

# 情報表現のテストをハックしてみた with Chrome拡張機能

※練習カテゴリー

まろん。 (@rin\_montblank)



# 自己紹介

- まろん。 a.k.a. rin\_montblank
- 学部1年
- 趣味
  - 読書
  - 外出/散策/温泉巡り
  - JavaScript (TypeScript)
  - リバースエンジニアリング
- About
  - Portfolio - <https://montblank.fun/>
  - Twitter - [@rin\\_montblank](https://twitter.com/rin_montblank)
  - GitHub - [@otoneko1102](https://github.com/otoneko1102)

connected from: 61.44.171.192:0 class: LJ  
あなたのアカウントは b1025212 です  
[アカウント再設定](#)

これは、選択式のテストです。プログラムの理解の基礎固めが目的です。  
すべてのカテゴリの「クリア(満点を連続3回とるごと)」により、  
基本概念の獲得を確認します。

ヒント付き問題 (結果は教員/TAが重複しない参考記録とします)

「練習」のための出題です。選択肢を選んだ時点で正解/不正解が表示されます。

解答中は問題ごとに、正解を選ぶまでは先に進めません。

練習 カテゴリ1	練習 カテゴリ2	練習 カテゴリ3
練習 カテゴリ4	練習 カテゴリ5	

「本番」の出題です(教員に指定された時刻に回答したものが有効になります。)

クリアすることが単位取得の必要条件となる小テストです。

本番 カテゴリ1	本番 カテゴリ2	本番 カテゴリ3
本番 カテゴリ4	本番 カテゴリ5	

全ての問題には制限時間が設定されています。

1. 授業中に設定された試験時間内に、カテゴリごとに
2. 3回連続して全問正解することを、目指してください

出題される問題を公開しています

すべての問題とその正解、提示される選択肢を公開します。  
テストでは、順番とあわせ、順序を変えて提示されます。

成績が上がらない場合は、本番前に十分「練習」をしてください。正答がわからない場合には、まずは、以下のリンク先でゆっくり学んでください。

問題と正解 全問題	問題と正解 カテゴリ1	問題と正解 カテゴリ2
問題と正解 カテゴリ3	問題と正解 カテゴリ4	問題と正解 カテゴリ5

自分の練習・本番試験の記録

"b10"のあとに5桁の数字からなるアカウントでログインしていることを前提にして、成績登録をしています。

練習と本番の記録
----------

# 情報表現入門のチェックテストとは

- ログインが学籍番号のみでできるから簡単
- シンプルで素晴らしいづくりのWebページ
- 答えが全部記載されているページがある
- いつでも練習/本番カテゴリーが公開されている

[https://\[\\*\\*\\*\\*\].sakura.ne.jp/pq/index.html](https://[****].sakura.ne.jp/pq/index.html)

テストの説明

connected from: 61.44.171.192:0 class: LJ  
あなたのアカウントは b1025212 です  
[アカウント再設定](#)

これは、選択式のテストです。プログラムの理解の基礎固めが目的です。  
すべてのカテゴリの「クリア(満点を連続3回超えること)」により、  
基本概念の獲得を確認します。

ヒント付き問題 (結果は教員TAが重複しない参考記録とします)

「練習」のための出題です。選択肢を選んだ時点で正解/不正解が表示されます。  
**解答中は問題ごとに、正解を選ぶまでは先に進めません。**

練習 カテゴリ1	練習 カテゴリ2	練習 カテゴリ3
練習 カテゴリ4	練習 カテゴリ5	

「本番」の出題です(教員に指定された時刻に回答したものが有効になります。)

