

2024年6月19日



AOJ 3.0 チュートリアル

<https://onlinejudge.u-aizu.ac.jp/beta/ice/>

UNIVERSITY OF AIZU

目次

はじめに	2
アカウント登録・ログイン	3
基本操作	4
設定	5
問題を選択する	6
コードを書く	9
テストする	10
AOJ へ提出する	13
解説を閲覧する	15
過去の提出履歴を閲覧する	17
模範解答を閲覧する	18
ブックマーク	20
アリーナ	21
前の問題・次の問題	23
問題を検索	24
ダッシュボード	25
他ユーザーのダッシュボードを表示	28
フリースペース	29
レビュー機能	30
ショートカットキー	33
お問い合わせ	33

はじめに

このチュートリアルでは、AOJ 3.0 の使い方を解説します。今後のリリースへ向けてユーザーインターフェースや機能が変更される可能性がありますので、あらかじめご了承ください。

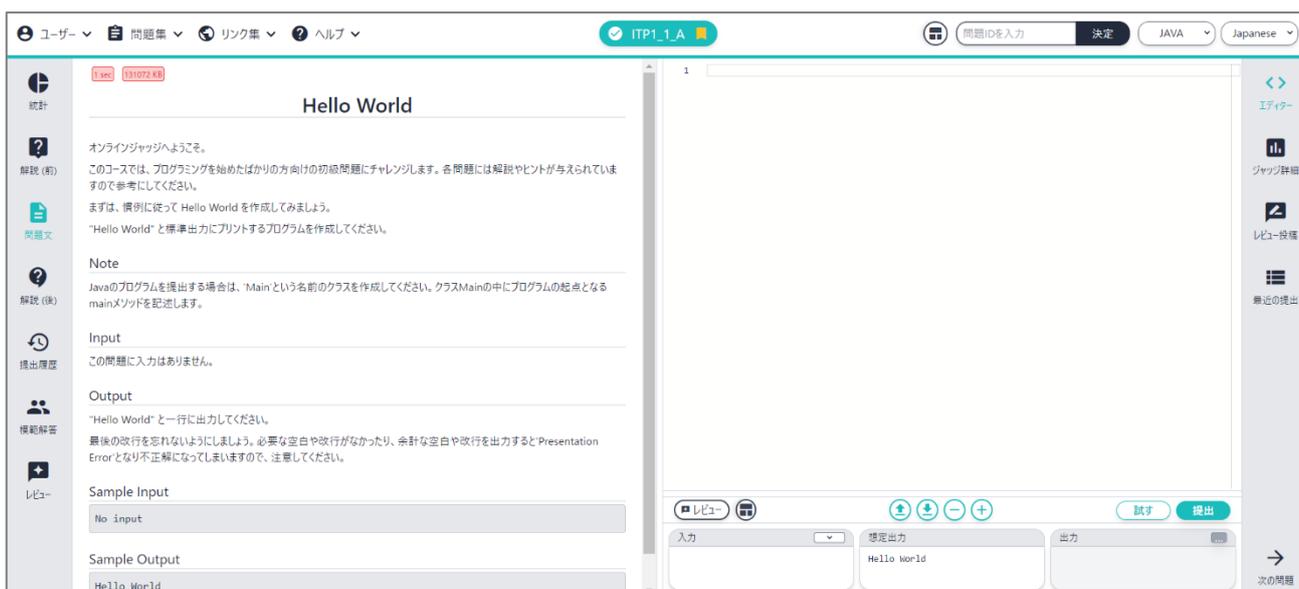
AOJ 3.0 はオンラインコーディングエディタとして実装されています。AOJ 3.0 を通して、以下の作業を画面ひとつで行うことができます。

- 問題の検索
- 問題文の閲覧
- コーディング
- サンプル入出力や指定された入出力を用いた動作確認
- プログラムの提出と判定結果確認
- 学習の進捗を確認
- その他

アカウント登録・ログイン

以下の URL から AOJ 3.0 にアクセスしてみましょう。AOJ 3.0 の全ての機能を使用するには AOJ のアカウントでシステムにログインする必要があります。

<https://onlinejudge.u-aizu.ac.jp/beta/ice/>



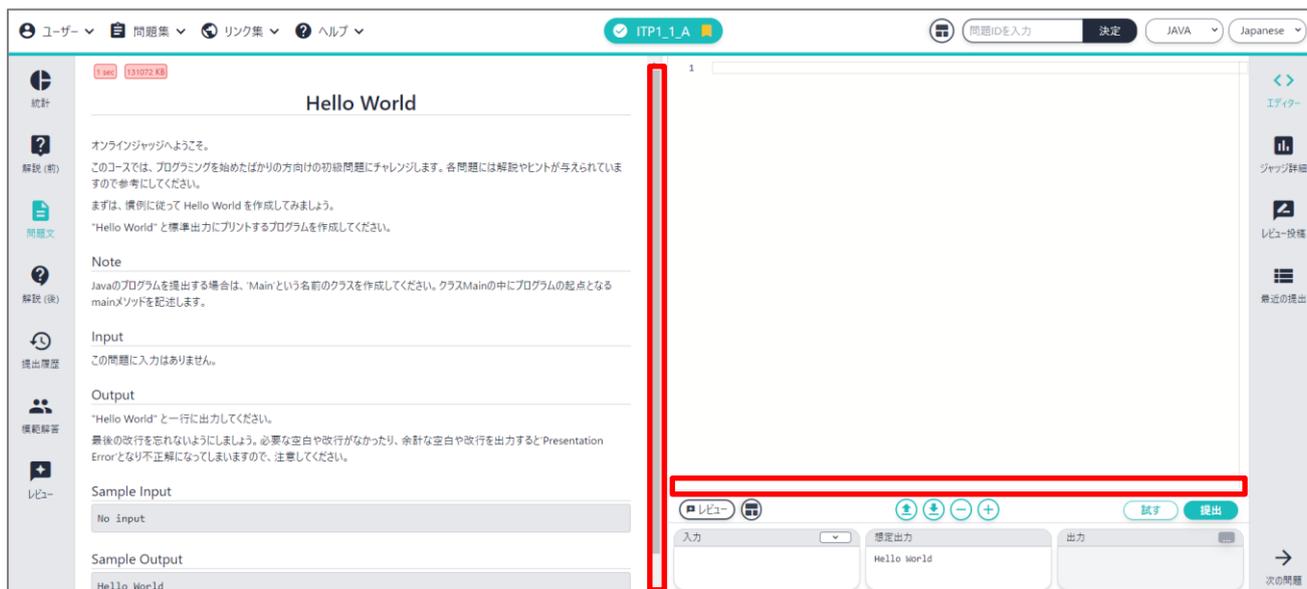
サイトにアクセスした後、

1. 左上のメニューから「ユーザー」→「ログイン」を選択してください。
2. ポップアップが表示されるので、
 - ① アカウントをお持ちでない方は「アカウント新規作成」から AOJ のアカウントを作成してください。
 - ② アカウントをお持ちの方は、ご自分のユーザー ID とパスワードを入力し、ログインしてください。

