

医療系大学におけるデザイン思考を取り入れた 異学年間プログラミング教育の実施

西牧 可織^{*1}, 二瓶 裕之^{*1}

^{*1} 北海道医療大学

Cross-grade programming educational model with design thinking at health sciences universities

Kaori Nishimaki^{*1}, Hiroyuki Nihei^{*1}

^{*1} Health Sciences University of Hokkaido

医療系大学生に対するプログラミング学習の授業において、上級年次学生が下級年次対象の授業を設計する仕組みを導入した。上級年次学生は自らのプログラミングスキルの醸成に加え、コマ数や理解度など授業運営における様々な課題を考慮したデザイン思考の観点から授業設計に取り組んだ。下級年次は学生視点を取り入れた演習課題を実施することでプログラミングに対する興味を低下させず高い満足度で授業を受けることができていた。

キーワード: プログラミング教育, 医療系大学, STEAM 教育, デザイン思考, 異学年間

1. はじめに

Society5.0 の実現に向け、社会全体で持続的に新たな価値を生みだすことができる人材の育成が求められている。教育現場においては数理・データサイエンス・AI に関する知識や技能の醸成のほか、学び方について変革の必要性が提言されている。例えば、「何を学び、身に着けることができたのか」学修者自らが社会に対し説明できる学習者本位の学び・能力や進度に応じて個別最適化された学び・異学年間の協働学修など多岐にわたる⁽¹⁾。特に Society5.0 では、AI や IoT, ロボットといった共通基盤技術と、産業コア技術、関連データの多様な組合せにより、革新的な製品・サービスを産みだすことが重要となる。特定の分野に偏らず多様な観点から課題にアプローチできる力を身につけるための学びが求められる。

そこで注目されているのが STEAM 教育である⁽²⁾。STEAM 教育は文理や学部を問わず、各教科での学習を実社会での課題解決に生かしていくための教科横断

的な教育である。この STEAM 教育やデザイン思考⁽³⁾の必要性を踏まえて、学生が必要とする教育をいかに提供していくかが高等教育における課題のひとつとなる。

STEAM 教育を実施する上で鍵となるのが「プログラミング学習」である。2020 年度より小学校で必修化され、2022 年度からの高等学校「情報」科目では新学習指導要領にもとづき、すべての生徒がプログラミングに触れることになるなど、これからの教育現場では必須の学びとなる。大学教育における数理・データサイエンス・AI (リテラシーレベル) モデルカリキュラムの項目としても「プログラミング基礎」が提示されている。

医療系総合大学である北海道医療大学(以下、本学)は数理データサイエンス AI 教育リテラシープラスに選定され、プログラミング学習を実施することとなった。しかし、医療系国家試験合格という定まった目標のある学生が多く、直接的に AI・システム開発に携わるといった専攻の学生と比較してプログラミングを主体

的に学べるような授業設計は難しい。学習者本位の学びや個別最適化された学び、異学年間の協働学修など学び方の工夫も考慮する必要がある。

そこで、学び方の工夫を考慮しながら医療系学生でも主体的にプログラミングを学べる環境を構築するために、プログラミング学習の授業において、上級年次の学生が、下級年次対象の授業を設計するといったデザイン思考を取り入れた異学年間プログラミング教育を実施した。対象としたのは本学心理学部臨床心理学科 3, 4 年生および 1 年生であり、学生の学習意欲を向上させるため小型の教育用ロボットをプログラミング学習用教材として活用した。

3, 4 年生の学生には予めプログラミングの基礎知識を身に付けてもらった上で、本学での学び、興味関心、コマ数や理解度など授業運営における様々な課題を考慮したデザイン思考の観点から 1 年生のプログラミング授業を設計することを課題とした。1 年生には、3, 4 年生が作成した演習課題を解決することを課題とした。

また、1 年生の学修状況・授業内容の評価を、教員を通じて 3, 4 年生にフィードバックすることで、間接的に異学年で学びの共有ができるようにした。クラウドアプリケーションも活用し、教員・学生間でリアルタイムかつスムーズに学習情報の共有ができるようにした。

