

TAOにおけるPCIモジュール(CBP)の採点方法の提案

日本大学文理学部情報科学科
谷研究室 篠遠孝

日本大学文理学部情報科学科卒業演習発表会
2024年2月7日

目次

I. はじめに

I. 概要

II. 準備

I. 日本の情報教育の現状

II. PBT・CBTとは

III. TAOとは

IV. CBPとは

V. 演習の目的

III. 提案

I. CBPの採点方法

II. 提案採点方法

III. 今後の課題

I.1 概要

日本の情報教育の現状

平成29・30・31年に新学習指導要領が改訂された

小学校

プログラミング的思考を育成

中学校

「計測・制御」・「ネットワークを利用した双方向性のあるコンテンツ」のプログラミング学習

高等学校

共通必修科目「情報I」を新設

「情報I」の単元「コンピュータとプログラム」では

アルゴリズムを表現しプログラミングによってコンピュータや情報通信ネットワークの機能を使う方法や技能を身に付ける

I.I 概要

日本の情報教育の現状

2025年度より大学入学共通テストにおいて、プログラミングに関する問題を含む教科「情報」が新設される

2025年度の出題形式

➤ 多肢選択式

➤ 数値や記号等で解答する形式

```
(1) Kouka = [1,5,10,50,100]
(2) kingaku = 46
(3) maisu = 0, nokori = kingaku
(4) i を  ながら繰り返す:
(5) | maisu =  + 
(6) | nokori = 
(7) 表示する(maisu)
```

図1 目標の金額ちょうどになる最小の硬貨枚数を計算するプログラム

・ の解答群

① $46 \div 10 + 1$	④ $46 \% 10 - 1$
② $46 \div 10$	③ $46 \% 10$

の解答群

① 5から1まで1ずつ減らし	④ 4から0まで1ずつ減らし
② 0から4まで1ずつ増やし	③ 1から5まで1ずつ増やし

の解答群

① 1	② maisu	③ i	④ nokori
-----	---------	-----	----------

・ の解答群

① $nokori \div Kouka[i]$	④ $nokori \% Kouka[i]$
② $maisu \div Kouka[i]$	③ $maisu \% Kouka[i]$

試作問題「情報I」

I.I 概要

演習の目的

プログラミングにおける重要な能力

- プログラム実行後の標準エラーの解読・デバッグ対処能力
- アルゴリズムの組み立て能力

大学入試センター (DNC) ではCBTでの出題を検討

- その一環として対話的にプログラミングができるテストのためのプログラミング問題PCI(CBP)モジュールを開発
- このCBPモジュールはプログラム実行環境を備えている

I.I 概要

演習の目的

DNCが開発したCBPモジュール

- プログラム実行環境を備えている
- 採点方法が不十分

CBPでは採点方法がソースコードの完全一致のみ

- 異なる実装方法を評価できない
- アルゴリズムの違いを評価できない

本演習では採点方法の模索を行う

- 完全一致ではない採点方法の模索
- 複数の解答プロセスがある場合の採点の模索

II.1 日本の情報教育の現状

平成29・30・31年に学習指導要領が改訂された

小・中・高等学校共通に

- 情報活用能力を言語能力と同等の「学習の基盤となる資質・能力」と位置付け
- 学校のICT環境整備とICTを活用した学校活動の充実

II.1 日本の情報教育の現状

小学校(2020年に実施)

- 文字入力など基本的な操作を習得
- プログラミング的思考の育成

中学校(2021年に実施)

- プログラミングの計測・制御
- ネットワークを利用した双方向性のあるコンテンツのプログラミング

高等学校(2022年に実施)

- 共通必修科目「情報I」を新設
- プログラミングに加えネットワークやデータベースの基礎等を学習

II.1 日本の情報教育の現状

2022年度の新学習指導要領では

高等学校の教科「情報」として

- 「情報I」を必修科目
- 「情報II」を選択科目

が新設された

II.1 日本の情報教育の現状

2025年度大学入学共通テストにて新教科としてプログラミング等の「情報I」を範囲とした問題が出題される予定になった

- 国立大学入試の多くが今後「5教科7科目」に情報科「情報I」を加えた「6教科8科目」を科すことを原則とする方針
- 私立大学入試の多くが今後「選択科目」に「情報I」を加える方針

