を当てている。参考とした基準や規準の例は以下の通りである。

- CSTA K-12 Computer Science Standards [3]
- ISTE Standards for Students [4]
- Rubric for Programming Education for Elementary School Student (Rubric ProEEs) [15][16]
- 第 2 版「プログラミングで育成する資質・能力の評価規準（試行版）」Ver.2.0.0 [17]

我々が提案するルーブリックの一部と評価観点を付録 1 から付録 4 に示す。

3.1 ルーブリックの評価観点について

我々が提案したルーブリックの評価観点は、大項目とし

て、「プログラミング的思考力」、「プログラミング技術力」、「コンピュータの使用と社会での関り」の 3 つに分類した。何故ならば、プログラミング的思考力とプログラミング技術力では、思考と技術の差がある。加えて、プログラミング的思考力とプログラミング技術力を使用し、どのように社会と関わるかの力も重要であるため、これら 3 つの大項目とした。また、それぞれの大項目には中項目と小項目を定め評価観点(付録 1)を設定した。

3.1.1 プログラミング的思考力

この大項目は、順次や繰り返しといったアルゴリズム的な考え方や抽象化、推論といった論理的思考力についての評価観点を含んだ項目である。中項目として「アルゴリズムの理解 (アルゴリズムの考え方)」、「プログラムの設計・作成における思考力」、「プログラムの理解 (読解・編集・評価)」の 3 項目である。

「アルゴリズムの理解 (アルゴリズムの考え方)」の小項目として「順次」、「分岐」、「反復」、「変数」、「配列」、「関数」、「再帰」、「並び替え (ソート)」の 8 項目を定めた。

「プログラムの設計・作成における思考力」の小項目として「問題の細分化」、「事象の分析」、「動作の抽出」、「動作の構築」、「関数化」、「一般化」、「抽象化」、「推論」、「論理代数」、「演算子」の 10 項目を定めた。

「プログラムの理解 (読解・編集・評価)」の小項目として「プログラムの理解」の 1 項目を定めた。

故に、この大項目の小項目は合計で 19 項目である。

3.1.2 プログラミング技術力

プログラミング技術力は、アルゴリズムの使用やプログラムの設計等のプログラムを作るための力についての評価観点を含んだ項目である。中項目として「アルゴリズムの使用」、「プログラムの設計・作成」、「プログラムの理解・編集・評価」の 3 項目である。

「アルゴリズムの使用」の小項目として「各要素を用いたプログラムの作成」の 1 項目を定めた。

「プログラムの設計・作成」の小項目として「設計書の作成」、「論理代数の使用」、「演算子の使用」、「プログラミング用ソフトウェアの使用」、「プログラミング言語の使用」、「思考の表現」の 6 項目を定めた。

「プログラムの理解・編集・評価」の小項目として「デバッグ」、「テスト」、「モディファイ」、「リファクタリング」の 4 項目を定めた。

故に、この大項目の小項目は合計で 11 項目である。

3.1.3 コンピュータの使用と社会での関り

この大項目はコンピュータについての学びの意欲や情報倫理、コンピュータを使った問題解決能力などの評価観点を含んだ項目である。中項目として「態度」、「情報倫理」、「協力・発表」、「コンピュータの仕組みの理解」、「コンピュータの使用」の 5 項目を定めた。

「態度」の小項目として「興味」、「関心」、「意欲」の3項目を定めた。

「情報倫理」の小項目はセヴァーソンの四つの原則[18]も参考に「セキュリティ」、「知的財産権」、「プライバシーの尊重」、「公正な情報」、「危害を与えない」の4項目を定めた。

「協力・発表」の小項目は「他者の発表理解」、「協力」、「発言」、「自分のアイデアの発表」、「発表資料の作成」の5項目を定めた。

「コンピュータの仕組の理解」の小項目は「中央処理装置」、「主記憶装置」、「補助記憶装置」、「入力装置」、「出力装置」の5項目を定めた。

「コンピュータの使用」の小項目は「基本操作」、「問題解決」、「創造力」の3項目を定めた。

故に、この大項目の小項目は合計21項目である。

3.2 ルーブリックの学習目標について

我々はルーブリックの学習目標として、一つの評価観点に対して、SOLO タクソノミーに基づき5段階で設定した。故に、5段階は表2のように分類した。また、学習目標の例として図1を示す。

