

XTetraの開発と授業実践による評価

大門 巧¹ 大西建輔² 青山 浩³

概要: 我々は、DNCLのブラウザ上での実行環境として Tetra を開発したが、高等学校での授業で使用するためには、機能の充実を図ることが必要であった。そこで、Tetra を改良した DNCL のブラウザ上での実行環境である XTetra を開発した。XTetra には、従来の Tetra の機能に加え、コンソールからの標準入力、描画関数、エディタのシンタックスハイライト機能が実装されている。高等学校において、Scratch の使用経験のあるクラスと、使用経験のほとんどないクラスで XTetra を用いた授業を実施し、アンケートを用いて処理系の評価をおこなった。その結果、Scratch の使用経験のあるクラスの理解度が使用経験のないクラスと比較して高くなった。また、XTetra はビジュアルプログラミング言語の要素とテキストベースのプログラミング言語の要素の両方を備えているため、ビジュアルプログラミング言語でのプログラミング経験のある生徒の理解を高める効果が期待できるという仮説が立った。

キーワード: プログラミング, DNCL, 入力支援

Development of Browser-Based Programming Execution Environment XTetra and Its Evaluation with Educational Practice

TAKUMI DAIMON¹ KENSUKE ONISHI² HIROSHI AOYAMA³

Abstract: We developed Tetra as a browser-based execution environment for DNCL, but it needed to be enhanced to be used in a high school class. In addition to the functions of Tetra, XTetra has standard input from the console, drawing functions, and syntax highlighting for the editor. In a high school, we conducted classes using XTetra in a class with experience using Scratch and in a class with little experience using Scratch, and evaluated the execution environment using a questionnaire. As a result, the understanding level of the class with Scratch experience was higher than that of the class with no experience. We also hypothesized that because XTetra has elements of both a visual programming language and a text-based programming language, it can be expected to enhance the understanding of students who have experience programming in a visual programming language.

Keywords: Programming, DNCL, Input support

1. はじめに

大学入試センターは、2025年以降の大学入試共通テストにおける「情報」の出題を正式に発表した。これまでのセ

ンター試験や大学入学共通テストでは、問題において手順やアルゴリズムを記述するために、独自の疑似言語である DNCL[9] が使用されてきた。2025年以降の大学入学共通テストでも、DNCLのような日本語の疑似言語が用いられることが示唆されている [10]。

過去に我々は DNCL に着目し、ブラウザ上での実行環境である Tetra[8] を開発した。Tetra には、ダイアログを用いた入力支援システムが実装されており、DNCL の実行、停止、変数リストと実行ログによるプログラムの追跡が可能である。しかし、実際に授業で使用するためには、標準

¹ 東海大学大学院理学研究科数理学専攻
Tokai University Graduate School of Science Course of Mathematical Sciences, Kanagawa, 259-1292, Japan

² 東海大学
Tokai University, Kanagawa, 259-1292, Japan

³ 智辯学園中学校・高等学校
Chiben Gakuen Junior and Senior High School, Nara, 637-0034, Japan

入力関数の導入や処理系の機能を充実させることが課題となっていた。そこで、我々は Tetra に標準入力やエディタのシンタックスハイライト機能を追加した DNCL の処理系である XTetra を開発した。また、XTetra を高等学校の情報の授業の実習で実際に使用し、アンケートを用いて XTetra の評価をおこなった。

本論文の構成は次の通りである。2 節では、DNCL の処理系の関連研究を紹介する。3 節では、XTetra の概要と Tetra からの改良点について述べる。4 節では授業実践の内容について、5 節では実施したアンケートの詳細とそのデータについて説明する。6 節では、アンケート結果についての考察をおこなう。7 節では、アンケート結果についての議論をおこなう。8 節では本論文のまとめを述べる。

2. 関連研究

2.1 PEN

PEN[1] は Java 仮想マシン上で動作する DNCL の実行環境である。PEN では、DNCL に変数の型宣言やグラフィックス用の組み込み関数などの拡張を施した、xDNCL が実行可能である。プログラムの入力を支援する入力支援ボタンやステップ実行の機能が実装されている。PEN を用いた授業実践もおこなわれている。授業実践のアンケートでは、初学者に対して好評であることが確認されている。また、JavaScript を用いた授業と PEN を用いた授業による比較実験を実施し、自己評価と試験による分析から、PEN を用いたクラス理解度の方が高くなり、プログラミング入門用の環境としての有用性が確認されている。