II.1 日本の情報教育の現状

2025年度大学入学共通テストの新教科「情報」の範囲としてプログラミング問題が検討されている

試作問題「情報I」

多肢選択式または数値や記号等で解答する形式

```
(1) Kouka = [1,5,10,50,100]
(2) kingaku = 46
(3) maisu = 0, nokori = kingaku
(4) i を  キ ながら繰り返す:
(5) | maisu =  ク +  ケ
(6) | nokori =  コ
(7) 表示する (maisu)
```

図1 目標の金額ちょうどになる最小の硬貨枚数を計算するプログラム

オ · カ の解答群	
☐ ① $46 \div 10 + 1$	☐ ④ $46 \% 10 - 1$
☐ ② $46 \div 10$	☐ ③ $46 \% 10$

キ の解答群	
☐ ① 5から1まで1ずつ減らし	☐ ④ 4から0まで1ずつ減らし
☐ ② 0から4まで1ずつ増やし	☐ ③ 1から5まで1ずつ増やし

ク の解答群			
☐ ① 1	☐ ① maisu	☐ ② i	☐ ③ nokori

ケ · コ の解答群			
☐ ① $nokori \div Kouka[i]$	☐ ① $nokori \% Kouka[i]$		
☐ ② $maisu \div Kouka[i]$	☐ ③ $maisu \% Kouka[i]$		

大学入試センターではCBTでの出題も検討

II.II PBT・CBTとは

PBT（Paper Based Testing） CBT（Computer Based Testing）

```
(1) Kouka = [1,5,10,50,100]
(2) kingaku = 46
(3) maisu = 0, nokori = kingaku
(4) i を [キ] ながら繰り返す:
(5)   maisu = [ク] + [ケ]
(6)   nokori = [コ]
(7) 表示する(maisu)
```

図1 目標の金額ちょうどになる最小の硬貨枚数を計算するプログラム

オ・カの解答群

① $46 \div 10 + 1$	② $46 \% 10 - 1$
③ $46 \div 10$	④ $46 \% 10$

キの解答群

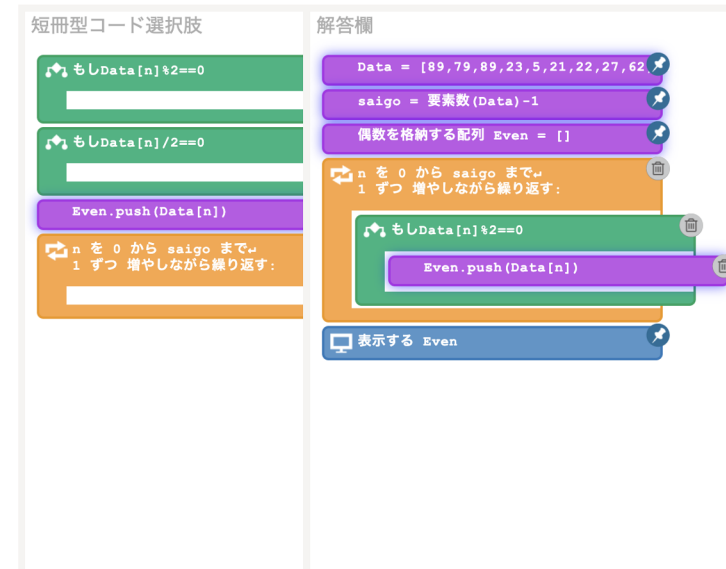
① 5から1まで1ずつ減らし	② 4から0まで1ずつ減らし
③ 0から4まで1ずつ増やし	④ 1から5まで1ずつ増やし

クの解答群

① 1	② maisu	③ i	④ nokori
-----	---------	-----	----------

ケ・コの解答群

① $nokori \div Kouka[i]$	② $nokori \% Kouka[i]$
③ $maisu \div Kouka[i]$	④ $maisu \% Kouka[i]$