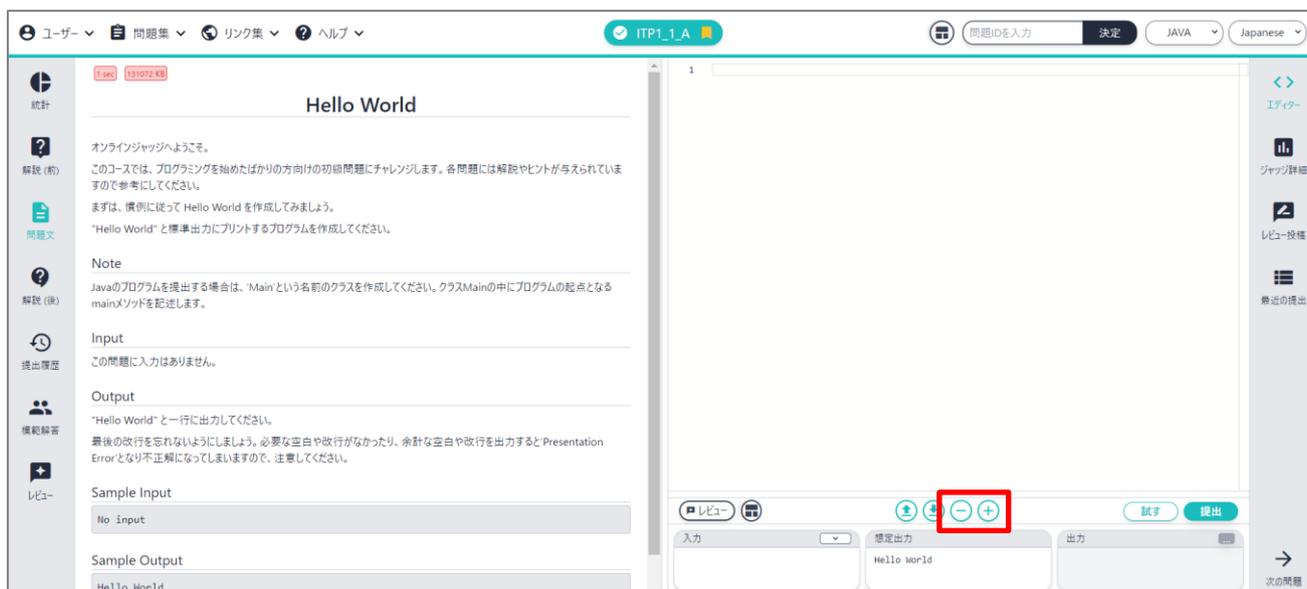


基本操作

画面中央にあるバーをクリックしながら左右に移動させることで、パネルの幅を変更できます。またテキストエディタ下部のバーをクリックしながら上下に移動させると、エディタの高さを変更できます。



エディタ下部の拡大・縮小ボタンをクリックすると、エディタのフォントサイズを変更できます。



設定

「ユーザー」→「設定」からサイトの設定を変更できます。



設定画面ではショートカットキーの変更とダークモードへの切り替えが可能です。

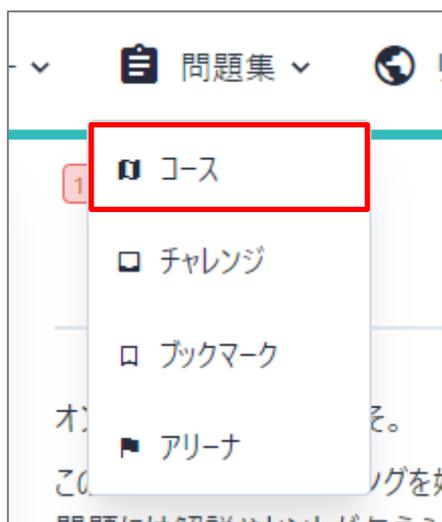
「ジャッジステータスの通知を表示」では、「試す」・「提出」の際に、通知を表示するかどうか変更できます。



問題を選択する

問題を検索して、問題文を開いてみましょう。

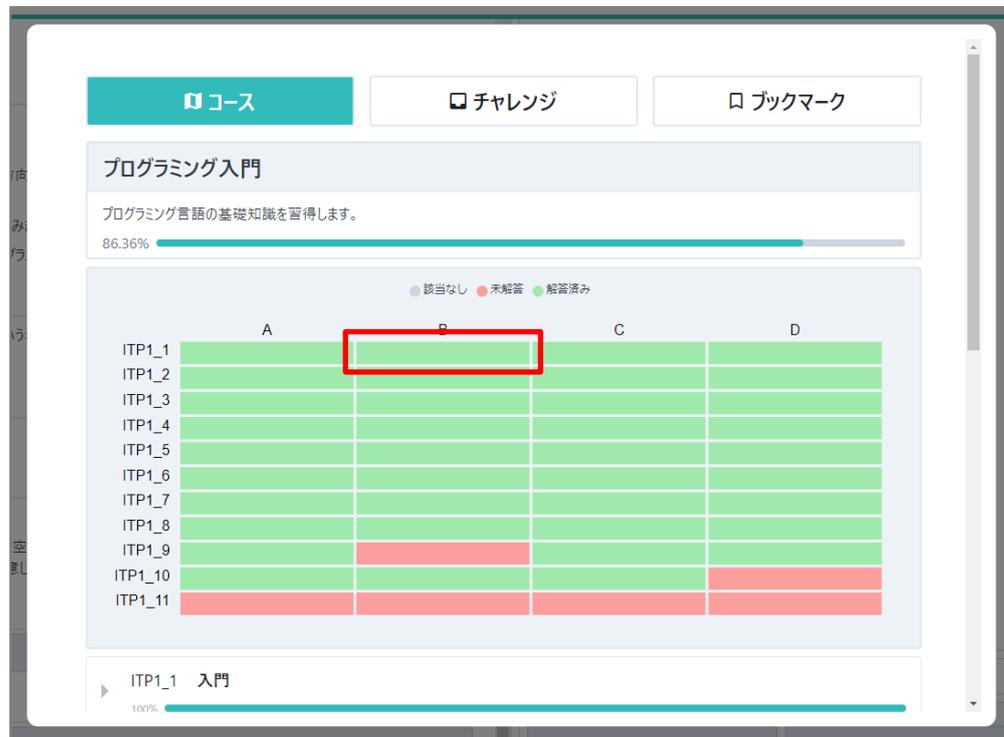
1. 「問題集」 → 「コース」 ・ 「チャレンジ」 ・ 「ブックマーク」 ・ 「アリーナ」 いずれかをクリックすると、問題検索パネルが開きます。



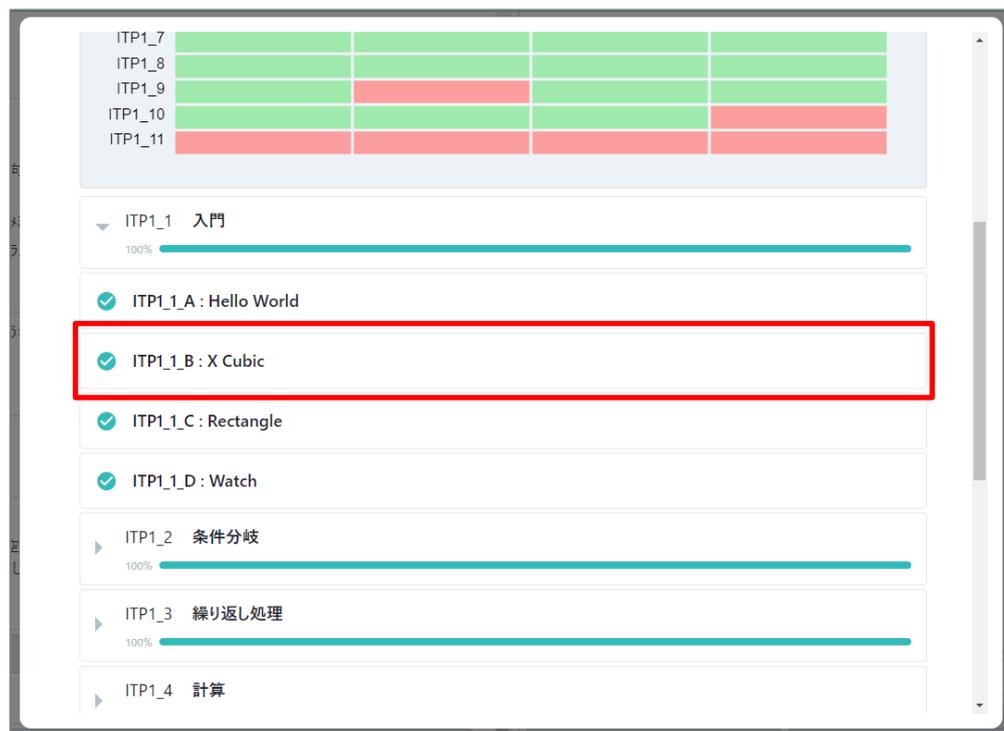
2. 「コース」を選んでみましょう。各コースはカードで表されています。各カードに表示されているプログレスバーは、そのコースの進捗状況を示します。