クリアすることが単位取得の必要条件となる小テストです。

本番 カテゴリ1	本番 カテゴリ2	本番 カテゴリ3
本番 カテゴリ4	本番 カテゴリ5	

全ての問題には制限時間が設定されています。

- 授業中に設定された試験時間内に、カテゴリごとに
- 3回連続して全問正解することを、目指してください

出題される問題を公開しています

すべての問題とその正解、提示される選択肢を公開します。  
テストでは、順番とあわせ、順序を変えて提示されます。

成績が上がらない場合は、本番前に十分「練習」をしてください。正答がわからない場合には、まずは、以下のリンク先でゆっくり学んでください。

問題と正解 全問題	問題と正解 カテゴリ1	問題と正解 カテゴリ2
問題と正解 カテゴリ3	問題と正解 カテゴリ4	問題と正解 カテゴリ5

自分の練習・本番試験の記録

\*b10\*のあとに5桁の数字からなるアカウントでログインしていることを前提にして、成績登録をしています。

練習と本番の記録
----------

# ここがまずい

- ログインが学籍番号のみでできるから簡単  
→ 他の人の学籍番号でテストを受けられてしまう
- シンプルで素晴らしいつくりのWebページ  
→ 情報を抜き取って整理しやすい
- 答えが全部記載されているページがある  
→ 情報を抜き取って(ry
- いつでも練習/本番カテゴリーが公開されている  
→ ツールテストし放題

https://[\*\*\*\*\*].sakura.ne.jp/pq/index.html

いろいろやれそう

**自動完答ツールをChrome拡張機能で実装しよう！**

**本番では絶対に使用しないでください！**

ソースコードも一般に配布しません :)

# Chrome拡張機能とは

- Chromium系統のブラウザで利用できる(ex: Google Chrome, Edge, Brave, ...)
- HTML/CSS/JavaScriptで記述する
- Chrome Extension APIを活用したブラウザで動作するアプリケーションを作成できる(ex: Storage, Runtime, Bookmark, ...)
- Chrome ウェブストアのデベロッパー登録(初回のみ5\$)で拡張機能を一般公開できる

# 1. 解答の取得

一般に公開されている  
\_\_\_\_.html(解答一覧)か  
ら取得します

1. 解答ページを  
fetchで取得
2. DOMParserで要  
素ごとに分けて扱  
いやすい形式に
3. 問と解答を抜き取  
りjson形式に

```
1 (async () => {
2   const res = await fetch("https://takegawalab.sakura.ne.jp/pq/prob/____.html");
3   const html = await res.text(); const parser = new DOMParser();
4   const doc = parser.parseFromString(html, "text/html");
5
6   const results = {};
7
8   const anchors = doc.querySelectorAll("a[name]");
9   anchors.forEach((anchor) => {
10    const name = anchor.getAttribute("name");
11    const expl = anchor.querySelector("div.expl");
12    if (!expl) return;
13
14    const raw_html = expl.innerHTML;
15    const split = raw_html.split(/<span class="ans">/);
16    if (split.length < 2) return;
17
18    const q_html = split[0].trim(); const a_html = split[1].trim();
19
20    const tmp = document.createElement("div");
21    tmp.innerHTML = q_html;
22    const q = tmp.textContent.trim().replace(/\s+/g, " ");
23    tmp.innerHTML = a_html;
24    const a = tmp.textContent.trim().replace(/^正解:\s*/, "");
25
26    results[`${name}`] = { q, a };
27  });
28
29  console.log(results);
30 });
```

# 1. 解答の取得

結果:

```
{
  "0001": {
    "q": "座標 (0, 0) と座標 (100, 100) を結ぶ直線を引く記述はどれか (1): line(0,0,100,100); (2): line(0,100,0,100) (3): line(100,0,0,100); (4): line(100,0,0,0);",
    "a": "line(0,0,100,100);"
  },
  "0002": {
    "q": "座標 (10, 40) と座標 (50, 90) を結ぶ直線を引く記述はどれか (1): line(10,40,50,90); (2): line(50,40,10,90); (3): line(40,10,90,50); (4): line(10,50,40,90);",
    "a": "line(10,40,50,90);"
  },
  "0003": {
    "q": "座標 (70, 30) と座標 (70, 90) を結ぶ直線を引く記述はどれか (1): line(70,30,70,90); (2): line(70,70,30,90); (3): line(300,70,70,30); (4): line(70,30,90,70);",
    "a": "line(70,30,70,90);"
  },
  ...,
  "0774": {
    ...,
  }
}
```