紙を用いて行われる試験

大学入学共通テスト、
全国学力・学習状況調査など

コンピュータを用いて行われる試験

TOEFL、情報処理技術者試験など

II.II PBT・CBTとは

2025年度の大学入学者共通テスト「情報I」は PBT方式で行われる

プログラミング問題における大学入学共通テストでの問題点

- 紙上で表現できる問題形式でしか出題できない
- 多肢選択式の問題が主流であり解答に至る思考過程を評価することが困難

II.II PBT・CBTとは

2025年度大学入学共通テストからしばらくは、大学入試センターはPBT方式で大学入学共通テストを行う方針

プログラミング問題におけるCBT活用の検討

- プログラム実行後の標準エラーの解読・デバッグ対処能力
- アルゴリズムの組み立て能力

II.III TAOとは

大学入試センターはオープンソースのCBTプラットフォーム(TAO)で大学入学共通テストのCBT方式の開発へ

TAOとは

- コンピュータを用いて試験や受験者の回答を評価する
- LTI規格・QTI規格に準拠していてテストの作成・管理ができる

II.III TAOとは

2022年5月1日にQTIのVersion3.0が公開

Portable Custom Interactions(PCI)への対応

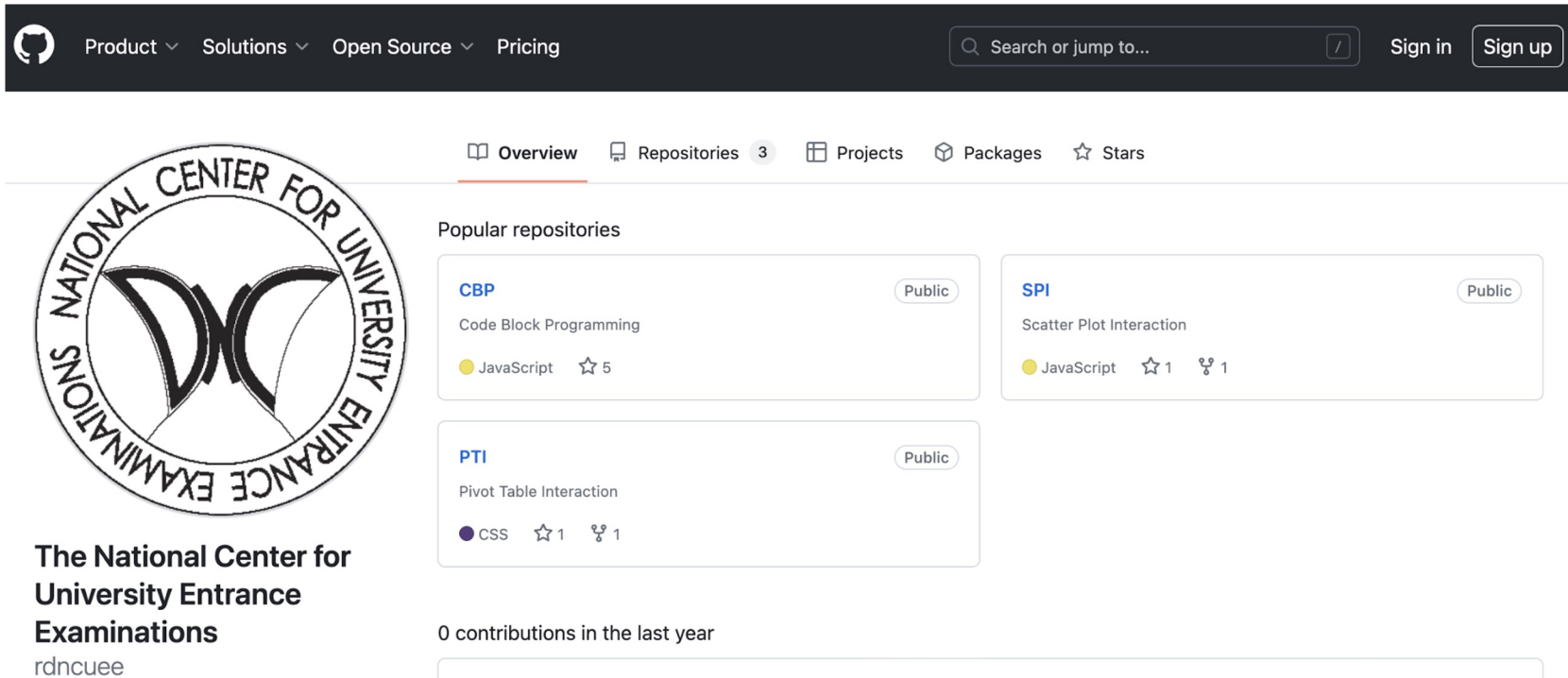


独自に作成した試験形式を取り入れることが可能

QTI(Question & Test Interoperability)規格 : LMSとテスト問題をつなぐインターフェース

II.IV CBPとは

2022年6月、大学入試センターはGitHub上に「プログラミング問題PCIモジュール(CBP)」を公開



The screenshot shows the GitHub profile page for 'rdncuee', which is The National Center for University Entrance Examinations. The profile includes a circular logo with the text 'NATIONAL CENTER FOR UNIVERSITY ENTRANCE EXAMINATIONS' and a stylized 'N' and 'C' inside. The bio reads 'The National Center for University Entrance Examinations' and 'rdncuee'. The page shows 3 repositories: 'CBP' (Code Block Programming, JavaScript, 5 stars), 'SPI' (Scatter Plot Interaction, JavaScript, 1 star, 1 fork), and 'PTI' (Pivot Table Interaction, CSS, 1 star, 1 fork). The 'Popular repositories' section is visible, and the 'Contributions' section shows 0 contributions in the last year.