表2. ルーブリックにおける到達段階

段階	到達の状態
1	ある観点について何も知らない. これから勉強を始めようとする状態
2	ある観点について一つの要素を理解している状態
3	ある観点について複数の要素を理解している状態
4	ある観点について複数の要素を関連させることができる状態
5	ある観点について別の観点などにも応用させることができる状態

中項目	小項目	5	4	3	2	1
プログラムの理解 (読解・編集・評価)	プログラムの理解	手順や動作の読解、評価、編集を理解し、問題解決や他の事柄に活用できる	手順や動作の読解、評価、編集を関連付けできる	手順や動作の読解、評価、編集がそれぞれできる	手順や動作を読むことができる	プログラムを理解できない

図1. 学習目標の例

3.3 既存指標との対応付け

我々はルーブリックを提案するにあたり、既存の基準や規準と対応付けを実施した。例えば、我々はルーブリックの対応として、CSTA K-12 Computer Science Standards の評価項目 1B-AP-10 であればプログラミング技術力の「各要素を用いたプログラムの作成 (アルゴリズムの使用)」に対応付けし「プログラミングで育成する資質・能力の評価規準 (試行版)」の目標の「動きに分ける」であれば、プログラミング的思考力の「動作の抽出 (プログラムの設計・作成における思考力)」に対応付けした。このように、相互関

係を保つことによって、既存の基準や規準も併せて参照し評価を実施することができる。故に、より客観性の高い評価が可能となる。

4. 考察

本研究では研究課題として「RQ1: 多くの評価基準・規準の評価項目を共通化、段階化にできるか?」と「RQ2: 情報活用能力について多くの評価観点を含む統一化された評価指標を作成できるか?」の2つを調査した。

先ず、「RQ1: 多くの評価基準・規準の評価項目を共通化、段階化にできるか?」の回答であるが、共通化、段階化をすることは可能である。例えば、CSTA K-12 Computer Standards のような基準は学習目標が広く設定されており、多くの観点を含んでいる。故に、目標を整理し観点を洗い出すことで、他の基準や規準の評価項目と共通化可能である。また、発達段階や学習段階の分類法を用いることで、評価項目を段階化することができる。

次に「RQ2: 情報活用能力について多くの評価観点を含む統一化された評価指標を作成できるか?」についての回答であるが、多くの評価観点を含む評価指標は作成可能である。我々は、評価指標をルーブリックという形で提案した。提案したルーブリックの評価観点は全体で51項目である。したがって、既存基準や規準を整理することにより、学習観点を多く含むことが十分に可能であると示された。

5. まとめと今後について

本研究では情報活用能力の到達段階を評価するルーブリックを提案した。提案したルーブリックはプログラミング教育、コンピュータサイエンス教育に焦点を当て、既存の基準や規準の評価観点等を参照し作成した。結果として、5段階評価の51項目の評価観点を設定した。また、本ルーブリックを用いることで若年層における情報活用能力の個別特性を可視化やプログラミング教育を中心とした情報活用能力の客観的な評価を容易にでき、体系化を図ることを期待できる。

5.1 今後について

本研究ではルーブリックの提案のみをした。今後は小学校プログラミング教育の手引きA分類の小学校5年生の算数「正多角形」、6年生の理科「身の回りの電気の性質や働き」の授業等に適用し有効性と妥当性の検証をする。加えて、A分類のみではなくB分類からF分類までについての検証や中高生向けの情報活用教育の授業等に適用していく予定である。

5.2 制限事項

本ルーブリックには次のような制限事項がある。

- 提案されたルーブリックは開発段階であり、評価項目や観点が変更される可能性がある
- 提案されたルーブリックは、実際の授業等に適用し

検証されていない。

上述の制限事項により、提案したルーブリックには有効性や妥当性に課題を残している。これらの、制限事項は、今後、多くの授業等に適用し、ルーブリックそのものを評価、検証し解決する。