3. 続いて「プログラミング入門」を開いてみましょう。コースは複数のトピックから構成されています。ヒートマップや各トピックのプログレスバーから進捗を確認できます。



4. ヒートマップ上のセル「ITP1_1_B」または、「ITP1_1 入門」→「X Cubic」を開いてみましょう。



5. 左側のパネルに問題文が表示されます。右側のパネルにはテキストエディタが表示され、下部のコンソールには選択した問題のサンプルケースが自動的に入力されます。

The screenshot shows the AOJ 3.0 Tutorial interface for problem "xの3乗". The interface is divided into several panels:

- Left Panel:** Contains navigation icons and a list of problem details. The problem title "xの3乗" is displayed at the top. Below it, the problem description states: "1つの整数 x を読み込んで、 x の3乗を計算し結果を出力するプログラムを作成して下さい。"
- Input:** "入力として1つの整数 x が与えられます。"
- Output:** " x の3乗の計算結果を1行に出力して下さい。最後の改行を忘れないようにしましょう。"
- Constraints:** "Constraints (制約)には、採点に使用される入力データの上限などの情報が書かれています。変数の型を混同しときや、アルゴリズムを考えると必要な情報になります。この問題に関する制約は以下の通りとなります：
 - $1 \leq x \leq 100$
- Sample Input 1:** 2
- Sample Output 1:** 8
- Sample Input 2:** 3

- Right Panel:** A large text editor area for writing the program code. Below the editor is a control bar with buttons for "レビュー" (Review), "試す" (Test), and "提出" (Submit). At the bottom, there are input fields for "入力" (Input) with the value 2, "想定出力" (Expected Output) with the value 8, and an empty "出力" (Output) field.

コードを書く

テキストエディタでコードを書いてみましょう。



赤い枠で示した部分にプログラムのコードを書きます。または、エディタ下部のアップロードボタンでエクスプローラーを開き、ローカルファイルシステムからソースコードを読み込むこともできます。加えて、ダウンロードボタンから、エディタ上のソースコードをローカルファイルシステムへ保存することができます。

フリースペースモードを選択すると「想定出力」エリアが非表示になり、「出力」エリアに、プログラムの出力のみ表示されます。ジャッジステータスは表示されません。回答プログラムの出力のみ確認したい場合は、フリースペースモードをご利用ください。



ジャッジは、以下のいずれかのステータスを返します。

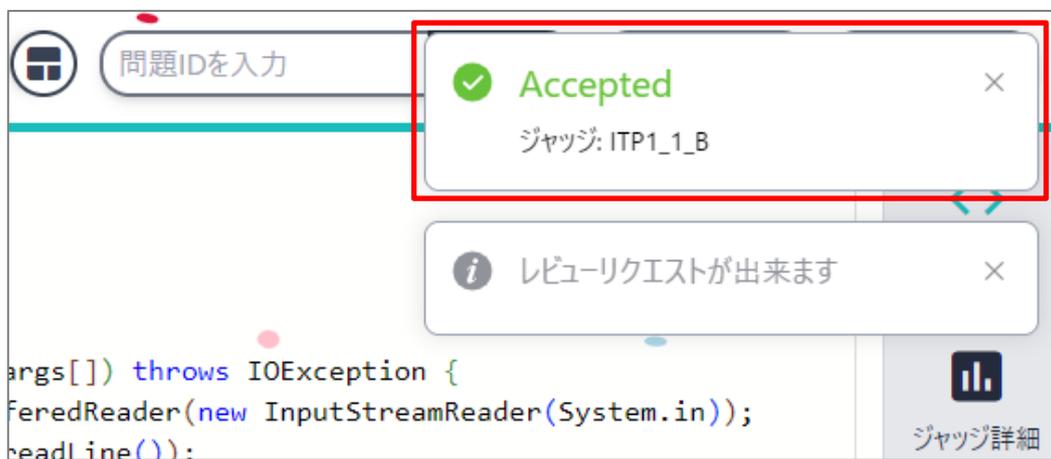
AC 正解 (Accepted)
WA 不正解 (Wrong Answer) <ul style="list-style-type: none">・ ジャッジデータが想定した出力とは異なる出力が行われた
PE 表示形式に誤り (Presentation Error) <ul style="list-style-type: none">・ 余計な空白や改行がある、あるいは必要な空白や改行がない
CE コンパイルエラー (Compile Error) <ul style="list-style-type: none">・ 提出されたプログラムのコンパイルに失敗した
TLE 時間制限超過 (Time Limit Exceeded) <ul style="list-style-type: none">・ 問題で指定された制限時間内にプログラムが終了しなかった
MLE メモリ制限超過 (Memory Limit Exceeded) <ul style="list-style-type: none">・ 問題で指定された以上のメモリを使用した
RE 実行時エラー (Runtime Error) <ul style="list-style-type: none">・ 提出されたプログラムの実行中にエラーが発生した

AOJ へ提出する

テストで正しい結果が得られたら AOJ に提出しましょう。



「提出」ボタンからコードを提出します。提出されたプログラムは、ジャッジデータによって厳格に採点されます。提出結果はサイト上部に通知されます。



ステータスを詳しく確認したい場合は、メニューの「ジャッジの詳細」アイコンをクリックします。テストケースごとの詳しいジャッジの結果を確認することができます。

The screenshot shows the AOJ submission interface. At the top, there is a search bar for the problem ID, a '決定' (Submit) button, and dropdown menus for the programming language (set to 'JAVA') and the language (set to 'Japanese'). Below this, a progress bar shows the submission status: Submitted, Sent to Judge, Build, Run, Resource Limit Check, Result Check, and Presentation Check, all marked with green checkmarks. A summary table shows the language (JAVA), time (0.03 sec), memory (39628 KB), and code length (347 B). A table of test cases is displayed below, with columns for Case, Status, Time, Memory, Input, Output, Case Name, and Attach. The 'Attach' column contains clipboard icons, with the one for case #1 highlighted by a red box. On the right sidebar, the 'ジャッジ詳細' (Judge Details) icon is also highlighted with a red box. Other sidebar options include 'エディター', 'レビュー投稿', '最近の提出', and '次の問題'.

