

# CGI による動的なネットワークサービス提供体験ができる プログラミング教材の提案

光永 法明<sup>1,a)</sup>

概要：インターネットがインフラとして普及しその仕組みを中学校段階でプログラミングを通して学ぶよう次期学習指導要領で求められている。そのため、その教育に将来携わる中学校 技術・家庭科 技術分野の教員を目指す学生には十分な理解が求められる。そこで CGI プログラムの書き換えを通して動的なサービスを作り、学生が自分の PC でサービスを提供する体験ができる教材を作成したので紹介する。プログラミング言語は Python とし、ウェブサーバには Python の `http.server` モジュールを利用する。教材は CGI プログラムのサンプルとプログラムを解説するテキストから構成される。大学生向けの講義 (2 回生向け、受講生 11 名) で教材を利用したところ、教材に大きな問題はないと分かった。また受講生全員がプログラムのサンプルを改良し作品を作成する課題を提出できた。

## 1. はじめに

現在、インターネットがインフラとして普及しその仕組みを中学校段階でプログラミングを通して学ぶよう次期学習指導要領で求められている [1]。そのため、その教育に将来携わる中学校 技術・家庭科 技術分野の教員を目指す学生には十分な理解が求められる。

インターネットの主な特徴としては IP アドレスを使うこと、ルータによりネットワークを相互接続し、IP アドレスと FQDN を結びつける DNS があり、分散管理されたネットワークでありインターネットへの接続やインターネット上のサービス提供などはそれぞれの管理者が許可すれば可能であることが挙げられる。またインターネット上の様々な情報やサービスが WWW の仕組みを利用して提供されている。WWW は主に HTTP/HTTPS プロトコル、HTML ドキュメント、URL によるリソースの指定とそれらを扱うウェブサーバとウェブブラウザ (クライアント) で構成される。

更新の少ない情報は静的な (変化しない) HTML ファイルに記述しておき、ファイルをウェブサーバに置く。ウェブブラウザで URL を開くよう操作すると、ブラウザは URL で指定されたサーバにリソースを要求しウェブサーバはファイルを提供する。ブラウザは HTML ファイルの記述に従って必要な画像などのファイルをさらに要求して

HTML の記述に従って整形し表示する。

一方でユーザの要求に応じて提供するデータを生成する動的なウェブサービスがある。動的なサービスには、検索エンジン、ウェブメール、FAQ 検索、電子掲示板、訪問者カウンタ、企業や大学の情報提供やブログなどに使われる CMS (contents management system) などがある。動的なサービスを提供するにはウェブサーバ側にそのためのプログラムが必要となる。古くからある動的なサービスを提供する仕組みに CGI (common gateway interface) がある。CGI の仕組みでは、ユーザからのリクエストに応じてウェブサーバが CGI プログラムを呼び出し、CGI プログラムの出力 (HTTP プロトコルヘッダの一部と動的に作成された HTML など) を受け取り、ユーザに返す。CGI プログラムはテキスト処理が主体であり、複雑なサービス提供は難しいがプログラムの構造が簡単である。

そこで CGI プログラムの書き換えを通して動的なサービスを作り、学生が自分の PC でサービスを提供する体験ができる教材を作成したので紹介する。プログラミング言語は教育でも実用でも広く使われている Python とし、ウェブサーバには Python の `http.server` モジュールを利用する。教材は CGI プログラムのサンプルとプログラムを解説するテキストから構成される。なお、本論文は以前に投稿した論文 [2] を加筆修正したものである。

## 2. CGI プログラムのサンプル

### 2.1 CGI プログラム

ウェブサーバが特定のリソースにアクセスされると CGI

<sup>1</sup> 大阪教育大学  
Osaka Kyoiku University, Kashiwara, Osaka 582-8582,  
Japan  
<sup>a)</sup> mitunaga @ cc.osaka-kyoiku.ac.jp

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
import cgi
import io
import os
import sys

sys.stdout = io.TextIOWrapper(sys.stdout.buffer, \
encoding='utf-8')
form = cgi.FieldStorage()

print('Content-type: text/html; charset="UTF-8"')
print()
print('<html>')
print('<head>')
print('<meta charset="UTF-8">')
print('</head>')
print('<body>')
print('Hello, World!<br>こんにちは<br>')
print('</body>')
print('</html>')
```

図 1 CGI の "hello world" プログラム

プログラムが呼び出される。呼び出された CGI プログラムが標準出力に HTTP ヘッダの一部(コンテンツのメタデータ)とコンテンツを出すと、それをウェブサーバはウェブクライアントへ返す。つまり最も簡単な CGI プログラムは print 文だけで構成できる(図 1)。図 1 のプログラムの場合はシェバンと Python スクリプトの文字コード指定、ライブラリの読み込み(省略できるものを含む)、標準出力の文字コードの指定、CGI のパラメータ受け渡し(フォーム)の準備、HTTP ヘッダを出力する print 文、ヘッダとコンテンツの境界を示す空行を出力する print 文、コンテンツの HTML を出力する print 文から成り立っている。

Python の cgi ライブラリを使い、ウェブサーバから CGI プログラムへ渡されるパラメータを受け取るには次のようにする。パラメータの名前が year で値が整数の場合は