2.2 PenFlowchart

PenFlowchart[3] は、フローチャート进行操作することで、PEN で実行可能なプログラムを生成し、実行することができるシステムである。文献 [5] では、PenFlowchart を用いた授業実践について報告されている。生徒に実施したアンケートからは、PenFlowchart をプログラミング学習の導入時に利用することで、初学者の学習がスムーズにおこなわれることが確認されている。また、授業内で実施したテストの点数から、フローチャートやプログラムの構造を把握する問題について得点の向上が見られたことが確認されている。文献 [6] では、PenFlowchart と PEN の比較実験について報告されている。PenFlowchart を用いた場合、PEN だけで学習をおこなっていた場合と比較して、生徒の理解度が高くなり、つまり生徒が少ないため、授業の進行がスムーズになることが確認されている。また、期末試験におけるプログラムとフローチャートの変換をおこなう問題の点数の向上が、成績下位者に顕著に見られたことが確認されている。

2.3 WaPEN

WaPEN[2] は、Java で実装された PenFlowchart[3] の環境を、ウェブブラウザ上で実現した DNCL の実行環境である。WaPEN で実行可能な DNCL には、描画関数、組み込み関数、関数定義などの拡張要素がある。DNCL をフローチャートに変換する機能と、フローチャートから DNCL プログラムを生成する機能が実装されている。DNCL を Python のような文法で記述することができる PyPen[4] も公開されている。

2.4 どんくり

どんくり [7] は、ブラウザ上で動作する DNCL の実行環境である。どんくりには、プログラムの実行速度や制御構文の実行回数等を計測する機能や、DNCL と英語表記の疑似言語の相互変換をおこなう機能がある。

2.5 Tetra

Tetra[8]^{*1} は、ブラウザ上で動作する DNCL の実行環境である。Tetra には、ダイアログによる入力支援システムが実装されており、DNCL の実行、停止、変数リストと実行ログによるプログラムの追跡が可能である。

3. XTetra

3.1 概要

XTetra^{*2} は、Tetra の入力支援システム、詳細なエラーレポート等の機能を継承し、コンソールによる標準入力や描画関数などの機能を実装した DNCL の処理系である。XTetra で実行可能な DNCL の文法は、仕様書 [9] に準拠している。その上で、標準入力、描画関数、関数定義、拡張ループ (for-in 文) を追加した。XTetra の IDE を図 1 に示す。画面上部には、入力支援システムを開くボタンが並ぶ。画面の中央部は左右に分割されており、左半分にはプログラムを記述するエディタ、右半分にはタブによる切り替えが可能なコンソール、描画領域、実行ログ、変数リストを表示するエリアが配置されている。エディタには、構文を色分けして表示するシンタックスハイライトの機能や、インデントの階層を示すルーラーを自動で表示する機能がある。エディタの上部には、各種設定、プログラムの保存、プログラムを開く、デバッグ、実行、ステップ実行のためのボタンが並ぶ。IDE の Web ページは、端末のサイズに応じて配置を変化させるレスポンシブウェブデザインを取り入れている。

3.2 入力支援システム

DNCL は日本語で記述するため、タイピングによる記述では変換ミスが多発する。入力支援ボタンを用いてこの問

*1 <https://t-daimon.jp/Tetra>

*2 <https://t-daimon.jp/XTetra>

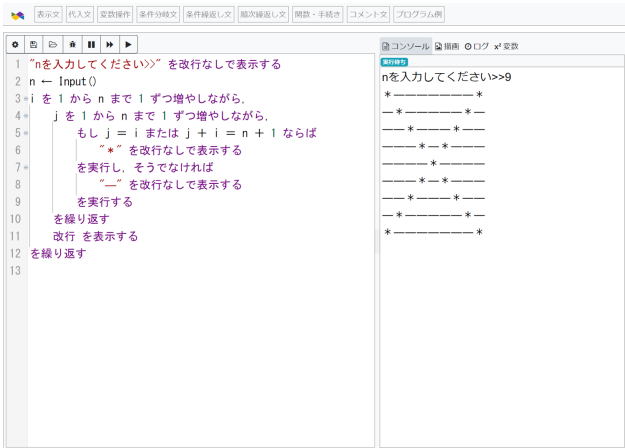


図 1 X Tetra の IDE

題を解決している。入力支援ボタンは DNCL のすべての構文に対応している。Tetra では、図 2 の左に示す入力支援システムが実装されている。構文が単色で表現されているため、ユーザーが今自分が何を入力しているのかを把握できなくなる恐れがある。そのため、XTetra では構文を色分けした(図 3)。

DNCL の構文では、増やしながらか繰り返す順次繰返し文と、減らしながらか繰り返す順次繰返し文のように、似た構造を持つものが存在する。似たような部分がある構文を切り替える場合、すべてのプロパティを再入力する必要があり、手間がかかる。そこで、XTetra では図 2 の右のように、構文を色付きで表示し、「増やしながらか」や「減らしながらか」などのプロパティをすぐに切り替えることができるように改良した。