言語	時間	メモリー	コード長
JAVA	0.03 sec	39628 KB	347 B

ケース	ステータス	時間	メモリー	入力	出力	ケース名	貼り付け
# 1	AC	0.03 sec	39108 KB	2 B	2 B	test1	
# 2	AC	0.02 sec	37388 KB	2 B	3 B	test2	
# 3	AC	0.03 sec	39628 KB	3 B	7 B	test3	
# 4	AC	0.02 sec	37096 KB	4 B	8 B	test4	

画面右端の「貼り付け」ボタンをクリックすると、エディタ下部の「入力」エリアと「想定出力」エリアにそれぞれ選択したテストケースの「入力」・「出力」が貼り付けられます。

The screenshot shows the editor interface. At the top, there is a 'レビュー' (Review) button, a search icon, and navigation buttons (up, down, left, right). There are also '試す' (Test) and '提出' (Submit) buttons. Below this, the editor has three main areas: '入力' (Input), '想定出力' (Expected Output), and '出力' (Output). The '入力' area contains the number '2', and the '想定出力' area contains the number '1'. The '出力' area contains the number '8' and is marked with 'AC'. A red box highlights the '入力' and '想定出力' areas. On the right sidebar, there is a '次の問題' (Next Problem) button.

解説を閲覧する

解答プログラムの書き方が分からない場合や、模範解答を知りたい場合は解説を参考にしましょう。メニューの「解説（前）」と「解説（後）」から、解説を表示できます。

「解説（前）」では、主に問題文を読む前に役立つ、文法や方針の説明を閲覧することができます。

ユーザー ▾ 問題集 ▾ リンク集 ▾ ヘルプ ▾

統計

解説(前)

問題文

解説(後)

提出履歴

模範解答

レビュー

前の問題

言語解説 JAVA ▾

変数

コンピュータ（プログラム）に入力された値は一時的にメモリに記録する必要があります。メモリの特定領域を確保して、指定した名前でアクセスするための仕組みが**変数**です。

変数は、そのデータ型（整数、浮動小数点、文字、など）と名前の組で**宣言**する必要があります。たとえば、1つの整数データを保持することができる x という名前の変数は以下のように宣言します。

```
int x;
```

代入演算

変数の役割は、プログラムで扱う値をメモリへ書き込み、必要な時に参照することです。最も基本となる**代入演算**では、 $=$ の左側の変数（メモリ領域）に、右側の式の計算結果が書き込まれます。変数はその名の通り「変わる数」で、代入演算によって値を何度でも変更することができます。一方、式の中に含まれる変数は、それが指すメモリ領域の値を返します。たとえば、次のプログラムは、変数 a , b にそれぞれ 5, 8 を格納し、その後それらの中身を交換するプログラムです。

```
int a, b, t; // 3つの整数を宣言
a = 5;      // a に 5 を格納
b = 8;      // b に 8 を格納
t = a;      // t に a の値を格納
a = b;      // a に b の値を格納、この直後 a の値は 8 になる
b = t;      // b に t の値を格納、この直後 b の値は 5 になる
```

算術演算

プログラミングの目的はコンピュータに**計算**をさせることです。計算の最も基本となるものが**算術演算**です。定数、変数、関数などに対して算術演算を行い計算式を記述します。四則演算はそれぞれ $+$, $-$, $*$, $/$ で行います。たとえば

「解説（後）」では、解答後に役立つ、模範解答や詳しい説明を閲覧することができます。

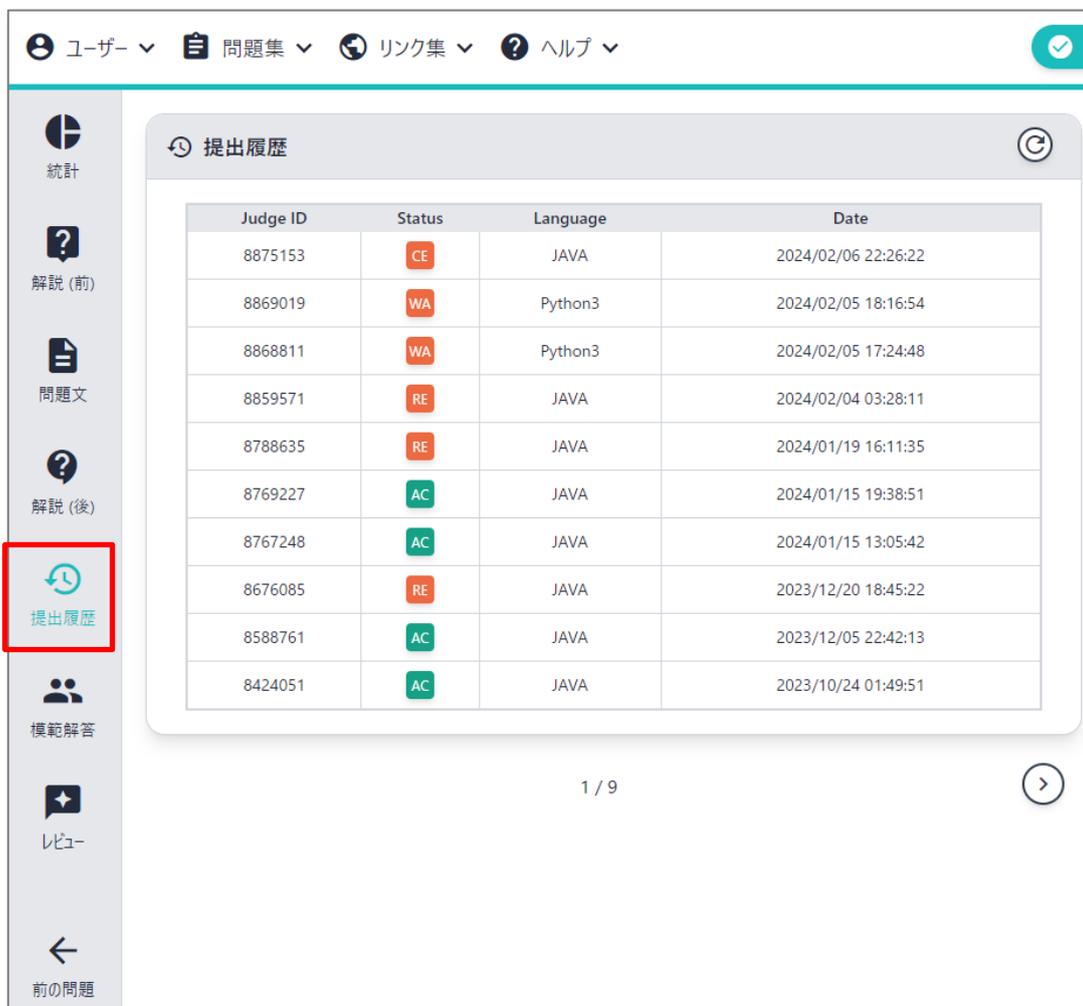
```
import java.util.Scanner;

class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int x = sc.nextInt();
        System.out.println(x * x * x);
    }
}
```

解説には、「言語解説」や「一般解説」などの種類があります。言語解説では、指定したプログラミング言語の文法に関する解説を閲覧することができます。一般解説では、想定するアルゴリズムやヒントを閲覧することができます。（※解説がサポートされていない問題もあります。予めご了承ください）

過去の提出履歴を閲覧する

過去に提出したソースコードを確認できます。メニューの「提出履歴」アイコンをクリックすると、選択中の問題に対する過去の提出履歴を表示することができます。



The screenshot displays the AOJ submission history interface. At the top, there are navigation menus for 'ユーザー' (User), '問題集' (Problem Set), 'リンク集' (Link Set), and 'ヘルプ' (Help). The main content area is titled '提出履歴' (Submission History) and contains a table with the following data:

Judge ID	Status	Language	Date
8875153	CE	JAVA	2024/02/06 22:26:22
8869019	WA	Python3	2024/02/05 18:16:54
8868811	WA	Python3	2024/02/05 17:24:48
8859571	RE	JAVA	2024/02/04 03:28:11
8788635	RE	JAVA	2024/01/19 16:11:35
8769227	AC	JAVA	2024/01/15 19:38:51
8767248	AC	JAVA	2024/01/15 13:05:42
8676085	RE	JAVA	2023/12/20 18:45:22
8588761	AC	JAVA	2023/12/05 22:42:13
8424051	AC	JAVA	2023/10/24 01:49:51

The left sidebar contains several menu items: '統計' (Statistics), '解説 (前)' (Previous Explanation), '問題文' (Problem Statement), '解説 (後)' (Next Explanation), '提出履歴' (Submission History), '模範解答' (Model Answer), 'レビュー' (Review), and '前の問題' (Previous Problem). The '提出履歴' item is highlighted with a red box. At the bottom of the main content area, there is a pagination indicator '1 / 9' and a right arrow button.

リストから特定の提出履歴をクリックすると、そのコードを閲覧することができます。

模範解答を閲覧する

他のユーザーが解答したコードを閲覧することができます。メニューの「模範解答」アイコンをクリックすると、選択中の問題に対する模範解答を表示することができます。

The screenshot shows the AOJ 3.0 interface. At the top, there are navigation menus for 'ユーザー', '問題集', 'リンク集', and 'ヘルプ'. The main content area displays a table of solutions for a problem. The table has columns for Policy, Judge ID, Status, Author, Rating, Language, Time, and Code size. The '模範解答' (Model Solution) icon in the sidebar menu is highlighted with a red box.

Policy	Judge ID	Status	Author	Rating	Language	Time	Code size
	6390975	AC	naoto172	4808.22	JAVA	0.05 sec	246 B
	4913905	AC	mohejin	1845.58	JAVA	0.05 sec	245 B
	2064551	AC	ei1333	1495.71	JAVA	0.03 sec	336 B
	4663439	AC	square1001	1399.36	JAVA	0.08 sec	213 B
	9312919	AC	vjudge4	1066.08	JAVA	0.06 sec	255 B
	9312826	AC	vjudge3	1050.20	JAVA	0.06 sec	211 B
	9242065	AC	vjudge5	1035.77	JAVA	0.05 sec	207 B
	9243649	AC	vjudge2	1034.22	JAVA	0.06 sec	435 B
	9308272	AC	vjudge1	1013.68	JAVA	0.06 sec	453 B
	514140	AC	kyuridenamida	824.39	JAVA	0.09 sec	202 B

1 / 769

解答は各ユーザーのレーティングをもとに降順に表示されます。提出履歴と同様に、リストから模範解答をクリックすると、そのコードを閲覧することができます。「Diff」ボタンをクリックするとご自身のコードと模範解答のコードを比較することができます。

Judge ID : 5924926 By : yutaka

```

1- import java.util.Scanner;
2-
3- class Main{
4-     public static void main(String
5-         Scanner sc = new Scanner(S
6-
7-         int x = sc.nextInt();
8-
9-         System.out.println(x*x*x);
10- }
11- }

```

```

1+ import java.io.BufferedReader;
2+ import java.io.IOException;
3+ import java.io.InputStreamReader;
4-
5+ class Main {
6+     public static void main(String
7+         BufferedReader in = new Bu
8+         int x = Integer.valueOf(ir
9+         x = x * x * x;
10+         System.out.println(x);
11- }
12- }
13+
14+
15+
16+
17+
18+
19+
20+
21+
22+

```

Diff コピー 閉じる

Policy	Judge ID	Status	Author	Rating	Language	Time	Code size
👁	2943892	AC	emtsu_ba	655.62	JAVA	0.06 sec	243 B
👁	2211379	AC	face4	616.96	JAVA	0.05 sec	377 B
👁	5924926	AC	yutaka	604.18	JAVA	0.05 sec	201 B

