

情報科入試と DNCL

東京理科大学 理学部第一部 中川 裕之
 数学科 教授 なか がわ ひろ ゆき

東京理科大学 理学部第一部 清水 克彦
 数学科 教授 しみず かつ ひこ

▶ DNCL の背景

2025 年の共通テストから出題加科目に「情報 I」(60 分)(100 点)が始まりました。2 単位の必修科目で 100 点は「お買い得科目」だと受験を勧めています。実は出題する側には問題があります。現行の「情報 I」の教科書を見ると、11 種あり、プログラム言語を前面に押し出しているものに実教出版の『高校情報 I Python』、『高校情報 I JavaScript』があり、表計算マクロや、ビジュアル型言語 Scratch など「コンピュータとプログラミング」の内容が執筆されているようです。さて、共通テスト「情報 I」でプログラミングの問題を出そうとします。実際の高校生は上に挙げた様々な言語のどれかしが学習していないのでどうしようかということになります。過去の数学科では新課程用の問題と旧課程用の問題を出題していましたが、「情報 I」では教科書で扱われている言語や環境に応じて、難易度が等質な問題を作成して出題することは不可能に近いでしょう。そこで、「大学入学者選抜に係る大学入学共通テスト問題作成方針」には次のように記述されています。「プログラミングに関する問題を出題する際のプログラム表記は、授業で多様なプログラミング言語が利用される可能性があることから、受験者が初見でも理解できる大学入試センター独自のプログラム表記を用いる。」なんと、テストのために架空の言語を規定したのです。2022 年に「共通テスト手順記述標準言語 (DNCL)」が独立行政法人大学入試センターにより最初は「情報関係基礎」のために発表されました。共通テスト「情報 I」の受験生はもう一つ言語を学ぶことになったのです。この言語の仕様は「変数と値」「表示文」「代入文」「演算」「制御文」「用意された関数の呼び出し」「新しい関数の定義」よりなります。「謎のプログラム」とか「実行できない言語を学ばせている」などの風評もありますが、現在では実行環境を Web 上に提供していただいている方もおり、今回はその環境「どんくり」でのプログラミングを紹介させていただきます。先生方の「受験指導」の一助になれば幸いです。

▶ DNCL (どんくり) の基本

DNCL のプログラムは日本語で命令を書くことができます。文字列は「」や“ ”で囲って記述します。**【図 1】**のように「を表示する」と指示すれば、その前部分を表示してくれます。授業で解説していると、日本語で入力する点に違和感をもつ学生さんが多いです。また、変数も使えます。変数への代入は`<`で指示します。`=`でない点も独特です。

演算については算術演算と比較演算、そして論理演算が可能です。加減乗除の演算は、「+」、「-」、「×」、「/」で指定できます。整数の除法では商を「÷」で、余りを「%」で計算することもできます。比較演算は、「=」、「≠」、「>」、「≥」、「≤」、「<」が使えます。最後に、論理演算では「かつ」、「または」、「でない」の演算子で指定します。

次に、それらの演算を用いた条件分岐文について紹介します。条件分岐は次のように書きます。

```
もし <条件> ならば
    <処理>
を 実行する
```

または、少し長くなりますが、次のように書くこともできます。

```
もし <条件> ならば
    <処理 1>
を 実行し、そうでなければ
    <処理 2>
を 実行する
```

例えば、変数に代入された数が偶数か奇数かを判定するプログラムを考えてみたいと思います。hen_suu に代入された値を 2 で割ったとき、余りが 0 ならば偶数となり、それ以外 (余りが 1) ならば奇数となります。このことを算術演算「%」を使って作成すると**【図 2】**のようになります。

最後に、条件繰返し文について紹介します。条件繰返し文には、前判定と後判定の 2 種類があります。前判定は次のように書きます。

プログラム	値の入力欄
1 「サイエンス」を表示する	
2	
3 hen_suu←12	
4 hen_suuと「個あった」を表示する	結果表示 サイエンス 12個あった ---