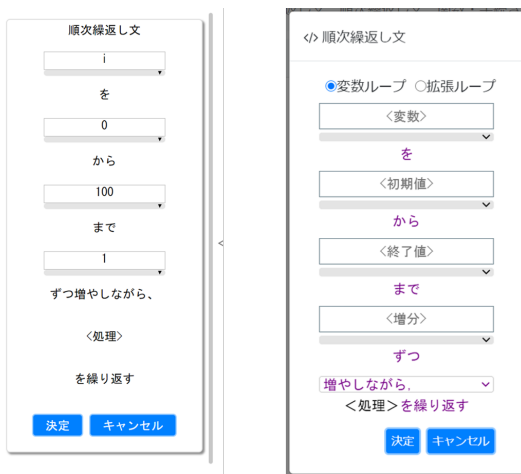


図 2 順次繰返し文の入力支援システム (左: Tetra, 右: X Tetra)

3.3 コンソールによる標準入力

プログラミングの学習において、標準入力は必要不可欠な要素である。DNCL の仕様には標準入力が存在しない。XTetra では、新たにコンソールによる標準入力の機能を実

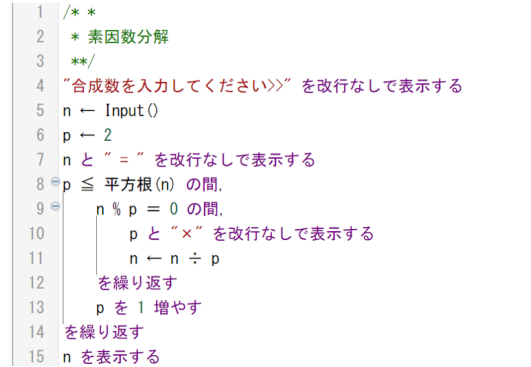


図 3 色分けされた構文

装した。標準入力は組み込み関数として提供している。表 1 に標準入力関数を示す。標準入力関数にはコンソールからの入力を文字列として取得する「入力」と、数値として取得する「Input」がある。標準入力関数の使用例を図 4 に示す。標準入力の待機中は、コンソールの上部に「入力待ち」と表示される。コンソールは HTML の textarea 要素で実装されている。標準出力で出力された値は編集不可能であり、標準入力を受け付ける部分は編集が可能になっている。

表 1 標準入力関数

入力 ()	コンソールに入力された値を文字列として返す。
Input()	コンソールに入力された値を数値に変換して返す。 入力値が変換できない場合は、NaN(非数値) を返す。

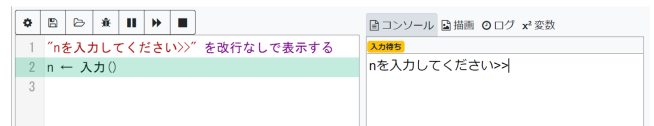


図 4 標準入力関数の使用例

3.4 描画関数

XTetra では、描画領域 (以下キャンバスと呼ぶ) に図形を描画するための機能を、組み込み関数として提供している。表 2 に描画関数とその役割を示す。図 5 は表 2 の描画関数を組み合わせて記述した、モンテカルロ法の動作を可視化するプログラムの例である。

3.5 エラーレポート

Tetra では詳細なエラーレポートの機能を提供している。XTetra では、エラーレポートの機能をさらに強化した。エラー発生時は、コンソールに詳細なエラーメッセージが表示されるだけでなく、図 6 のように、エラーの起きた箇所の直下にもエラーメッセージが表示される。

表 2 描画関数

キャンパス (w, h)	キャンパスのサイズを設定する。
背景の色 (c)	キャンパスの背景色を設定する。
$rgb(r, g, b)$	rgb による色の情報を生成する。
$rgba(r, g, b, a)$	$rgba$ による色の情報を生成する。
$hsl(h, s, l)$	hsl による色の情報を生成する。
$hsla(h, s, l, a)$	$hsla$ による色の情報を生成する。
図形の色 (c)	図形の塗りつぶしの色を設定する。
線の色 (c)	図形の枠線の色を設定する。
線の幅 (w)	図形の枠線の幅を設定する。
直線 (x_1, y_1, x_2, y_2)	直線を描画する。
四角形 (x, y, w, h)	四角形を描画する。
楕円 (x, y, w, h)	楕円を描画する。
文字列 (x, y, txt)	文字列を描画する。
文字列のサイズ (s)	描画する文字列のサイズを設定する。