3, 4 年生の考案した授業に対する 1 年生の学修目標達成度の自己評価および授業内容の評価により、異学年間で実施したプログラミング教育の効果を検証する。

2. 授業の概要と機材

2.1 対象とした授業科目と授業構成

今回の取り組みの対象としたのは本学心理学部臨床心理学科においてゼミナール形式で開講している 3 年生「専門演習 I」と 4 年生「専門演習 II」および「卒業研究」（以下、ゼミ）と 1 年生「情報科学」の授業である。ゼミは 3 年後期から配属先の各教員のもとで 4 年後期まで通年で実施される。筆者の指導する学生（以下、ゼミ生）は 3, 4 年生合わせて 2018 年度 8 名、2019 年度 5 名、2020 年度 2 名であった。「情報科学」

は 1 年次後期選択科目であり、授業回数は全 15 回、2019 年度の履修者は 28 名、2020 年度の履修者は 17 名であった。「情報科学」では、プログラミングに関する内容に加え、コンピュータのハードウェアの構造、インターネットと情報セキュリティなどの情報リテラシーの基礎を学ぶ。2020 年度のプログラミングに関する授業回は、第 8 回～第 14 回の 7 回であった。

2.2 使用した教育用小型ロボット教材

プログラミングを学ぶ上でゼミと情報科学共通で用いたツールが、図 1 左図に示す Makeblock 社の mbot と呼ばれる教育用小型ロボット教材であり、mcore と呼ばれる電子基板を搭載している⁽⁴⁾。Arduino Uno がベースとなっており、基板上に光センサ・LED・超音波センサ・赤外線センサそしてモーターが接続されている。モーターを制御しタイヤを動かすことで mbot を移動させることができる。

mbot を制御するために、図 1 右図に示すような mblock と呼ばれるソフトウェア上でアルゴリズムを構築してプログラムを作成する。作成したプログラムを USB ケーブルを通じて mbot に送信することで、mbot に接続されたセンサでデータを取得したり、LED を点灯したり、移動させたり様々な動作を表現することができる。

プログラムの作成は、ブロックを組み合わせることでアルゴリズムを構築する Scratch をベースとしたビジュアルプログラミングで行うことができるため、プログラミング言語の知識は基本的に不要である。本学科は従来パソコン必携であるため、学生の個人パソコンに mblock をインストールしてもらい、一人一人が適宜 mbot とパソコンを接続して動作確認ができるようにした。

ロボット教材を活用し、自分の作成したプログラムを視覚的・聴覚的に自由に表現できるようにすることで、プログラミングへの苦手意識の低減や興味関心の向上など多様な学習背景を持つ学生に対して学習意欲の向上を図ることができると考えた。今回はプログラミングの考え方・アルゴリズムの構築を理解することに主眼をおき、具体的なプログラミング言語の知識修

得を目的としていないため、授業内ではビジュアルプログラミングのみを扱うこととした。

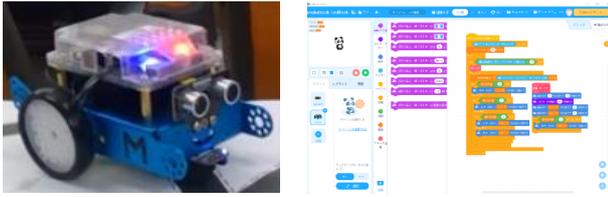


図1 mbot (左図) と mblock (右図)

3. 学びの共有の仕組みと上級年次学生（ゼミ生）による授業設計

3.1 学びの共有の仕組み

図2に学年毎の課題と異学年間での学びの共有の仕組みを示す。3,4年生のゼミ生は、ゼミ配属後 mbot や mblock を活用したプログラミングの基礎知識修得を目標として、初めに、mblock のインストール・mbot の組み立て・アルゴリズムの構築・各種センサの活用方法について一人一台 mbot を用いながら学ぶ。次に「次年度もしくは本年度の1年生にどのようにプログラミングの授業を提供するとよいか」の課題に取り組む。各自が身に付けたプログラミングの知識や技能に基づき、授業手順や課題作成・課題提示方法などの授業設計をゼミ生が協働で取り組むようにした。教員からは、学びの振り返りを促したり、1年生の興味関心、コマ数や理解度など授業運営におけるキーワードを適宜提示したりした。