参考文献

- [1] “小学校学習指導要領（平成 29 年告示）”。
https://www.mext.go.jp/content/1413522_001.pdf, (参照 2019-11-01).
- [2] “小学校プログラミング教育の手引（第二版）”。
https://www.mext.go.jp/component/a_menu/education/micro_detail/_icsFiles/afieldfile/2018/11/06/1403162_02_1.pdf, (参照 2019-10-30).
- [3] “CSTA K-12 Computer Science Standards”。
<https://drive.google.com/file/d/1-dPTAI1yk2HYPKUWZ6DqaM6aVUDa9iby/view>, (参照 2019-10-30).
- [4] “ISTE Standards for Students | ISTE”。
<https://www.iste.org/standards/for-students>, (参照 2019-11-06).
- [5] 小田 理代, 赤堀 侃司, 森本 康彦, 稲垣 忠, 兼宗 進, 島袋 舞子, 長瀧 寛之, 鷺崎 弘宜, 齋藤 大輔, 西端 律子, 福田 大年, 渡辺 美智子, 竹内 光悦, 玉田 和恵, 牧野 直道, 菅崎 直子. デジタル・情報活用評価規準の開発に向けて. 日本 STEM 教育学会 第 2 回年次大会 一般研究発表予稿集, p. 38-39.
- [6] 小林未歩, 宇都宮 晃, 宮澤豪臣, 福島健介. 授業実践に基づく小学校プログラミング教育「評価規準」の提案 - 授業における評価規準の必要性を踏まえて -. 2018 PC Conference, p. 257-260.
- [7] 西野 和典, 田中 太志朗, 近藤 秀樹, 山口 真之介*1, 大西 淑雅. プログラミング的思考を評価するルーブリックの作成とその試用. 教育システム情報学会 JSiSE2018 第 43 回全国大会, p. 141-142
- [8] Biggs John B, and Kevin F. Collis. Evaluating the quality of learning: The SOLO taxonomy (Structure of the Observed Learning Outcome). Academic Press, 2014.
- [9] Raymond Lister, Beth Simon, Errol Thompson, Jacqueline L. Whalley, and Christine Prasad. Not seeing the forest for the trees: novice programmers and the SOLO taxonomy. Proceedings of the 11th annual SIGCSE conference on Innovation and technology in computer science education, ACM, 2006, p. 118-122.
- [10] Jacqueline L. Whalley, Raymond Lister, Errol Thompson, Tony Clear, Phil Robbins, P. K. Ajith Kumar, and Christine Prasad. An Australasian study of reading and comprehension skills in novice programmers, using the bloom and SOLO taxonomies. Proceedings of the 8th Australasian Conference on Computing Education-Volume 58, Australian Computer Society, Inc., 2006, p. 243-252.
- [11] Bloom BS. Taxonomy of educational objectives. Vol. 1: Cognitive domain. New York: McKay, 1956, p. 20-24.
- [12] Krathwohl, D. R., and Anderson, L. W. A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives. Longman, 2009.
- [13] Errol Thompson, Andrew Luxton-Reilly, Jacqueline L. Whalley, Minjie Hu, and Phil Robbins. Bloom's taxonomy for CS assessment. Proceedings of the tenth conference on Australasian computing education-Volume 78, Australian Computer Society, Inc., 2008, p. 155-161.
- [14] Cynthia C. Selby. Relationships: computational thinking, pedagogy of programming, and Bloom's Taxonomy. Proceedings of the Workshop in Primary and Secondary Computing Education, ACM, 2015, p. 80-87.
- [15] 齋藤 大輔, 佐々木 綾奈, 鷺崎 弘宜, 深澤 良彰, 武藤 優介, 田村 麻里子, 西澤 利治. 小学生を対象としたプログラミング教育のためのルーブリックの提案. STEM 教育研究, 2018, vol. 1, 2018, p. 41-51.
- [16] “PLR for Elementary School Students ver1.1(最終更新日: 2017/11/17)”。
http://g7programming.jp/wp-content/uploads/2018/02/RubricProE-Es_171106.pdf, (参照 2019-11-29).
- [17] “第 2 版「プログラミングで育成する資質・能力の評価規準（試行版）」 Ver.2.0.0（2018/8/31 版）”。
<https://beneprog.com/wp-content/uploads/2018/08/ver2.0.0.pdf>, (参照 2019-10-30).
- [18] Richard J. Severson. The Principles of Information Ethics. M.E. Sharpe, 1997.