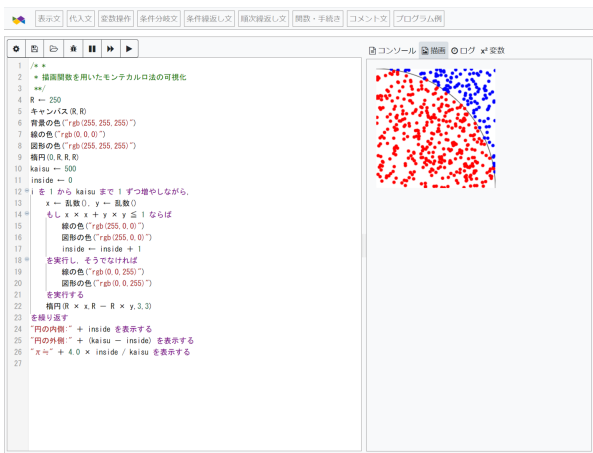


図 5 描画関数の使用例

```
//if 型
もし<条件>ならば
  <処理>
を実行する

//if - else 型
もし<条件>ならば
  <処理>
を実行し、そうでなければ
  <処理>
を実行する

//if - else if - else 型
もし<条件>ならば
  <処理>
を実行し、そうでなくもし<条件>ならば
  <処理>
を実行し、そうでなければ
  <処理>
を実行する
```

図 7 条件分岐文

```
<変数>を<初期値>から<終了値>まで<増分>ずつ増やしながら、
  <処理>
を繰り返す
```

図 8 順次繰返し文

```
<条件>の間、
  <処理>
を繰り返す
```

図 9 条件繰返し文

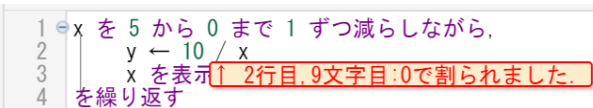


図 6 エラーレポートの様子

表 3 授業の概要

対象者	高等学校 1 年生 (4 クラス)
生徒数	142 名 (3 年制 30 名, 6 年制 112 名)
授業の構成	
1 回目	プログラミング学習の目的とアルゴリズム
2 回目	計算式の書き方
3 回目, 4 回目	変数, 代入, 入力
5 回目, 6 回目	順次繰返し文
7 回目, 8 回目	条件分岐文と条件繰返し文

4. 授業実践

授業実践の概要を表 3 に示す。高等学校 1 年生を対象とした情報の授業の実習で使用し、3 種類の条件分岐文 (図 7)、順次繰返し文 (図 8) と条件繰返し文 (図 9) の考え方を理解させることを目標とした。対象のクラスは、中高一貫の 6 年制クラスが 3 クラス、高校からの 3 年制クラスが 1

5. 評価

アンケートを用いて XTetra の評価をおこなった。アンケートは 8 回目の授業内で実施した。アンケートの冒頭部分では、集計したデータは論文のデータとして使用する予定であること、論文の執筆以外の目的には使用しないこと、データの利用方針に同意できる場合のみ回答することを注

意点として明記した。その結果、情報の授業を履修している生徒 142 人の内、134 人から回答があった。回答者数の内訳は、群 1 が 106 人、群 2 が 28 人である。アンケートは、4 択式の設問と、複数回答と自由記述が可能な設問から構成される。アンケートの項目と結果を表 6 と表 7 に示す。表 6 は群 1、表 7 は群 2 のものである。アンケートは 4 段階評価になっており、項目 1 から項目 3 は表 4、項目 4 から項目 6 は表 5 の選択肢になっている。4 段階評価の評価値は、4 を高評価とし、1 を低評価とした。表 6 の加重平均は、表 4、表 5 の評価値を用いて、それぞれの評価値に人数を掛け、和をとった値を回答者数 ($N = 106$) で割った値である。表 7 の加重平均は、表 4、表 5 の評価値を用いて、それぞれの評価値に人数を掛け、和をとった値を回答者数 ($N = 28$) で割った値である。表 8、表 9 は、アンケートの項目間の関係を分析するために計算した相関係数をまとめたものである。太字は相関のある項目 (0.5 以上) である。複数選択式のアンケートの項目と結果を表 10 に示す。複数選択式アンケートでは、その他の項目として自由記述欄を設けた。

アンケートの最後に、XTetra に追加してほしい機能の自由記述欄を設けた。この自由記述欄には、アンケートの回答者 134 人のうち 43 人から回答があった。図 10 は、この自由記述欄の意見から、頻出語を共起ネットワークで表現したものである。ノードの大きさは単語の出現回数を表している。円の面積が大きいほど、単語の出現回数が多いことを表している。辺上の数値はコサイン類似度である。コサイン類似度は文書の類似度を表す値である。ノード間の辺の色が濃いほど、コサイン類似度の値が大きく、2 つの単語がよく共起していることを表している。