2019年度の情報科学のプログラミング授業回から、実際にゼミ生が考案した授業を展開した。まず、2018年度のゼミ生が授業設計のたたき台を作成した。次に、これに基づき、2019年度のゼミ生が2019年度1年生の学修状況なども考慮しながら課題のブラッシュアップや動画教材の作成を新たに行った⁵⁾。1年生の学修状況は、個人情報に充分配慮した上で、教員を通じてゼミ生へフィードバックしている。最後に、教員が授業全体の調整を行った。

1年生は、授業内においてゼミ生が作成した課題を協働で解決することとした。また、授業毎の学びの振り返りとして、Google フォームを活用し、学修目標達成度の自己評価および授業内容の評価を送信することとした。その他に、授業アンケートや定期試験も実施

した。

教員は、授業内容の調整や個人情報の管理、ゼミ生と1年生の相互の指導なども含め、異学年間で間接的に学びを共有し、互いの課題解決を円滑に実施できるようサポートをする形で携わった。

学びの共有のためのツールとしてクラウドアプリケーションを活用した。1年生の学生が課題解決のためにアイデアを学生・教員と共有できるように、教員が学生の学修状況をリアルタイムで確認できるようにした。例えば、google スライド上での意見交換は学生にとって意見を出しやすいなどの利点があることから、スライド上にアイデアや課題解決のプロセスを記載して教員・学生間で情報共有をした⁶⁾。

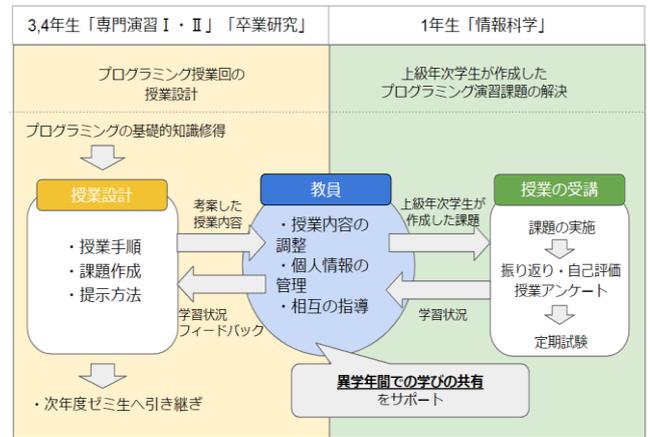


図2 異学年間の学びの共有の仕組み

3.2 2020年度における授業設計

表1に2020年度のゼミ生が携わった2020年度情報科学におけるプログラミング授業回の学修目標および2019年度からの変更点を示す。2020年度の情報科学の授業では、2019年度の課題実施状況やアンケートに基づき、プログラミング授業回を2回分増やすこととした。

2020年度ゼミ生は、新たに、表1の第9回、第10回の授業回における「レベル別・テーマ別の選択課題」を作成した。これは、1年生が自身の学びの進捗や興味などに応じて選択して実施することができる演習課題であり、個別最適化を意識した学びの実現を目標とした。図3にゼミ生が作成した課題の一部を示す。ポップなイラストやヒントなど学生視点からの「わかりやすさ」と、テーマや難易度の視覚化を促すことで教