## 2. 答えを表示

learning.html(テスト(練習)ページ)の要素を書き換えます

1. MutationObserverで問の(再)表示を監視
2. ページの#titleから問番号を特定
3. 先ほど取得したjsonから適したものを取得
4. ページに『\*答えは「...」です。』と表示
5. 正しい解答(.choice)を文字列の一致で判別
6. 正しい解答(.choice)の色を変えて視覚的にわかりやすく

```
30
31 // 続き
32
33 const observer = new MutationObserver(() => {
34   const q = document.getElementById("problem");
35   if (q && !q.innerHTML.includes("*答えは")) {
36     getAns(q, results);
37   }
38 });
39
40 observer.observe(document.body, { childList: true, subtree: true });
41 });
42
43 function getAns(q, results) {
44   setTimeout(() => {
45     const title = document.getElementById("title");
46     if (!title) return;
47
48     const number = title.textContent?.split("/")[2];
49     if (!number) return;
50
51     const result = results[number];
52     if (!result) return;
53
54     console.log("問題番号: " + number);
55     console.log(result);
56
57     q.style.fontSize = "14px";
58     q.innerHTML = `
59     ${q.innerHTML}
60     <br>
61     <br>
62     *答えは 「${result.a}」 です。
63     `;
64
65     document.querySelectorAll("div.choice").forEach((choiceDiv) => {
66       const spans = choiceDiv.querySelectorAll("span.choice");
67       if (spans.length >= 2) {
68         const text = spans[1].textContent.trim();
69         if (text === result.a) {
70           spans[0].style.backgroundColor = "green";
71         }
72       }
73     });
74   }, 100);
75 }
```

## 2. 答えの表示

結果:

※練習カテゴリー

学習モードです。ここで間違えても成績には関係しません

0101/図形描画/0017

### 3秒 / 60秒 経過

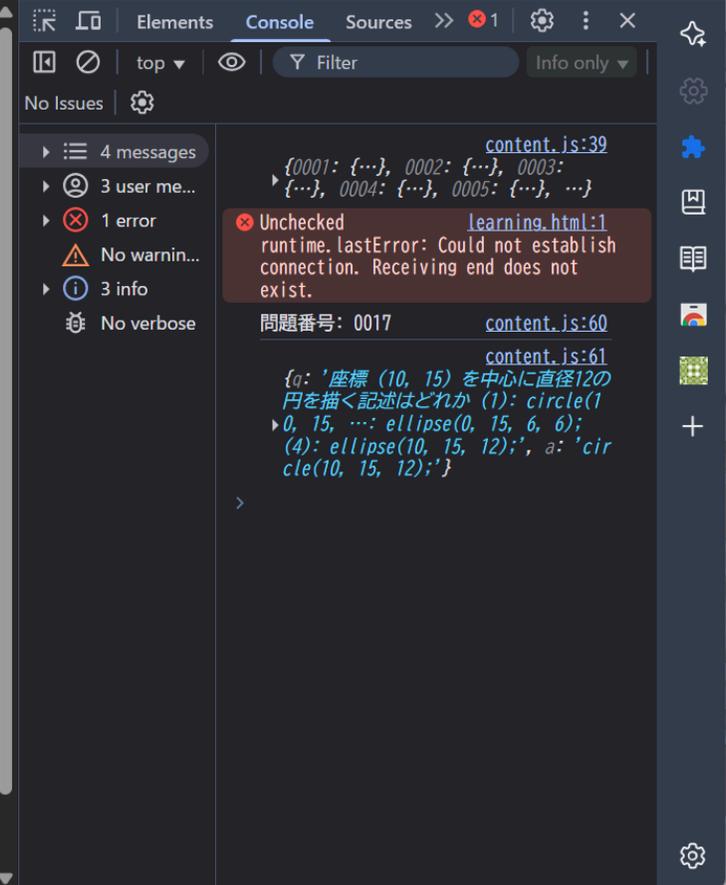
座標 (10, 15) を中心に直径12の円を描く記述はどれか

\*答えは「circle(10, 15, 12);」です。

- 1 circle(0, 15, 6);
- 2 circle(10, 15, 12);
- 3 ellipse(10, 15, 12);
- 4 ellipse(0, 15, 6, 6);