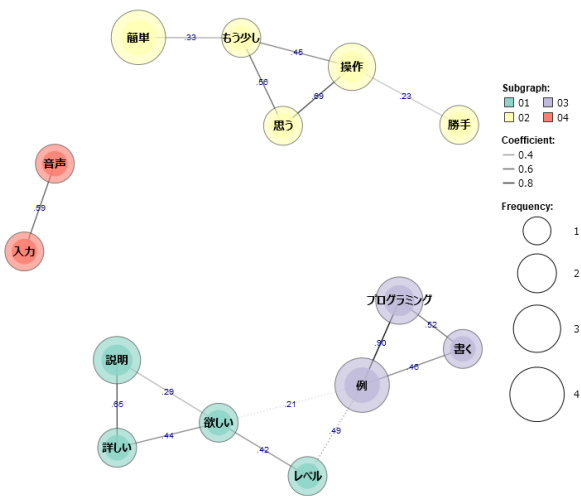


図 10 XTetra に追加してほしい機能の共起ネットワーク

表 4 表 6、表 7 の項目 1, 2, 3 の選択肢と評価値

選択肢	評価値
とても当てはまる	4
少し当てはまる	3
あまり当てはまらない	2
全く当てはまらない	1

表 5 表 6、表 7 の項目 4, 5, 6 の選択肢と評価値

選択肢	評価値
よく理解できた	4
少し理解できた	3
あまり理解できなかった	2
全く理解できなかった	1

6. 考察

6.1 授業の理解度

表 6 と表 7 の項目 4 から項目 6 についての考察をおこなう。これらの 3 項目は授業の理解度についての設問である。まず、表 6 と表 7 の項目 4 から項目 6 の共通点について述べる。項目 6 の条件繰返し文の書き方の理解度は、項目 4、項目 5 と比較して、「あまり理解できなかった」、「全く理解できなかった」と答えた生徒の割合が増加している。順次繰返し文と条件分岐文を学習した上で、ループを記述する方法として、新たに条件繰返し文が加わったことで、順次繰返し文と条件繰返し文のどちらを使用すればよいか分からない生徒が多数いたことが原因と考えられる。

表 6 の項目 4 から項目 6 についての考察をおこなう。3 項目それぞれの加重平均は、それぞれ 2.5 を上回っており、「よく理解できた」、「少し理解できた」と答えた生徒が 70% 以上を占めている。評価値の割合は、「少し理解できた」、「よく理解できた」、「あまり理解できなかった」、「全く理解できなかった」の順に多い。群 1 では、多くの生徒が授業内容を理解できたことが分かる。

表 7 の項目 4 から項目 6 についての考察をおこなう。3 項目それぞれの加重平均は、それぞれ 2.5 を上回っているが、群 1 よりも低い。「少し理解できた」と答えた生徒が 60% を占めている。評価値の割合は、「少し理解できた」、「あまり理解できなかった」、「よく理解できた」、「全く理解できなかった」の順に多い。群 1 と比較して、「よく理解できた」と答えた生徒の割合が小さくなっている。群 2 は、群 1 よりも理解度が低かったことが分かる。「全く理解できなかった」と答えた生徒の割合は、群 1 より低い結果となったが、これは、「あまり理解できなかった」と答えた生徒が「少し理解できた」に次いで多く存在したことが原因と考えられる。

6.2 処理系の評価

表 6、表 7 の項目 1 から項目 3 についての考察をおこな

表 6 群 1 の 4 択式アンケートの項目と評価 $N = 106$

項目番号	項目	加重平均	評価値 / %			
			4	3	2	1
1	XTetra でプログラムを簡単に書くことができましたか?	2.76	20.7	41.5	31.1	6.6
2	XTetra はプログラミングの勉強に役に立つと思いますか?	3.29	46.2	40.5	9.4	3.7
3	XTetra を今後のプログラミングの学習に活用したいと思いますか?	3.00	31.1	43.3	19.8	5.6
4	条件分岐文の書き方は理解できましたか?	3.08	32.0	48.1	16.0	3.7
5	順次繰返し文の書き方は理解できましたか?	3.09	33.0	47.1	16.0	3.7
6	条件繰返し文の書き方は理解できましたか?	3.03	33.0	42.4	19.8	4.7