Product ▾ Solutions ▾ Open Source ▾ Pricing

Search or jump to... / Sign in Sign up

Overview Repositories 3 Projects Packages Stars

Popular repositories

- CBP** (Public)
Code Block Programming
JavaScript ☆ 5
- SPI** (Public)
Scatter Plot Interaction
JavaScript ☆ 1 🍴 1
- PTI** (Public)
Pivot Table Interaction
CSS ☆ 1 🍴 1

0 contributions in the last year

II.IV CBPとは

プログラミング問題PCIモジュール(CBP)とは 指定した内容のプログラムを「短冊」の並び替えによって 適したプログラムにする問題形式

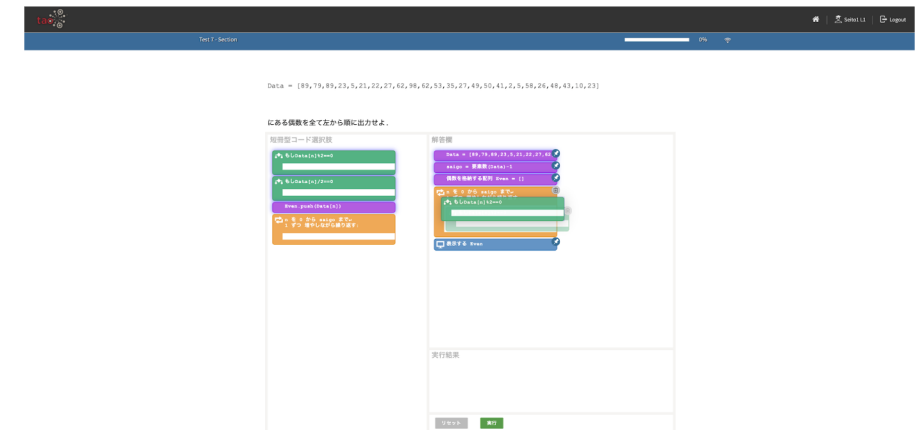
The screenshot displays the PCI module interface. At the top, there's a header with the 'ta' logo, a home icon, a user profile 'Seito111', and a 'Logout' button. Below the header, a progress bar shows 'Test 7 - Section' with a 0% completion indicator. The main content area contains a data array: `Data = [89, 79, 89, 23, 5, 21, 22, 27, 62, 98, 62, 53, 35, 27, 49, 50, 41, 2, 5, 58, 26, 48, 43, 10, 23]`. Below the array, the instruction reads: 'にある偶数を全て左から順に出力せよ。' (Output all even numbers from left to right in order). The interface is divided into two panels. The left panel, titled '短冊型コード選択肢' (Strip-type code selection), shows four code blocks: two green blocks with conditions `もしData[n]%2==0` and `もしData[n]/2==0`, a purple block with `Even.push(Data[n])`, and an orange block with a loop instruction. The right panel, titled '解答欄' (Answer area), shows a sequence of blocks: a data assignment, a variable `saigo` set to `偶数数(Data)-1`, an array `Even` initialized as `[]`, a loop block with the instruction 'n を 0 から saigo まで、1 ずつ 増やしながら繰り返す:', a green block with `もしData[n]%2==0`, a purple block with `Even.push(Data[n])`, and a blue block to '表示する Even'. At the bottom, there are buttons for 'End test' and 'Skip and end test'.