```
form = cgi.FieldStorage()
```

で FieldStorage クラスのインスタンスを取得して

```
year = int(form.getvalue("year"))
```

とすればよい。int() で変換しない場合は文字列扱いとなる。year がウェブサーバから渡されないときにも正常動作するためには

```
if form.getvalue("year"):
    year = int(form.getvalue("year"))
else:
    year = 1989
```

とすればよい。これを使って西暦から和暦へ変換するプログラムは図 2 と書ける。

HTTP の GET 要求で localhost のポート 8000 番で接続を待つサーバの cgi-bin/year.py の CGI プログラムを year パラメータの値 2000 として実行するには

```
http://localhost:8000/cgi-bin/year.py?year=2000
```

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
import cgi
import io
import os
import sys

sys.stdout = io.TextIOWrapper(sys.stdout.buffer, \
encoding='utf-8')

form = cgi.FieldStorage()
if form.getvalue("year"):
    year = int(form.getvalue("year"))
else:
    year = 1989

hyear = year + 12 - 2000
conv = '平成' + str(hyear)

html="""Content-type: text/html; charset="UTF-8"

<html>
<head>
    <meta charset="UTF-8">
</head>
<body>
西暦 YEAR 年 = CONV 年
</body>
</html>"""

html = html.replace('YEAR', str(year))
html = html.replace('CONV', conv)

print(html)
```

図 2 西暦を和暦に変換して結果を表示する CGI プログラム

```
<html>
<head>
<meta charset="UTF-8">
</head>
<body>
<form>
西暦: <input name="year"> 年
<button type="submit" formaction="cgi-bin/year.py"
    formmethod="post">変換</button>
</form>
</body>
```

図 3 year.py にパラメータを渡して呼び出すフォームを含む HTML

をブラウザで開けばよい。この方法ではパラメータに何を引き渡せばよいかをあらかじめ知らないと開けないため、CGI プログラムを呼び出すフォームを含む HTML ファイルを用意することが多い。たとえば、図 3 の HTML を用意し、ブラウザで開くと図 4 が表示され、空欄に数値をいれ、変換ボタンをクリックすると year パラメータの値として数値が設定されサーバに渡される。また POST 要求にすることで、URL にパラメータが含まれない。GET、POST 要求のいずれを使うかは用途によるが、パラメータを見せない方がセキュリティ上よいと考えられる場合など

西暦:  年

図 4 西暦を和暦に変換するプログラムを呼び出すための HTML のブラウザでの表示



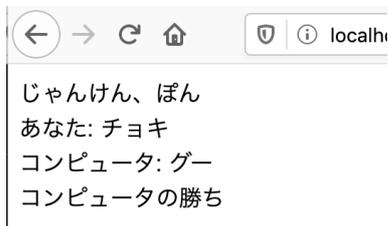
西暦 1985 年 = 昭和 60 年

図 5 西暦を和暦に変換するプログラムの出力



大阪太郎さんの前世は川で猫でした

図 6 診断メーカープログラムの出力



じゃんけん、ぼん  
あなた: チョキ  
コンピュータ: グー  
コンピュータの勝ち

図 7 じゃんけんプログラムの出力

に POST 要求が使われる。

## 2.2 サンプルプログラム

対象とする学生はプログラミングについての経験がそれほどないと考え、CGI プログラムと呼び出すフォームのサンプルとして以下を用意することにした。それぞれで重要となる主なプログラミング要素とサンプルの出力例を図に示している。

- Hello world : print 文
- 西暦・和暦変換: 整数の計算, 数値で判別する if 文, 文字の置換 (図 5)
- 診断メーカー (ランダムな文章を作成するプログラム): 乱数, リスト (配列) (図 6)
- じゃんけん: 乱数, リスト, 文字列で判別する if 文 (図 7)
- 換字暗号: リスト, 文字列の置換 (図 8)
- クイズ: リスト, 隠しパラメータの受け渡し (図 9)
- 掲示板: ファイルへの出力, ファイルからの読み込み (図 10)
- 投票: ファイルからの数値の読み込み, ファイルへの出力 (図 11)

これらのサンプルを順に解説 (クイズまで) し, 開発環境のインストール, サーバの起動についても書いたテキスト (A4, 34 ページ) を学生が自分のペースで学習を進められるよう用意した。



暗号文: ここしあふるとせき

図 8 暗号プログラムの出力。「おおきなふるとけい」を換字暗号で暗号化した。



問題: 熊猫は何を表す?  
あなたの答えはパンダ  
正解です。

## 第3問

問題: ギネスに認定された世界一重たいイチゴの重さは?  
 1kg  500g  250g  50g  10g

図 9 クイズプログラムの出力

名無しさん@192.168.11.18  
おもちゃ  
000@192.168.11.19  
hi

名前:   
  
コメント:

図 10 掲示板プログラムの出力。これ以外に画像を投稿できるものも用意した。



あなたはキノコ派? タケノコ派?