表 7 群 2 の 4 択式アンケートの項目と評価 $N = 28$

項目番号	項目	加重平均	評価値 / %			
			4	3	2	1
1	XTetra でプログラムを簡単に書くことができましたか?	2.39	10.7	32.1	42.8	14.2
2	XTetra はプログラミングの勉強に役に立つと思いますか?	3.28	39.2	50.0	10.7	0.0
3	XTetra を今後のプログラミングの学習に活用したいと思いますか?	2.96	14.2	67.8	17.8	0.0
4	条件分岐文の書き方は理解できましたか?	2.96	17.8	60.7	21.4	0.0
5	順次繰返し文の書き方は理解できましたか?	2.78	7.1	67.8	21.4	3.5
6	条件繰返し文の書き方は理解できましたか?	2.75	7.1	60.7	32.1	0.0

表 8 表 6 の評価項目の相関係数

項目番号	2	3	4	5	6
1	0.37	0.38	0.39	0.50	0.50
2		0.72	0.38	0.42	0.48
3			0.36	0.39	0.48
4				0.51	0.55
5					0.75

表 9 表 7 の評価項目の相関係数

項目番号	2	3	4	5	6
1	0.37	0.32	0.09	0.35	0.27
2		0.51	-0.06	0.06	0.28
3			0.09	0.08	0.30
4				0.16	0.07
5					0.75

表 10 複数選択式アンケートの質問と結果

XTetra の機能のどのような点が良いと思われましたか?		
項目	群 1 $N = 106$	群 2 $N = 28$
「ボタン」でプログラムが書ける点	62.2%	57.1%
エラーの場所が細かく分かる点	50.9%	46.4%
変数の値が「変数リスト」から分かる点	29.2%	35.7%
ステップ実行ができる点	21.6%	14.2%
ソースコードが色分けされている点	28.3%	32.1%

う。これらの 3 項目は処理系の評価についての設問である。まず、表 6 と表 7 の項目 1 から項目 3 の共通点について述べる。項目 2 は、どちらのクラスにおいても 80% 以上の生徒が「とても当てはまる」、「少し当てはまる」と答えていた。多くの生徒が XTetra はプログラミングの勉強に役立つツールであると考えていたことが分かる。

表 6 の項目 1 では、「とても当てはまる」、「少し当ては

まる」と答えた生徒が 60% を超えている一方で、「あまり当てはまらない」と答えた生徒は 31.1% である。「全く当てはまらない」と答えた生徒も 6.6% 存在する。XTetra に対して使いにくさやシステムの複雑さを感じた生徒が存在することが分かる。項目 3 では、「とても当てはまる」、「少し当てはまる」と答えた生徒は全体の 70% を超えていた。多くの生徒が XTetra を今後のプログラミングの学習に活用したいと考えていたことが分かる。

表 7 の項目 1 では、「あまり当てはまらない」、「全く当てはまらない」と答えた生徒が 50% を超えている。群 2 には、XTetra に対して使いにくさやシステムの複雑さを感じた生徒が群 1 よりも多いことが分かる。項目 3 では、「少し当てはまる」と答えた生徒は 60% を超えていたが、「とても当てはまる」と答えた生徒は少なかった。授業を実施した青山は、群 2 では、分からない点を質問する生徒や、自分の書いたプログラムについて質問する生徒が多く、群 1 よりも生徒の積極性が高いという印象を受けている。群 2 において、XTetra に対して使いにくさやシステムの複雑さを感じた生徒が群 1 よりも多いのは、情報科に対する生徒の取り組みに差があったことが理由と考えられる。

次に、表 10 の考察をおこなう。群 1 では、「『ボタン』でプログラムが書ける点」を挙げた生徒が 62.2% であった。多くの生徒が、必要最低限の入力でプログラムを記述できることの良さを感じたことが分かる。表 6 の項目 1 において、「とても当てはまる」、「少し当てはまる」と答えた生徒が 60% を超えていたことも踏まえると、群 1 の生徒の方が、入力支援システムに有用性を感じた生徒が多いと考えられる。また、「エラーの場所が細かく分かる点」を挙げた生徒は、群 1 では 50.9%、群 2 では 46.4% であった。多くの生徒が、図 6 に示したエラーレポートの方法について、

良さを感じたことが分かる。自由記述では、「すべての機能が効率的で、UIも非常に分かりやすく、全く操作が分からない状態から初めても楽しかったです。」、「計算機能が素早く、非常にいいと思いました。」、「何もない」という回答が見られた。