ブックマーク

ヘッダーにある「ブックマーク」ボタンをクリックすることで、現在の問題を保存できます。

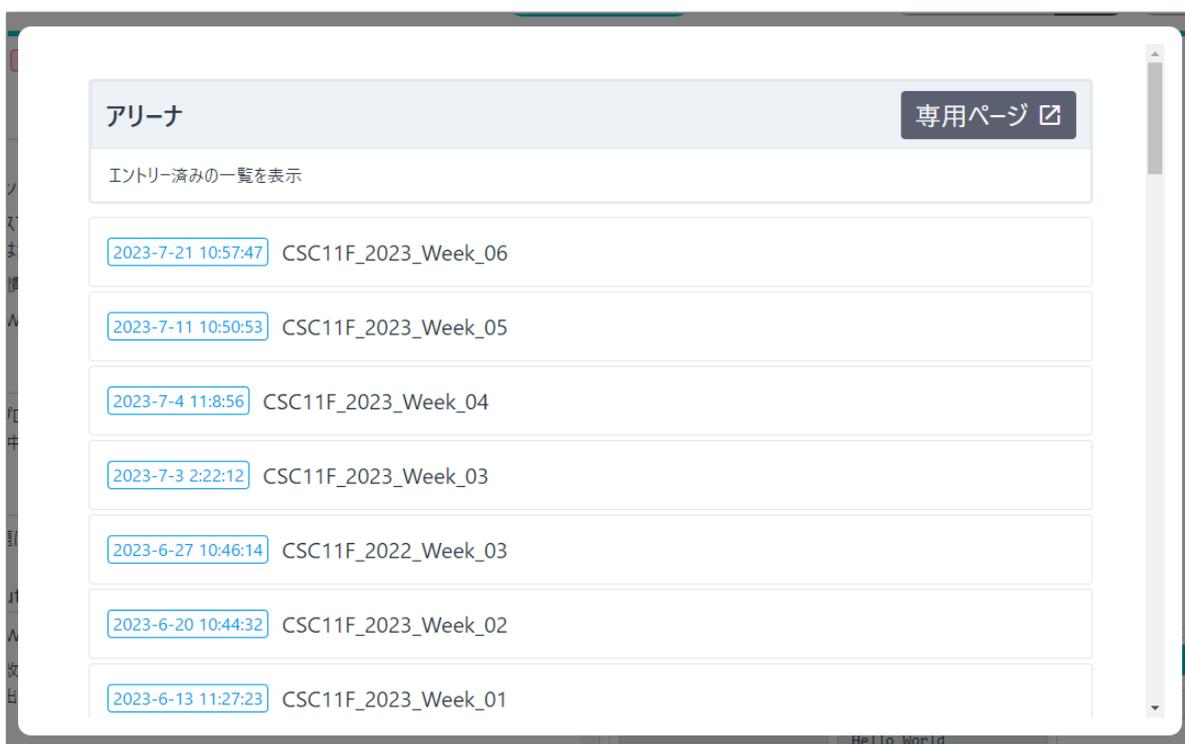


ブックマークした問題は検索パネルから確認できます。

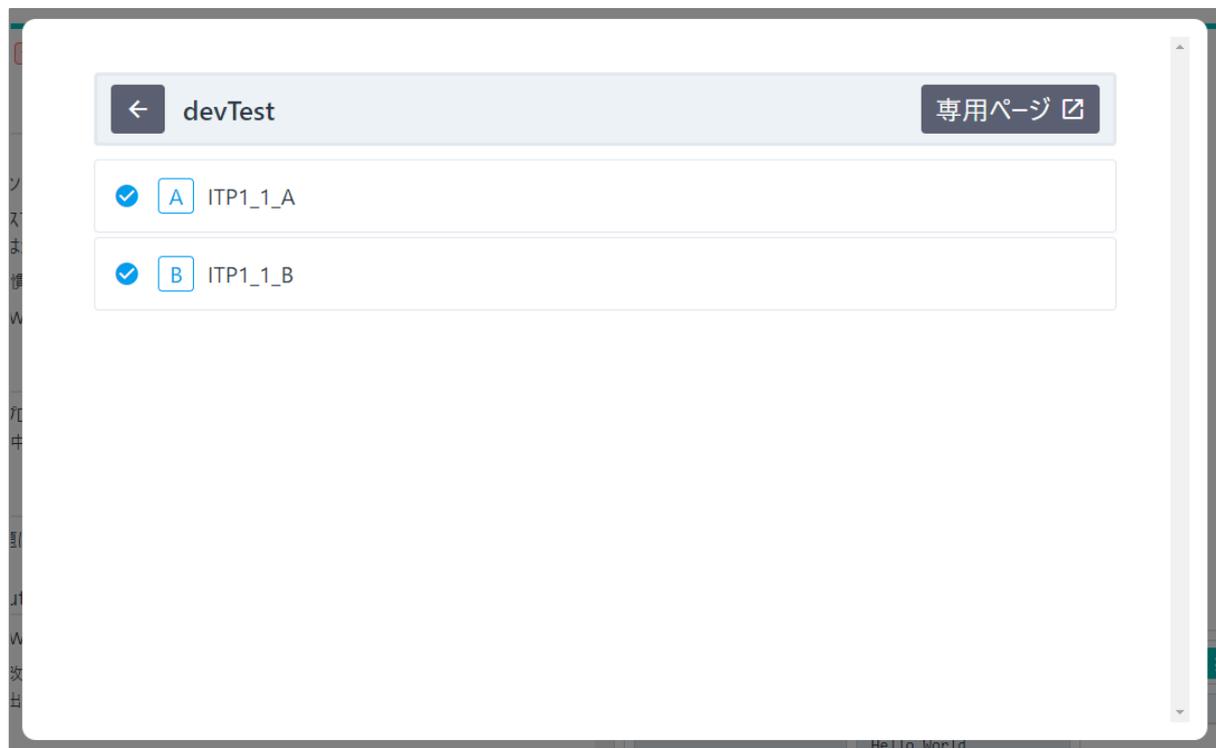


アリーナ

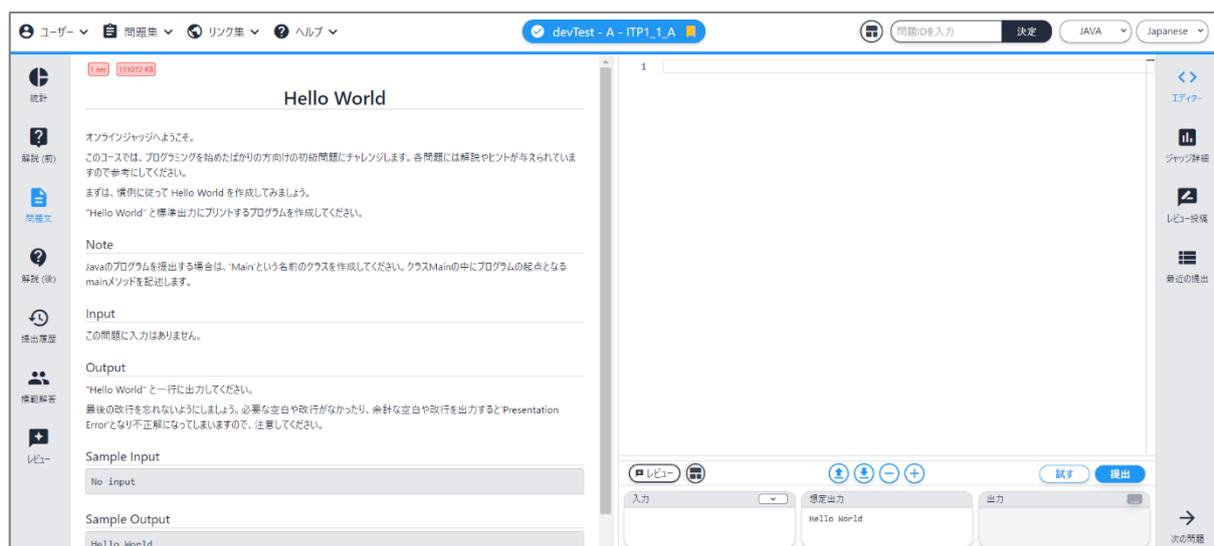
「問題集」 → 「アリーナ」と選択すると、エントリー済みのアリーナー一覧が表示されます。



一覧から特定のコンテストを選択すると、そのコンテストに含まれる問題の一覧が表示されます。「専用ページ」をクリックすると AOJ のアリーナ専用ページにアクセスできます。



一覧から問題を選択すると、サイトカラーが通常とは異なり、水色で表示されます。この状態で AOJ に提出をすると、通常の提出とは異なりコンテストへの提出と判別されます。



前の問題・次の問題

サイドメニューにある「前の問題」「次の問題」ボタンをクリックすると、各コースの前の問題・次の問題が表示されます。

The screenshot shows the AOJ 3.0 interface for the problem "xの3乗" (x^3). The page is divided into several sections:

- Header:** User profile, problem set, link set, help, and problem ID (ITP1_1_B).
- Problem Title:** xの3乗
- Description:** 1つの整数 x を読み込んで、 x の3乗を計算し結果を出力するプログラムを作成して下さい。
- Input:** 入力として1つの整数 x が与えられます。
- Output:** x の3乗の計算結果を1行に出力して下さい。最後の改行を忘れないようにしましょう。
- Constraints:** Constraints (制約)には、採点に使用される入力データの上界などの情報が書かれています。変数の型を混同し、アルゴリズムを考えるときに必要な情報となります。この問題に関する制約は以下の通りとなります：
 - $1 \leq x \leq 100$
- Sample Input 1:** 2
- Sample Output 1:** 8
- Sample Input 2:** 3

At the bottom of the page, there are buttons for "レビュー" (Review), "入力" (Input), "想定出力" (Expected Output), "出力" (Output), "試す" (Try), and "提出" (Submit). The "前の問題" (Previous Problem) button is located on the left sidebar, and the "次の問題" (Next Problem) button is located on the right sidebar. Both buttons are highlighted with red boxes in the image.