## 結果発表

キノコ派: 2人  
タケノコ派: 3人

図 11 投票プログラムの出力

## 3. 授業での活用例と学生の制作例

2019 年 7 月に本学の学部 2 回生向け講義 (必修科目, 受講生 11 名, 講義時間は 1 コマ 90 分を 2 コマ) で教材を用いた講義を実施した。受講学生 (1 名除く) は 1 回生時にプログラミングの入門科目 (選択科目) を受講していた。HTML については該当の 2 回生向け講義で説明し, HTML ファイルの作成課題を済ませていた。本学では BYOD が進められているため, プログラムの作成等には学生所有の PC を使うことにした。そこで Anaconda をインストール

して Python を利用出来る環境を準備させ、エディタ (開発環境) として Spyder を利用させることにした。サンプルプログラムと資料はあらかじめデータで配付し、クイズと投票についてはプログラムリストを印刷して配付した。講義では次のような順で説明・指示した。

- (1) 学生の PC への固定 IP アドレスの割り当て (ホワイトボードに割り当て一覧を書く) と設定
- (2) Anaconda プロンプトを開きウェブサーバを起動 (Python スクリプトを配布)
- (3) `http://localhost:8000/` をブラウザで開く
- (4) ローカルのファイルシステム上のファイルの位置関係と URL の関係を確かめる
- (5) HTTP プロトコルの簡単な説明 (テキストベース、GET/POST/PUT)
- (6) telnet (Python の telnet ライブラリ使用) で localhost へつなぎ、GET 要求を送る
- (7) サーバからクライアントへの応答がヘッダとボディからなることを確認
- (8) 適当な画像ファイルを見れるよう `test.html` を改変
- (9) ほかの人に自分のサーバにアクセスしてもらう (`http://各 PC の IP アドレス:8000/` を開く)
- (10) 動的な応答のために CGI の仕組みがあることを説明
- (11) Spyder で Hello world プログラム (`test.py`) を開き、実行してヘッダの一部と本文を出力することを確認
- (12) `http://localhost:8000/cgi-bin/test.py` を開き動作することを確認
- (13) `http://localhost:8000/cgi-bin/year.py?year=2018` を開き西暦・和暦変換のプログラムの結果を見る
- (14) フォームの説明
- (15) 西暦・和暦変換プログラムを改良する課題 (昭和対応、令和対応、昭和/平成/令和対応)
- (16) 診断メカ/じゃんけんサーバを改良する課題 (適切な画像をフォーム/CGI の出力に表示する)
- (17) クイズと投票の解説 (クイズの正解数をグラフにする方法の説明を含む)
- (18) クイズと投票の改良課題 (適切な画像を表示、クイズの数/投票先を増やす、工夫する、2人以上から感想をもらう)

講義の進行に大きな問題はなかった。講義中に終わらなかった課題については宿題とし後日メールで提出させた。すべての学生が課題を提出した。提出されたものを確認すると、じゃんけんの画像の選択やクイズの問題に個性/工夫が見られた。なかに 1 名の学生は図 12 のように投票数をグラフで見せるように改良していた。

また大学院生対象講義 (演習科目、複数教員で担当の科目で担当学生は 3 名) においてネットワークサービスについて理解するためになんらかのサービスを作成してみるという課題を与えた。そのとき 3 名のうち 1 名がこの CGI プ



図 12 学生の投票を基にした改良例

図 13 掲示版を基にした栽培日誌の入力フォーム

ログラムのサンプルを基に制作した (他の学生は別のプログラム例を基にすることを選択した)。その学生がサンプルの掲示版を応用して作成した栽培日誌の入力フォーム部分を図 10 に示す。ラジオボタンやチェックボックスを使うことで UI を改善していた。

## 4. まとめ

本稿では CGI プログラムの書き換えを通して動的なサービスを作り、学生が自分の PC でサービスを提供する経験ができる教材を作成し紹介した。また教材を用いた学部 2 回生向けに講義をしたところ教材に大きな問題はないとわかり、受講生全員が講義の課題を提出できた。また講義を通して学生が作成した作品を紹介した。

## 参考文献

- [1] 文部科学省：中学校学習指導要領 (平成 29 年告示) 解説技術・家庭編、開隆堂出版株式会社 (2018)。
- [2] 光永法明：動的なネットワークサービス提供体験ができるプログラミング教材の提案。日本産業技術教育学会 近畿支部 第 36 回研究発表会講演論文集, A-7 (2019)。