6.3 XTetra に対する意見

図 10 に示した共起ネットワークについての考察をおこなう。XTetra に追加してほしい機能についての自由記述欄であるが、XTetra に対する意見を記述した生徒も見られた。「もう少し」、「思う」、「操作」からなる部分グラフでは、各単語間に強い共起関係が見られ、「簡単」、「勝手」からなる部分グラフと共起関係を持っている。実際に、4人の生徒が XTetra に対して「使い方を簡単にしてほしい」、「操作を簡単にしてほしい」という内容の意見を記述していた。表 6 の項目 1 についての考察でも述べたように、図 10 の共起ネットワークからも、XTetra に対して使いにくさやシステムの複雑さを感じた生徒が存在することが読み取れる。「説明」、「詳しい」、「欲しい」からなる部分グラフでは、各単語間に強い共起関係が見られる。また、「例」、「レベル」、「プログラミング」、「書く」からなる部分グラフでも、各単語間に強い共起関係が見られる。実際に、5人の生徒が「詳しい説明が欲しい」、「プログラミング例のレベルに何らかの問題がある」という内容の意見を記述していた。操作方法やプログラム例について、教員向けのドキュメントを Web ページとして公開しているが、生徒向けのドキュメントは作成していない。高校生が理解できるレベルのドキュメントも用意することで、改善が見られるのではないかと考えている。「音声」、「入力」からなるグラフでは、2つの単語間に強い共起関係が見られる。これは、「音声入力システムがあればよい」という内容の意見が見られたことを意味する。音声入力が標準入力であるか、入力支援システムの必要最低限の入力であるか、何を指すのかは不明であるが、どちらの場合であったとしても、実装を検討する価値はあると考えている。

6.4 評価項目に見られる相関

表 8、表 9 の項目間の相関についての考察をおこなう。項目 2 と項目 3 の間には、群 1、群 2 の両方において正の相関が見られた。項目 2 と項目 3 には同じような回答をした生徒が多いことが分かる。XTetra を今後のプログラミングの学習に活用したいと感じる生徒を増やすことは、生徒が自ら進んでプログラミングの学習に取り組む姿勢を育む上で大切なことである。今後の学習に活用したいと感じる生徒を増やすためには、XTetra が今後のプログラミングの勉強の役に立つシステムであることを授業内で実感してもらおうとよい。

項目 5 と項目 6 では、群 1、群 2 の両方において強い正

の相関が見られた。順次繰返し文と条件繰返し文の理解度が同程度である傾向が強いことが分かる。これらの 2 つの構文の理解度は、どちらか一方の構文の理解度の指標にもなり得るので、授業を進める上で注視すべきである。

項目 4 と項目 5、項目 4 と項目 6 は、群 1 では項目間で相関が見られたが、群 2 では無相関であった。条件分岐文と順次繰返し文、条件分岐文と条件繰返し文の理解度に基づきがあることが分かる。群 2 において、これらの構文を他方の構文の理解度の指標とするには不十分である。

7. 議論

群 1 と群 2 では、群 1 の方が、順次繰返し文、条件分岐文、条件繰返し文の書き方に対する理解度が高くなった。また、群 1 では、プログラムを簡単に書くことができたと言った生徒も多く見られ、入力支援システムを良い機能と評価していた。XTetra では Scratch のように、入力支援システムを使用し、GUI を操作してプログラムを作成することも、エディタに直接タイピングしてプログラムを作成することもできる。XTetra は、ビジュアルプログラミング言語の要素とテキストベースのプログラミング言語の要素の両方を備えた処理系とも言える。Scratch のようなビジュアルプログラミング言語の経験を積んだ生徒が XTetra を経由し、段階を踏んだ学習をおこなうことで、テキストベースのプログラミングに親しむことができたと考えられる。これにより、ビジュアルプログラミング言語でのプログラミング経験のある生徒の理解を高める効果も期待できる。一方で、XTetra に対して使いにくさやシステムの複雑さを感じた生徒が、群 1 と群 2 の両方に存在した。自由記述欄においても、「操作を簡単にしてほしい」という内容の意見が 11%見られた。そこで、生徒がプログラムを記述するために最も頻繁に操作した入力支援システムの操作を簡単にすることで、使いにくさやシステムの複雑さを感じる生徒は減少すると予想している。

8. 入力支援システムの改良

現在我々は「つちのこ」*3の開発を進めている。「つちのこ」は、大学入試共通テスト「情報」の試作問題 [10] において使用されている疑似言語の処理系である。疑似言語の処理系として利用するだけでなく、Python の導入のためのツールとして利用することも想定して開発した。IDE には入力支援システムやデータビジュアライゼーションの機能がある。また、オブジェクト指向によるプログラミングや、Python で利用される多くの構文を、Python のような日本語で記述することができる。「つちのこ」には、XTetra の評価実験で得られた処理系に対する意見を踏まえ、入力支援システムに Scratch のブロックの操作のような直感的

