

「マイコンを使ってプログラミングの基礎を学ぶ」授業モデル案(教材)

明治大学 情報コミュニケーション学部

和田 悟

概要

本教材は、文系学生 1,2 年生向けに、3 回程度の授業でプログラミングを扱うことを念頭において作成した。LED 等を搭載した安価なマイコンを使うことで、プログラムの動作をあきらかにするとともに、単純な動作であっても学生自らのプログラムによりマイコンを動作させたという達成感を得られるようにすることを狙っている。

1. 教材のねらい

学生らがプログラミングを学ぶことで期待しているのは、かれらが日常親しんでいるスマホのアプリやゲームなどけっこう高度なことなのではなかろうか。プログラミングの初歩で学ぶ事柄と、それらの実装の間には結構な乖離がある。1 学期を使った授業などでは可能だろうが、数回の授業ではそれも難しい。オンライン教材なども豊富な Unity などのプラットフォームでのゲーム作りの実践につなげることも考えられるが、ここではゴール(やれそうなこと、やれること)自体を簡単なレベルに引き下げても興味を惹きつける題材として「マイコンの制御」という課題を選んだ。

2. 教材を使用した実践について

筆者は 2018 年より、明治大学情報コミュニケーション学部(理系学部ではありません)の 2 年生を対象としたゼミナール相当の授業で、安価なマイコンとセンサーを利用する簡単な電子回路を作成する実習を行っており、そのつど価格や機能を考慮してマイコンを選んできた。2022 年度から使い始めたマイコン(Maker Nano(当時 900 円弱))が LED、ボタン、スピーカーを搭載し回路を組まずにそれらを使うことができることから、回路を組んだりするといった周辺事情にとらわれずに、数回の授業でプログラミングを学ぶのに活かせるのではないかと考え、今回の教材をまとめてみた。

Maker Nano (\$8.90, 1500 円弱(2024 年 2 月時点))

<https://www.cytron.io/p-maker-nano-simplifying-arduino-for-projects>

<https://akizukidenshi.com/catalog/g/g116282/>

マイコンボードから端子がでているので、ブレッドボードもあわせて購入する必要がある

だろうから、一人当たり 2000 円と USB(micro B)のケーブルが必要となる。

当方の授業では、別途センサーなどを繋ぎ回路を組ませる内容を含むので、回路をショートさせるなどのトラブルがありうるため、大学の教室備え付けの PC では行わず、学生自身の PC に開発環境をインストールさせて毎回持参させて授業を行った。外部の回路を作成しないのであれば、教室備え付けの PC に開発環境(Arduino IDE)をインストールして授業を行うことができるとおもう。その場合には、環境による操作の違いなどによる時間のロスをなくすることができる。

受講者は定員 20 名で、アシスタントなしで授業を行っている。12-16 名程度が適正規模ではないかとの実感を得ている。

3. この授業モデルの課題

2023 年度の授業(全 14 回)の第 3~5 回で、今回掲載した教材に基づいて授業を行ってみた。スライド記載事項に加えて、その場で説明をかなり補ったように思うが、実践を踏まえての反省点は以下の通りである。

- 文法事項の説明に時間を多くとられすぎ、自身が手を動かしたり考えたりする時間が少なくなってしまった。学生はプログラムの中身よりも、プログラムをなぞる(そのまま入力する)だけになってしまいがちだった。
→ 事前学習として指示する内容と、授業での実習内容の見直しが必要。

※ 教材マイコンは、教科書と同様に学生自身の負担とする方がのぞましい。大学予算で購入し、マイコンを貸与する場合には返却に関する誓約書などをとる必要がある。当方は、2018-2022 年度は、マイコンなどは基本的に学生の負担とし、自宅での自習なども行うことができたが、2023 年度は学生の負担を減らそうと、試みに大学で教材を用意し、履修学生に貸与する形で実施した。そうしたところ、例年より途中リタイアする学生が増え(2022 年度まではリタイア 0 名だった)、教材を返却しない学生もでてきた。持ち出しに制限を設けたところ、持ち帰り授業外で実習する学生がいなくなってしまった。

この教材を参考に授業を実施する場合には、教材マイコン費用を自己負担とし、授業外での自習が自由にできるようにするのが望ましいのではないかと考える。

- 割り切って、マイコンを操作できるという実感を持たせることに集中すべき。

通例であれば、「繰り返し」や「関数」など、後の回(7-8回あたり)であり挙げる内容を無理矢理3回の中に含めている。プログラミングにおいて非常に重要な事柄であるだけに、むりに3回に納めることの功罪を考える必要がある。いっそ割り切って、これらを取りあげず、自らのプログラムにより、現実世界への働きかけ(マイコンの動き)などを実現させるということに注力した方がよいかもしれない。

- プログラムの書法(コーディングスタイル)指導にすこし時間をとる方がよい。
学生は、なかなかインデントなどの規則を守らず、不具合のあるプログラムの点検にも時間がかかる。今回3回の授業の中では、その余裕がなく、多くの学生の実習課題ではインデント規則を守らないプログラムが多く、その問題点を指摘しても、その重要さを理解する学生は少なかった(前年度にPythonやProcessingなどのプログラミングの経験のあった学生の多くは、きちんと書けていた)。
なにか問題のあるプログラム例を挙げて、適切に修正(デバッグ)させるような課題を与え、プログラムの構造がわかることの大切さを知ってもらえるようにする課題を取り組ませるのがよいかもしれない。
- 3回の授業で網羅するより、自己効力感をもたせることが大切。
文法的に重要な項目を短時間で教え込むというよりは、自分が考えたコードで、コンピュータが動くという「自己効力感」を持たせることに注力したい。そうすれば、その次のステップの学びに繋がるのではないかと考える。そのために、より適切なプログラミングの課題を考えるのを次の課題としたい。

4. おわりに

今回の教材は、10回以上の授業で行う内容のうちプログラミングに関する内容を3回で納めようと意図したものであるが、その目的は十分に果たせてはいない。実際の授業では後の授業でいろいろと補足説明をしており、残念ながら結局3回では完結していない。来年度は、教材配布・事前準備の後、2~4回の内容として完結する内容にし、学生らがプログラミングに自発的に取り組めるような関心を維持できるような課題の考案に向けて改善したい。

今回の教材は3回完結を旨として設計したが、半期14回の授業でプログラミングを学ぶのであれば、数多く出版されているプログラミングの教科書を用いればよいだろう。本学

での実例では、藤本佳久『作って覚える やさしい Python 入門』(学術図書出版,2022)というテキストがあり、経営学部の学生を対象に Python を使って簡単なゲームを作る授業のよい実践例がある。