付録

付録 1. ルーブリックの評価観点

プログラミング的思考力		プログラミング技術力		コンピュータの使用と社会での関り	
大項目	小項目	大項目	小項目	大項目	小項目
アルゴリズムの理解（アルゴリズムの考え方）	順次	アルゴリズムの使用	各要素を用いたプログラムの作成	態度	興味
	分岐		設計書の作成		関心
	反復		論理代数の使用	情報倫理	意欲
	変数	プログラムの設計・作成	演算子の使用		セキュリティ
	配列		プログラミング用ソフトウェアの使用		知的財産権
	関数		プログラミング言語の使用		プライバシーの尊重
プログラムの設計・作成における思考力	再帰	プログラムの読解・編集・評価	思考の表現	テスト	公正な情報
	並び替え(ソート)		デバッグ		危害を与えない
	問題の細分化		テスト	協力・発表	他者の発表理解
	事象の分析		モディファイ		協力
	動作の抽出	プログラムの設計・作成における思考力	リファクタリング		発言
	動作の構築			コンピュータの仕組みの理解	自分のアイデアの発表
	関数化				発表資料の作成
	一般化				中央処理装置
	抽象化				主記憶装置
	推論				補助記憶装置
プログラムの理解（読解・編集・評価）	論理代数			コンピュータの使用	入力装置
	演算子				出力装置
	プログラムの理解				基本操作
					問題解決
					創造力

付録2. プログラミング的思考力の評価項目(一部)

中項目	小項目	5	4	3	2	1
アルゴリズムの理解 (アルゴリズムの考え方)	順次	順次の考え方を理解し他の考え方と結び付けることができる	複数の事柄を順序だてて、結び付けて考えることができる	複数の事柄をそれぞれ順序だてることができる	一つの事柄について順序だてることができる	順次の考え方を理解していない
	分岐	分岐の考え方を他の考え方と結びつけることができる	ある事柄について複数の条件を考え、組み合わせで分岐することができる	ある事柄について複数の条件で分岐することができる	ある事柄について一つの条件で分岐することができる	分岐の考え方を理解していない
	反復	反復の考え方を他の考え方と結びつけることができる	複数の反復を組み合わせて使うことができる(二重ループ)	ある手順から複数の反復に気づくことができる	ある手順から1つの反復に気づくことができる	反復の考え方を理解していない
	変数	変数の考え方を他の考え方と結びつけることができる	ある事柄について関連した複数の変数を作ることができる	ある事柄について複数の変数を作ることができる	ある事柄について一つの変数を作ることができる	変数の考え方を理解していない
	配列	配列の考え方を他の考え方と結びつけることができる	いくつかの要素を要素と関連した羅列にできる	いくつかの要素を複数の別々の羅列にできる	いくつかの要素を一つの羅列にできる	配列の考え方を理解していない
	関数	関数の考え方を他の考え方と結びつけることができる	いくつかの手順を複数の要素にそれぞれ関連した形でまとめられる	いくつかの手順を複数の要素にまとめられる	いくつかの手順を一つの要素にまとめられる	関数の考え方を理解していない
	再帰	再帰の考え方を他の考え方と結びつけることができる	再起できる関数とできない関数に気づき、再帰できる関数については再帰を使う	ある関数について、関数内で関数自身を呼び出すことができる	ある関数について、関数内で関数自身を呼び出せることに気づける	再帰の考え方を理解していない
	並び替え(ソート)	並び替えの考え方を他の考え方と結びつけることができる	いくつかの要素を要素に関連した最適な方法で並び替えることができる	いくつかの要素を複数の違う方法でそれぞれ並び替えることができる	いくつかの要素を一つの方法で並び替えることができる	並び替えの考え方を理解していない

付録3. プログラミング技術力の評価項目(一部)