*3 <https://t-daimon.jp/tsuchinoko>

な操作ができる機能を実装した。「つちのこ」の入力支援システムを図 11 に示す。構文の形がそのまま反映されており、図 2 に示した Tetra と XTetra の入力支援システムよりも、さらに直感的な操作が可能になっている。

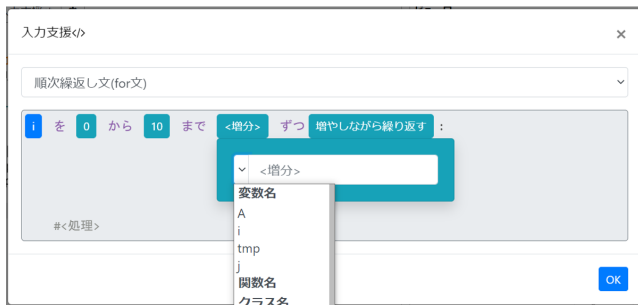


図 11 「つちのこ」の入力支援システム

9. まとめ

本稿では、DNCL のブラウザ上での実行環境である XTetra の開発と、XTetra を用いた授業実践について述べた。評価実験により、XTetra はビジュアルプログラミング言語の要素とテキストベースのプログラミング言語の要素の両方を備えているため、ビジュアルプログラミング言語でのプログラミング経験のある生徒の理解を高める効果が期待できるという仮説が立った。また、XTetra のビジュアルプログラミング言語から汎用言語への橋渡し役としての位置づけを明らかにし、処理系の改善すべき点を示すデータを得ることができた。今後は、テキストベースのプログラミング言語の導入のための環境としての XTetra の有用性を立証するために、XTetra を経由してからテキストベースのプログラミング言語を学習する群と、XTetra を使用せずにテキストベースのプログラミング言語を学習する群に分け、対照実験により検証をしていく必要がある。また、大学入試センターによる試作問題の公開や、本稿で述べた入力支援システムについての効果を踏まえ、評価実験で得られた知見を反映した「つちのこ」を用いて再度検証をおこなうことも計画している。Scratch のようなビジュアルプログラミング言語の要素を、テキストベースのプログラムを記述する手段として取り入れることで、処理系の使いやすさを向上させ、生徒の理解度を高めることができるのではないかと考えている。

謝辞 本研究のための授業を実施するにあたり、智辯学園中学校・高等学校の先生方ならびに生徒の皆さんの協力を得ています。評価実験にご協力いただき、感謝いたします。

参考文献

[1] 西田知博, 原田章, 中村亮太, 宮本友介, 松浦敏雄. 初学者用プログラミング学習環境 PEN の実装と評価. 情報処

- 理学会論文誌, pp.2736-2747, Vol.48, No.8, 2007.
- [2] 中西渉. Web ブラウザ上のプログラミング学習環境 WaPEN の改良. 情報処理学会, 情報教育シンポジウム (2019), pp.130-135, 2019.
- [3] 中西渉. PenFlowchart の開発. 情報処理学会研究報告, CE(113), pp.1-6, No.13, 2012.
- [4] 中西渉. Web ブラウザ上のプログラミング学習環境 PyPEN を用いた授業の提案. 情報処理学会第 83 回全国大会, pp.411-412, 2021.
- [5] 中西渉, 辰己丈夫, 西田知博. PenFlowchart によるプログラミング導入教育の評価. 情報処理学会研究報告, CE(121), pp.1-7, No.9, 2013.
- [6] 中西渉, 辰己丈夫, 西田知博. PenFlowchart を用いた、フローチャートによるプログラミング学習の効果に対する評価. 情報処理学会論文誌, pp.75-82, Vol.1, No.4, 2015.
- [7] 本多佑希, 兼宗進. ブラウザ上で動作する DNCL 学習環境「どんくり」の開発. 情報処理学会 コンピュータと教育研究会, CE(147), pp.1-4, No.10, 2018.
- [8] 大門巧, 大西建輔. ブラウザ上で動作する DNCL 処理系「Tetra」の開発. 情報処理学会研究報告, CE(151), pp.1-5, No.7, 2019.
- [9] 大学入試センター: センター試験用手順記述標準言語 (DNCL) の説明, 入手先 (<https://www.dnc.ac.jp/albums/abm00004841.pdf>)(accessed 2021-7-19).
- [10] 大学入試センター: 令和 7 年度以降の試験, 入手先 (https://www.dnc.ac.jp/kyotsu/shiken_jouhou/r7ikou.html)(accessed 2021-7-19).
- [11] Scratch Foundation: Scratch, 入手先 (<https://scratch.mit.edu/>)(accessed 2021-7-19).