員視点からの「限られた授業時間内で学生が主体的に学べる工夫」の双方の視点を意識させた。

表 1 2020 年度の学修目標

授業回数	授業内容	2020年度学修目標	2019年度からの変更点
第8回	概要・mbotの動作確認	第8回_1. プログラミングを学ぶ上での学修目標を説明できる	
		第8回_2. mblockとmbotを使ったビジュアルプログラミングについて基本的な操作方法を説明できる	
		第8回_3. グループで協力しながらプログラミングの課題を作成し、mbotを使って動作確認ができる	
第9回	基本課題	第9回_1. プログラミングの基本課題を解決することができる 第9回_2. 基本課題の解決方法についてスライドにまとめることができる	レベル別・テーマ別の選択課題(基本レベル)に変更
第10回	応用課題	第10回_1. プログラミングの応用課題(光・音・動く)を解決することができる 第10回_2. 応用課題の解決方法についてスライドにまとめることができる	※コマ数拡張 レベル別・テーマ別の選択課題(応用レベル)を導入
第11回	応用テーマ	第11回_1. 応用テーマ1(超音波センサー)のサンプルプログラムを説明できる 第11回_2. 応用テーマ1の課題解決方法についてスライドにまとめることができる 第11回_3. 応用テーマ2(ライトレーズセンサー)について概説できる	※コマ数拡張 センサに関する演習課題の解決を学習目標として導入
第12回	自由創作課題①	第12回_1. プログラミングのテーマを自分自身で決めることができる 第12回_2. 決めたテーマについてPDCAサイクルで課題解決を進めることができる	感染防止対策のためグループで1テーマから1人1テーマに変更 クラウド上での意見交換あり
第13回	自由創作課題②	第13回_1. プログラミングのテーマやアルゴリズムについて自分の考えを説明できる 第13回_2. テーマに関する課題解決をPDCAサイクルに沿って計画的に実行することができる	
第14回	創作課題の発表	第14回_1. 課題解決の結果(mbotを動作させた様子)を動画で撮影することができる 第14回_2. テーマに関する課題解決の過程と結果を動画とスライドを使って説明することができる。	

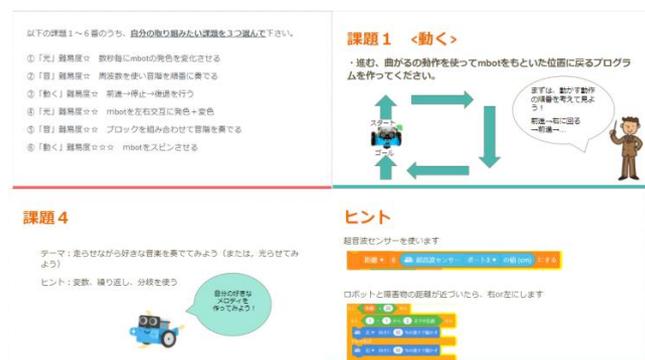


図 3 レベル別・テーマ別選択課題

4. 下級年次学生(1年生)による達成度の自己評価と授業評価

4.1 達成度の自己評価

図 4 は、2020 年度情報科学のプログラミング授業回を受講した 1 年生の学修目標達成度の自己評価である。縦軸は表 1 の学修目標と対応しており、横軸は学生の評価の割合である。◎は「良く達成できた」、○は「達成できた」、△は「達成できなかった」、×は「まったく達成できなかった」とした。第 11 回の一つの目標を除くすべての授業回・学修目標において、半数以上の学生は、目標を「よく達成できた」または「達成できた」と回答した。

特に、第 10 回と第 11 回は「良く達成できた」と答えた学生が半数であった。第 10 回は、2020 年度新たに導入したレベル別・テーマ別の選択課題の「応用レベル」の実施回であり、「光る」「動かす」など複数の動作を組み合わせたアルゴリズムの構築が課題であった。第 9 回においてレベル別・テーマ別の選択課題の「基本レベル」の演習課題を実施していたことや、応用レベルが難しい学生は基本レベルの課題をやり直してもよいこと、好きな課題を選択してよいなど学生が自由に取り組めるようにしたことも大きな達成感を得られた要因なのではないかと考える。