II.IV CBPとは

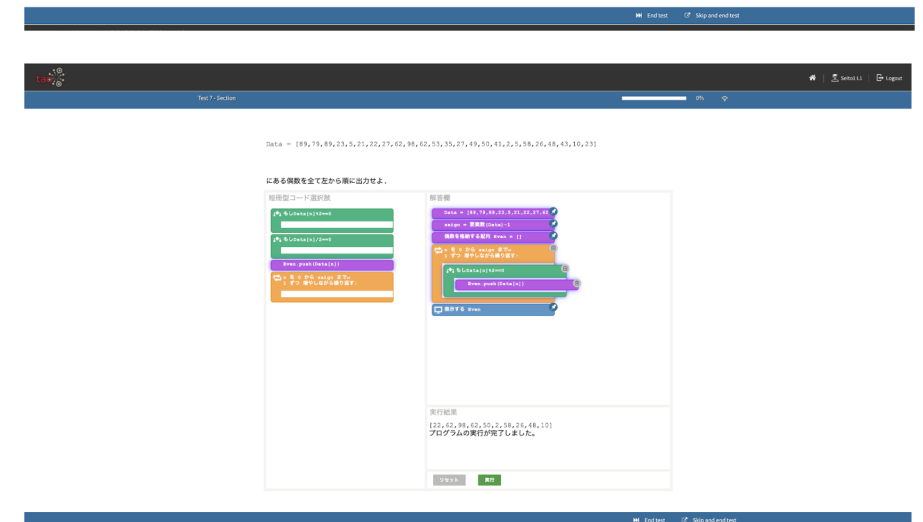
作成したCBP問題の受験手順

1. 短冊型コード選択肢にある短冊をドラッグ&ドロップで「解答欄」へ
2. 実行を押すと実行結果が出力される

1



2



II.V 演習の目的

プログラミング問題PCIモジュール(CBP)ではプログラム実行が可能なモジュールであり

- プログラム実行後の標準エラーの解読・デバッグ対処能力
- アルゴリズムの組み立て能力

が可能

II.V 演習の目的

CBPモジュールで作成された問題の採点方法は、現時点ではソースコードの完全一致



複数の解き方が生まれるプログラミング問題に、現時点の採点方法は適していないのでは？

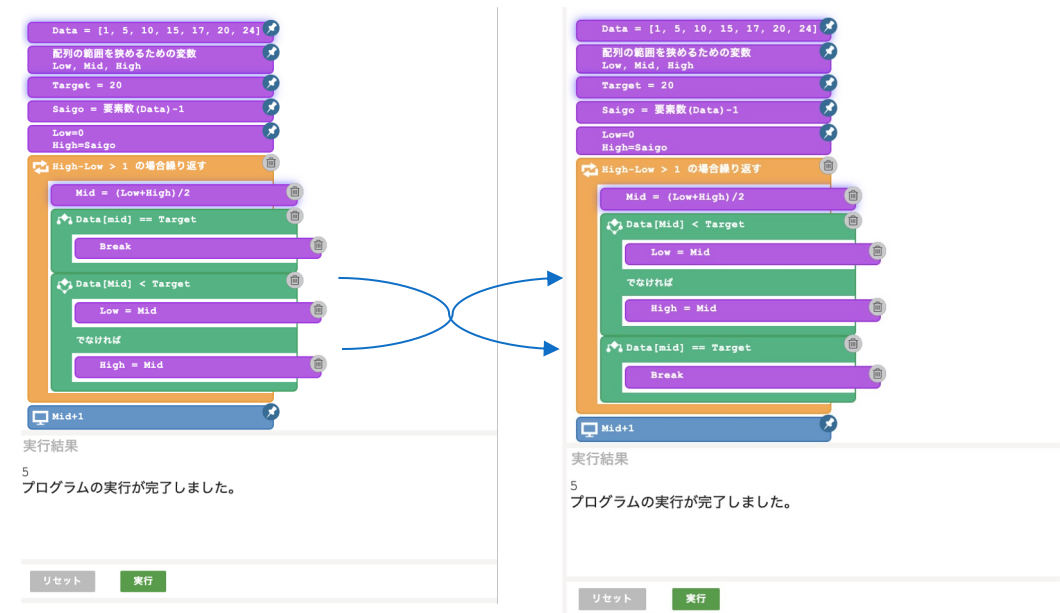
採点方法の提案

III.I CBPの採点方法

CBPモジュールの採点:
ソースコードを含む解答データの完全一致で判定

例) 処理の実行順が異なる場合

短冊型コードの解き方はどちらも正しいが
不正解となる



III.I CBPの採点方法

CBPモジュールの採点の問題点

異なる正しいアルゴリズムの採点を行えない

- アルゴリズムでも効率が異なる
- 正しいアルゴリズムでも複数考えられる

III. II 提案採点方法

採点方法の提案

提出されたソースコードの完全一致



提出されたソースコードに入力を行い、
出力の完全一致

III.II 提案採点方法

- テスト実行するためのテストデータを複数用意する
- 受験者が提出したソースコードを複数のテストデータを入力として実行し、それぞれの正解出力と比較することで採点する

III.III 今後の課題

- 提案手法の一部機能を実装するためCBPモジュールのソースコード解析等
- 提案手法の実装