回答提出

最初からやり直し top page 記録をみる



```
content.js:39
{0001: {...}, 0002: {...}, 0003:
 {...}, 0004: {...}, 0005: {...}, ...}
Unchecked learning.html:1
runtime.lastError: Could not establish
connection. Receiving end does not
exist.
問題番号: 0017 content.js:60
content.js:61
{a: '座標 (10, 15) を中心に直径12の
円を描く記述はどれか (1): circle(
0, 15, ..., ellipse(0, 15, 6, 6);
(4): ellipse(10, 15, 12);', a: 'cir
cle(10, 15, 12);'}
```

### 3. 自動で完答

この程度ならPuppeteerやSeleniumを使わなくてもHTMLElement.click()のみで実装可能

getAns()を少し変更します

1. HTMLElement.click()で正しい解答(.choice)をクリック(選択)
2. 同じく回答提出をクリック

```
function getAns(q, results) {
  setTimeout(() => {
    const title = document.getElementById("title"); if (!title) return;
    const number = title.textContent?.split("/") [2]; if (!number) return;
    const result = results[number]; if (!result) return;
    console.log("問題番号: " + number); console.log("正解: " + result.a);

    q.style.fontSize = "14px";
    q.innerHTML = `
      ${q.innerHTML}
      <br>
      <br>
      *答えは 「${result.a}」 です。
    `;

    let clicked = false;
    document.querySelectorAll("div.choice").forEach((choiceDiv) => {
      const spans = choiceDiv.querySelectorAll("span.choice");
      if (spans.length >= 2) {
        const text = spans[1].textContent.trim();
        if (text === result.a && !clicked) {
          spans[0].click();
          spans[0].style.backgroundColor = "green";
          clicked = true;
        }

        setTimeout(() => {
          const submitBtn = document.querySelector(
            'input[type="button"][value="回答提出"]'
          );
          if (submitBtn) submitBtn.click();
        }, 300);
      }
    });
  }, 400);
}
```

# 3. 自動で完答

結果:

※練習カテゴリ

The screenshot shows a video player interface. On the left, a quiz page is displayed with the following text:

- 基本概念の獲得を確認します。
- ヒント付き問題 (結果は教員/TAが重視しない参考記録とします)
- 「練習」のための出題です。選択肢を選んだ時点で正解/不正解が表示されます。
- 解答中は問題ごとに、正解を選ぶまでは先に進めません。

Below the text are five buttons labeled "練習 カテゴリ1" through "練習 カテゴリ5". A red arrow points to the "練習 カテゴリ3" button. At the bottom of the quiz page, it says: "「本番」の出題です(教員に指定された時刻に回答したものが有効になります。)" and "クリアすることが単位取得の必要条件となる小テストです。"

On the right, a browser's developer console is open, showing several error messages:

- 9 messages
- 1 user message
- 8 errors
- 1 info
- No warnings
- No verbose

The error messages in the console include:

- Error handling response: index.html:1 TypeError: parameter 1 is not of type 'Node'. at chrome-extension://hbbgkeijgefhnbnk...
- Unchecked runtime.lastError: Could not establish connection. Receiving end does not exist. index.html:1 env.js:34
- Uncaught (in promise) Error: Could not establish connection. Receiving end does not exist. index.html:1

At the bottom of the video player, a progress bar shows 0:00 / 0:14.

# まとめ

- Chrome拡張機能ってスゴイ
- 情報表現入門のテストページはセキュリティ周りを強化すべきだと思う
  - ログインは学籍番号+パスワードにする(不正ログイン対策)
  - 解答をログイン介さないと見れないようにする(スクレイピング対策)
  - reCaptcha/hCaptchaなどを導入する(スクレイピング対策)
  - etc.

最後に

**ズルいのはダメ！死刑！**

本番では絶対に使用しないでください！