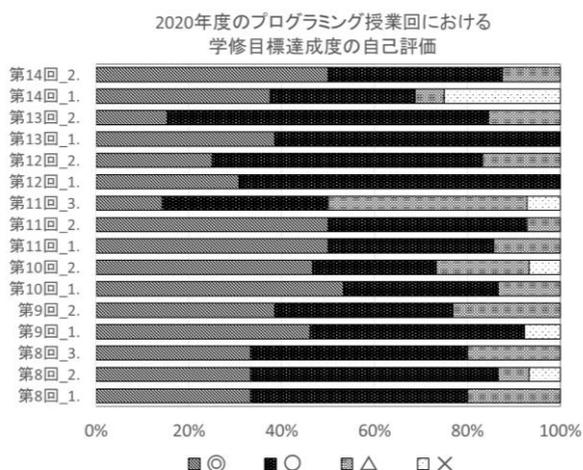


図4 学修目標達成度の自己評価

第11回は超音波センサやライントレースセンサの活用など、自動車の障害物回避や自動走行のアルゴリズムに関わる演習課題であった。超音波センサの演習課題については達成度が高く、ライントレースセンサの活用については達成度が大きく低下した。超音波センサの活用については mblock において予め超音波センサ専用のブロックが用意されている。一方で、ライントレースセンサによるライントレースを行うためには、センサで読み取った値に応じて条件分岐を行うアルゴリズムを構築する必要があった。さらに、mbot 本体にプログラムを転送する操作が必要であるため、ソフトウェアの操作に手間取ってしまったことも考えられる。

第12回以降は、これまで学んだアルゴリズムやセンサの活用を踏まえて、各自の自由な発想に基づいてテーマを考えプログラミングを行う「自由創作課題」を実施した。学生は、アルゴリズムの構築からプログラミング、mbot の動作確認、動作の撮影、プレゼンテーションスライドの作成まで行う。第11回までの与えられたテーマの課題実施状況と比較して、「達成できた」と答える学生が大幅に増加し、「よく達成できた」と答える学生が減少するなど自己評価の傾向が変化した。第11回までは課題の回答例としてサンプルプログラムを紹介していたため、学生自身が達成度を把握しやすかった。一方で、第12回以降は創作課題であるため回答例がないことや、2020年度はグループでの対

面での議論ができない状況下であったため、達成度の判断が難しかったのではないかと考える。

4.2 授業内容の評価

図5は各授業回における授業内容の評価の平均値を示したグラフである。横軸が授業回、縦軸が評価の受講者平均値である。授業の難易度、授業内容（資料・課題・説明など）の充実度、授業内容に対して興味をひかれた・面白いと思えたかの3項目について、それぞれ5段階で回答させた。

授業の難易度は「1:簡単」から「5:難しい」までの5段階である。授業7回分の平均は3.03($\sigma=0.27$)であった。初回の授業や創作課題と比較し応用レベル課題や超音波センサ・ライントレースセンサをテーマとした授業回を難しいと回答する学生がやや増加している。ただし、5と回答した学生はおらず、一般的に授業の進行とともに難易度が上昇すること、ライントレースセンサの活用を除き達成度の自己評価も低下していないことを考慮すると大きな影響はないものと考えられる。第8~9回は1と回答した学生がいたが、第10回以降は1または5と回答した学生はおらず、難易度の設定がおおむね適切であったと言える。

授業内容の充実度は、「1:まったく充実していない」から「5:とても充実している」までの5段階である。すべての授業回で4以上であった。特に第11回の平均値は4.79であり、達成度の自己評価も加味すると、難しいと感じる一方で、とても充実していると思えた学生が多いことが示唆される。

授業内容に対して興味をひかれた・面白いと思えたかについては、「1:まったく思わない」から「5:とてもそう思う」までの5段階である。授業内容の充実度と同様にすべての授業回で4を超えている。基本~応用レベルの演習課題から自由創作課題まで様々なパターンやレベルの課題を提示したが、興味や面白いと思うかの評価が低下することはなく、安定して興味を持たせ続けることができた。また、充実度の評価と強い相関が見られた($r=0.82$)。授業設計における資料・課題・説明が学生の興味や関心に大きく関係することが示された。