中項目	小項目	5	4	3	2	1
アルゴリズムの使用	各要素を用いたプログラムの作成	各要素を用いてプログラムを作成でき、目的や問題解決のために活用することができる	複数の考え方(アルゴリズム)を結び付けてプログラムを作成できる	それぞれ、複数の考え方(アルゴリズム)でプログラミングを作成できる	順次の考えを用いてプログラムを作成できる	プログラムを作成できない
プログラムの設計・作成	設計書の作成	設計書を作成でき、その設計書に沿ってプログラムを書ける	文字および図を関連させてプログラムの設計書が作成できる	文字および図を用いてプログラムの設計書が作成できる	文字もしくは図を用いてプログラムの設計書が作成できる	設計書作成できない
	論理代数の使用	論理代数を使い、アルゴリズムやソフトウェアを開発できる	複数の論理代数を組み合わせてプログラムで使える	複数の論理代数をプログラムで使える	いずれかの論理代数をプログラムに使える	論理代数をプログラムで使えない
	演算子の使用	演算子を使い、アルゴリズムやソフトウェアを開発できる	複数の演算子を組み合わせてプログラムで使える	複数の演算子をプログラムで使える	いずれかの演算子をプログラムに使える	演算子を使用してプログラムが書けない
	プログラミング用ソフトウェアの使用	プログラミング用のソフトウェアを用いてソフトウェアを開発することができる	やりたいことに応じて、適したプログラミング用のソフトウェアを使うことができる	複数のプログラミング用のソフトウェアを使うことができる	一つのプログラミング用のソフトウェアを使うことができる	プログラミング用のソフトウェアが使えない
	プログラミング言語の使用	プログラミング言語を用いてソフトウェアを開発することができる	やりたいことに応じて、適したプログラミング言語を選択して使用できる	複数のプログラミング言語を使用できる	一つのプログラミング言語を使用できる	プログラミング言語を使用できない
	思考の表現	思考を表現でき、その表現を活用してプログラムを書ける	文字および図を関連させて自身の思考を表現できる	文字および図を用いて自身の思考を表現できる	文字もしくは図を用いて自身の思考を表現できる	自身の思考をプログラミングを用いて表現できない

付録4. コンピュータの使用と社会での関りの評価項目(一部)

中項目	小項目	5	4	3	2	1
態度	興味	ある項目について凄く興味があり、自分から発展させようとしている	ある項目について凄く興味がある	ある項目について興味がある	ある項目について少し興味がある	全く興味がない
	関心	ある項目について凄く関心があり、自分から発展させようとしている	ある項目について凄く関心がある	ある項目について関心がある	ある項目について少し関心がある	全く関心がない
	意欲	ある項目について凄く意欲があり、自分から発展させようとしている	ある項目について凄く意欲がある	ある項目について意欲がある	ある項目について少し意欲がある	全く意欲がない
情報倫理	セキュリティ	セキュリティについて理解し、自身の活動や作品作りに活用できる	セキュリティの要素について複数理解し、それらについて対応や対策方法を知っている	セキュリティの要素について複数、理解している	セキュリティの要素について1つ理解している(例えば、コンピュータウィルスの脅威とか)	セキュリティを理解していない
	知的財産権	知的財産権を理解し、自身の活動や作品作りに活用できる	各知的財産権を結びつけることができる。(商標権と著作権のように)	著作権、商標権、特許権、その他(意匠、種苗権など)を複数、理解している	著作権、商標権、特許権、その他(意匠、種苗権など)のいずれかを理解している(インターネット上のイラストを勝手に使わないなど)	知的財産権を理解していない
	プライバシーの尊重	プライバシーを尊重でき、自身の活動や社会の中で実践できる	プライバシーを侵害することによるリスクと結びつけて、プライバシーを尊重できる	プライバシー尊重について理解し実践できる	プライバシーを尊重について理解している	プライバシーの尊重ができない
	公正な情報	自分の活動、社会活動の中において、公正な情報を分析、展開できる	ある情報について公正な情報が確認できる。また、正しく伝えることができる。	公正な情報かどうか確かめられる。	公正な情報かどうか確かめることができる	公正な情報について理解していない
	危害を与えない	危害を与えない発言、行動ができ、自身の活動や社会の中での活動で実践できる	危害を与えない発言や行動で起き得るリスクと結びつけて、危害を与えない発言、行動ができる	危害を与えない発言、行動を理解し、実践できる	危害を与えない発言、行動などを理解している	危害を与えない方法を理解していない