問題を検索

問題検索フォームに問題 ID を入力すると、その問題を開くことができます。問題 ID とはヘッダーに表示されている、各問題に割り当てられている ID を指します。



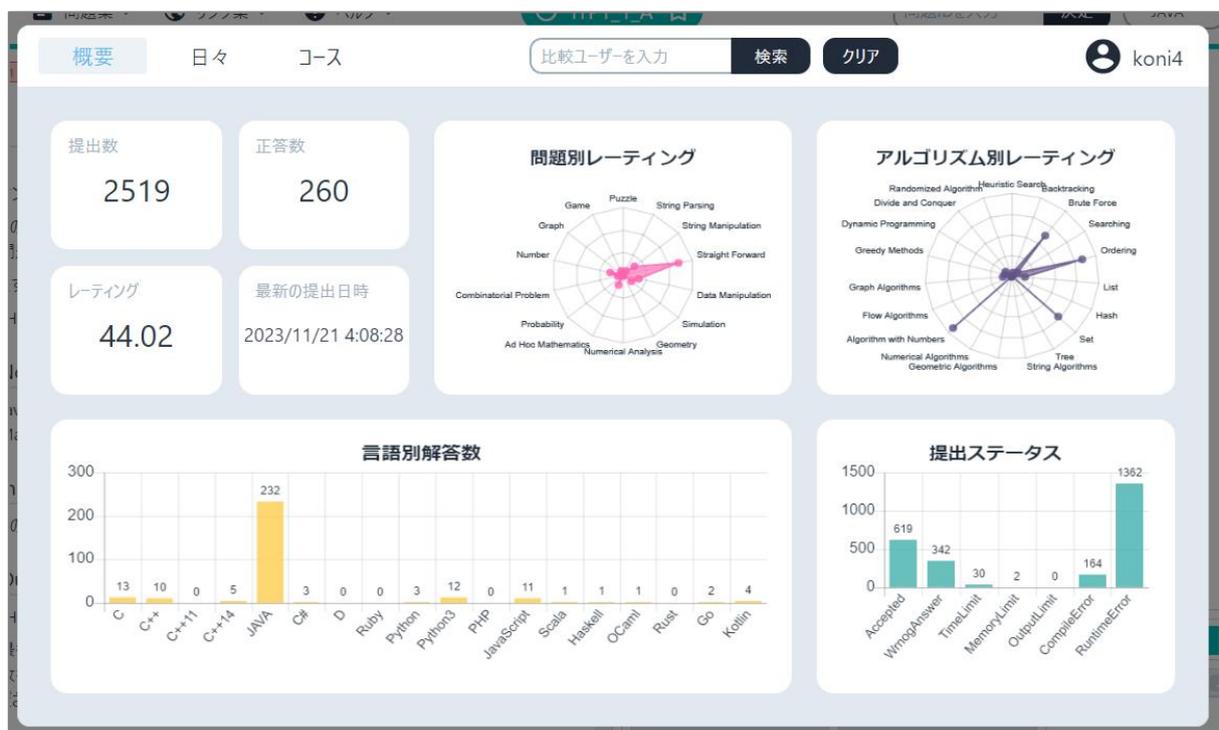
The screenshot shows a search interface with a search bar containing the text "ITP1_1_A" and a search button labeled "問題IDを入力" (Enter problem ID) with a "決定" (Submit) button. Below the search bar, a list of results is shown, with the first item "1" visible.

ダッシュボード

「ユーザー」→「ダッシュボード」をクリックすると、ユーザーの各ステータスがまとめられたダッシュボードを閲覧できます。



ダッシュボードには「ユーザー名」「提出数」「レーティング」などの基本的な情報が表示されます。加えて「直近1年のACヒートマップ」「毎週のレーティング」「言語別解答数」「問題別レーティング」「AOJ各コースの進捗ヒートマップ」等、より複雑な情報を閲覧できます。





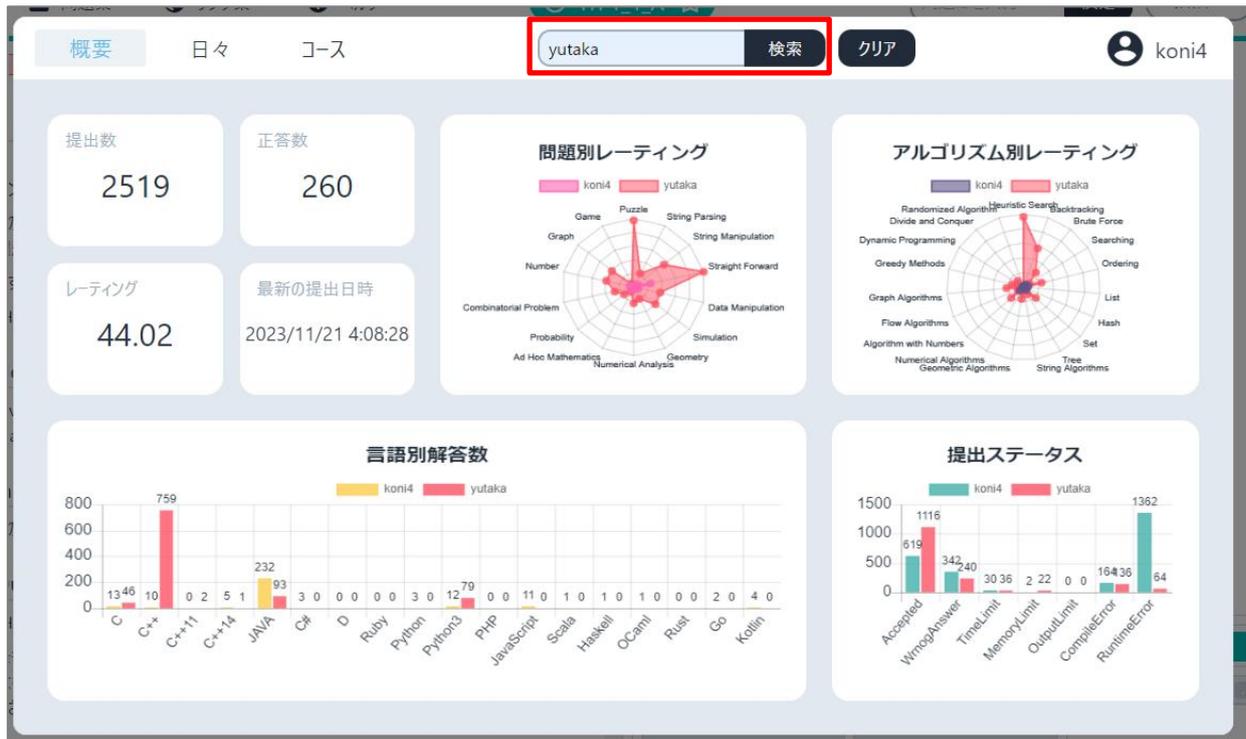
概要 日々 コース 比較ユーザーを入力 検索 クリア koni4

ITP1 ALDS1 ITP2 DSL DPL GRL CGL NTL INFO1

● 該当なし ● 未解答 ● 解答済み

	A	B	C	D
ITP1_1	解答済み	解答済み	解答済み	解答済み
ITP1_2	解答済み	解答済み	解答済み	解答済み
ITP1_3	解答済み	解答済み	解答済み	解答済み
ITP1_4	解答済み	解答済み	解答済み	解答済み
ITP1_5	解答済み	解答済み	解答済み	解答済み
ITP1_6	解答済み	解答済み	解答済み	解答済み
ITP1_7	解答済み	解答済み	解答済み	解答済み
ITP1_8	解答済み	解答済み	解答済み	解答済み
ITP1_9	解答済み	解答済み	解答済み	解答済み
ITP1_10	解答済み	解答済み	解答済み	未解答
ITP1_11	未解答	未解答	未解答	未解答

検索フォームに他ユーザーの ID を入力すると、そのユーザーと比較したステータスが確認できます。



他ユーザーのダッシュボードを表示

他のユーザーのダッシュボードを表示するためには、WebブラウザのURLから、

「<https://onlinejudge.u-aizu.ac.jp/beta/ice?userId=ユーザーID>」のように指定することでダッシュボードを表示できます。例えばユーザーIDが aizutaro の場合、