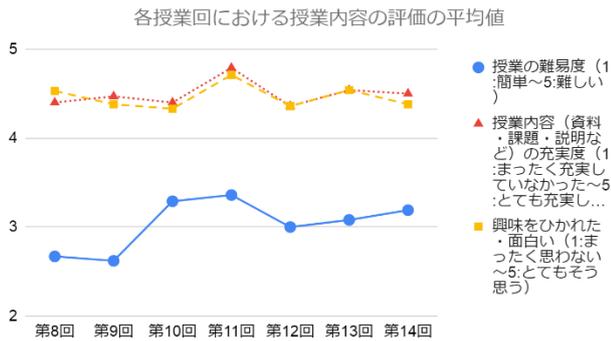


図5 授業内容の評価

5. おわりに

STEAM 教育やデザイン思考の観点から、上級年次の学生が下級年次の学生に対する授業設計を行い、異学年間で学びを共有することができた。また、学生自身が演習課題を選択する個別最適化に向けた工夫など society5.0 に向けた学び方の工夫を取り入れた。

下級年次の学生は上級年次の学生が作成した課題や提示資料を充実していると捉え、授業回を重ねても興味関心を保ったまま課題解決を行うことができていた。このことから、上級年次の学生は自らのプログラミングの基礎知識やスキルを修得していきながら1年生に対して達成度や授業内容の面で安定した授業を設計することができたと考える。年度毎に変動する1年生のプログラミング知識やスキル・ゼミ生の人数・授業回数などに対しても柔軟に対応することができていた。

特に、コロナ禍において教室の座席指定や対面での議論が難しい状況が加わった。しかし、学修効果を保ちながら mbot の可動域が自身の周囲だけであっても実施できる課題を考案するなど、授業運営側の教員・学生双方の視点やニーズを考慮し、デザイン思考を取り入れた授業設計ができていた。

ゼミの実施においても、コロナ禍対応として一時期 web 会議システムを活用したオンライン開催をすることとなったが、2020 年度ゼミ生は 2019 年度より継続して参加している 4 年生であったため、mbot や makeblock の基本的な知識や授業設計の流れ・次年度に向けた課題を把握できていた。また、情報共有にクラウドアプリケーションを活用していたことから情報共有や議論をスムーズに行うことができた。

報共有や議論をスムーズに行うことができた。

このように、不測の事態が生じて、2 学年を通して通年かつ合同で実施できるゼミのような形式の授業内で授業設計を行うことで、活動内容のスムーズな引き継ぎが可能となると考える。

今後は、定期試験の結果や具体的な課題解決のプロセスなどを検証し、プログラミング的思考力の定着を考慮した上で、異学年間での学びの共有や課題解決型学修を効果的に実施する手法を議論していきたい。また、ビジュアルプログラミングのみならず Python などプログラミング言語を用いたより深く難易度の高い学びにおいても適切な授業設計と課題の実施ができるか検討する。

謝辞

本論で報告した取り組みは、JSPS 科研費 19K14325 の助成を受けた。

参考文献

- (1) 中央教育審議会「2040 年に向けた高等教育のグランドデザイン (答申)」, https://www.mext.go.jp/content/20200312-mxt_koutou01-100006282_1.pdf (2022 年 2 月 15 日確認)
- (2) 山崎貞登, 松田孝, 二宮裕之, 久保田善彦, 磯部征尊, 川原田康文, 大森康正, 上野朝大: "Society5.0 を支える STEAM/STREAM 教育の推進に向けた小学校教育課程の教科等構成の在り方と学習指導形態", 上越教育大学研究紀要, 39, 2, 525-538, (2020)
- (3) 紺野登: "ビジネスのためのデザイン思考", 東洋経済新報社 (2010)
- (4) 久木田 寛直: "<改訂版>Makeblock 公式 mBot で楽しむ レッツ! ロボットプログラミング", FOM 出版 (2020)
- (5) 西牧可織: "教育用ロボット教材を用いた学年横断型プログラミング教育の導入", 北海道医療大学情報センター年報, Vol.13, 9-18 (2019)
- (6) 西牧可織, 二瓶裕之: "クラウド活用による同僚間アンケート調査を取り入れた問題発見課題解決型協働学修", ICT 利用による教育改善研究発表会論文・私立大学情報教育協会 1(1) 113 - 116 (2019)