【図1】 表示と変数

プログラム	値の入力欄
1 hen_suu←12	
2 もし hen_suu%2=0 ならば	
3 「偶数です」を表示する	
4 を実行し、そうでなければ	結果表示 偶数です ---
5 「奇数です」を表示する	
6 を実行する	

【図2】 偶数が奇数かを判定するプログラム

プログラム	値の入力欄
1 num←6	
2 num<10 の間、	
3 numを表示する	結果表示 6 7 8 9 ---
4 num←num+1	
5 を繰り返す	

【図3】 10の手前まで整数を増やすプログラム

〈条件〉の間、
〈処理〉
を繰り返す

また、後判定は次のように書きます。

繰り返し、
〈処理〉
を、〈条件〉になるまで実行する

例えば、決めた整数から10の手前まで数を1ずつ増やしていくプログラムを作成してみます。最初に繰り返しを終了する条件（10未満）を設定し、その後繰り返す内容を書けば【図3】のようになります。

このように、DNCLには、他の言語と同様に、演算や条件分岐、条件繰り返しといった処理が可能となっています。

▶ 学生が作成したプログラム

次に、学生たちが実際に作成したプログラムのいくつかを紹介したいと思います。私たちが担当している講義『情報科教育法』では、DNCLのプログラムを作成する演習を毎年行っており、学生たちに様々なプログラムを作成してもらっています。

例えば、条件分岐を用いて指定した時刻が午前か午後かを判断するプログラムを作ってもらったことがあります。最も典型的なものは【図4】のものでしょう。午前か午後かを表示するだけのプログラムですので、この課題を課した際プログラムのバリエーションはそれほど多くないと考えていました。しかし、提出されたプログラムを確認すると、学生のこだわりや想定の

実行	保存	読み込み	サンプル一覧
DNCL	英語表示		
1 time←15			
2 もし time < 12 ならば			
3 「午前」を表示する			
4 を実行し、そうでなければ			
5 「午後」を表示する			
6 を実行する			

【図4】 典型的なプログラム

実行	保存	読み込み	サンプル一覧
DNCL	英語表示		
1 jikan←13			
2 もし jikan<12ならば			
3 「午前」を表示する			
4 を実行し、そうでなくもしjikan>12ならば			
5 「午後」を表示する			
6 を実行し、そうでなければ			
7 「正午」を表示する			
8 を実行する			

【図5】 正午を含めたプログラム

実行	保存	読み込み	サンプル一覧
DNCL	英語表示		
1 time←26			
2 time-time%24			
3 もし time < 12 ならば			
4 "午前"を表示する			
5 を実行する			
6 もし time >= 12 ならば			
7 "午後"を表示する			
8 を実行する			

【図6】 24時以上を想定したプログラム

深さがうかがえるものもありました。例えば、【図5】は午前か午後だけでなく12時ちょうどを正午と表示するものです。正午が表示されない仕様では満足できなかったことがうかがえます。また、【図6】は25以上の時刻でも午前か午後かを判断できるものです。【図6】のプログラムは想定していないものでした。学生のこだわりを感じます。自分の思い、こだわりでどこまでも自由に変えられるのがプログラミングのよいところです。

▶ DNCL を使えるサイトの紹介

最初にも述べましたように、DNCL（どんくり）は大学入学共通テストにおいて特定の言語による偏りをなくしてアルゴリズムの理解度を評価するために作られた疑似言語です。この言語を作成した大学入試センターは開発環境を提供していません。そのため、DNCLを使用するには使用環境を提供してくれるサイトに行く、またはアプリをインストールする必要があります。例えば2025年末時点ですと次のようなサイトがありますので、自分に合ったサイトを見つけておくとよいでしょう。

- 共通テスト用プログラム表記学習環境「どんくり」
<https://dncl.eplang.jp/#!index.md>
- あんこエデュケーションはアシアル情報教育研究所
<https://anko.education/>
- DNCL 処理系 XTetra
<https://t-daimon.jp/XTetra/>