「<https://onlinejudge.u-aizu.ac.jp/beta/ice?userId=aizutaro>」と指定します。

指定したURLにアクセスすると、他のユーザーのダッシュボードが表示されます。

加えて、ユーザーID aizutaro のコース ITP1 のヒートマップを表示したい場合、

「<https://onlinejudge.u-aizu.ac.jp/beta/ice?userId=aizutaro&courseName=ITP1>」と指定します。

指定したURLにアクセスすると、ユーザーID aizutaro の ITP1 に関するヒートマップが表示されます。

フリースペース

画面上部のフリースペースモードを選択すると、フリースペース画面が表示されます。



この画面では問題に依存しない自由なプログラムのテストが可能です。

※プログラムデータサイズや入力データサイズに制限があります。

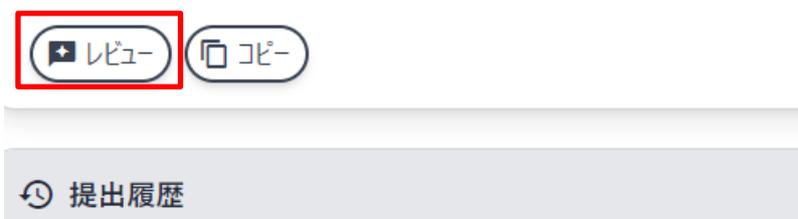


レビュー機能

エディタ下部や提出履歴画面に表示されているレビューボタンをクリックすると、レビューリクエスト画面が表示されます。



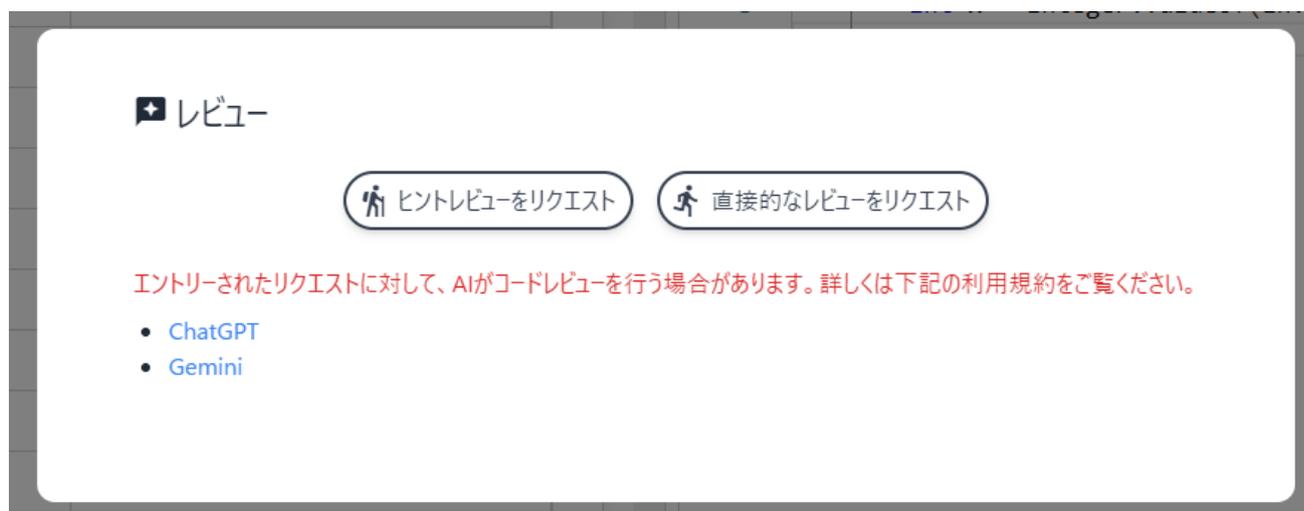
エントリーされたリクエストに対して、AIがコードレビューを行う場合があります。



レビュー機能を使用することで、レビュアーからプログラムに含まれるバグや改善点に関するフィードバックを得ることができます。レビューの種類について以下の違いがあります。

「ヒントレビューをリクエスト」：解答につながるヒントを得る

「直接的なレビューをリクエスト」：解答・解法を得る



レビュー内容を確認する場合は、メニューの「レビュー」アイコンをクリックします。

「オープンエントリー」には、現在レビューを受け付けているレビューエントリーが表示されます。

「クローズエントリー」にはすでにレビュー受付を終了したレビューエントリーが表示されます。

-  統計
-  解説 (前)
-  問題文
-  解説 (後)
-  提出履歴
-  模範解答
-  **レビュー**

オープンエントリー

ジャッジID	ステータス	言語	インストラクション	エントリー日時
9264266	AC	JAVA	Passive	2024/05/29 13:14:10

クローズエントリー

ジャッジID	ステータス	言語	インストラクション	エントリー日時
9052951	RE	JAVA	Passive	2024/04/02 20:05:52
7604146	RE	Python	Passive	2024/03/16 03:31:34
9052951	RE	JAVA	Passive	2024/04/02 20:03:58
7604151	WA	Python	Active	2024/02/03 04:46:38
7604163	AC	Python	Active	2024/02/03 02:05:33

レビューエントリーから投稿されたレビューを選択すると、レビュー内容が表示されます。ユーザーは投稿されたレビューに対して5段階評価を行うことができます。「エントリーをクローズ」をクリックするとレビューエントリーはクローズされ、これ以上レビューを受け取らなくなります。

レビュー

ReviewID : bb1ff444-3a88-443a-809d-2409a78bcff2 By : gpt-3.5-turbo

このレビューはAIが行いました。

ヒント :

- 変数名をより意味のあるものに変更すると、プログラムの可読性が向上します。例えば、変数aをheight、bをwidth、msをarea、nsをperimeterというように変更することができます。
- コードの一部を関数に切り出すことで、コードをより分かりやすくすることができます。たとえば、面積を計算する部分をcalculateArea()関数に、周の長さを計算する部分をcalculatePerimeter()関数に切り出すことができます。

5 4 3 2 1

エントリーをクローズ

メニューの「レビュー投稿」アイコンをクリックすると、他のユーザーのレビューエントリーに対して、レビューを投稿することができます。

新規レビューを投稿

ジャッジID	ユーザー	ステータス	言語	インストラクション
9264266	koni4	AC	JAVA	Passive

<> ソースコード:

```
import java.util.Scanner;

class Main{
public static void main(String[] args){
Scanner in = new Scanner(System.in);
int a,b,ms,ns;
a=in.nextInt();
b=in.nextInt();
ms=a*b;
ns=a*b*2;
```

レビューエディター:

レビュー投稿

新規レビューを投稿

ショートカットキー

AOJ 3.0 では以下のショートカットキーをサポートしています。

- Alt+Shift+X: 試す (デフォルト)
- Alt+X: 提出 (デフォルト)
- Ctrl+P: コースパネルを表示
- Ctrl+` : チャレンジパネルを表示
- Ctrl+{ : ブックマークパネルを表示
- Ctrl+M: ダッシュボード表示
- Ctrl+F: 次の問題
- Ctrl+D: 前の問題
- Ctrl+1, 2 ...9, 0: サンプルケースを選択
- Ctrl+I: 右側パネルの選択を上へ移動
- Ctrl+K: 右側パネルの選択を下へ移動
- Ctrl+U: 左側パネルの選択を上へ移動
- Ctrl+J: 左側パネルの選択を下へ移動
- Ctrl+Shift++: フォント拡大
- Ctrl+-: フォント縮小

お問い合わせ

本システムは現在開発中です。バグのご報告や、ご提案・ご要望は uoaojs@gmail.com または yutaka@u-aizu.ac.jp までご連絡ください。