



## AOJ チュートリアルドキュメント

<http://judge.u-aizu.ac.jp/onlinejudge/tutorial.pdf>

# Welcome to Aizu Online Judge (AOJ)

AOJ へようこそ！。このチュートリアルドキュメントでは、AOJ (Aizu Online Judge) の使い方を説明します。ユーザの目的や経験に応じていくつかの章に分かれていますので、目次から必要な情報を確認してください。

なお、AOJ は未だ開発途上のため、本チュートリアルドキュメントもシステムの更新に伴い頻繁に更新される可能性があります。あらかじめご了承ください。

## AOJ チュートリアルドキュメント アウトライン

### 1. イントロダクション

- Chapter 1-1 ユーザ登録
- Chapter 1-2 ユーザ登録情報を更新する
- Chapter 1-3 ログインする

### 2. 問題を解く

- Chapter 2-1 基本メニュー
- Chapter 2-2 問題の一覧を見る
- Chapter 2-3 問題を閲覧する
- Chapter 2-4 プログラムを提出する
- Chapter 2-5 審判結果を閲覧する
- Chapter 2-6 審判結果の詳細を確認する
- Chapter 2-7 コンパイラ・システムの詳細を確認する

### 3. ユーザステータス

- Chapter 3-1 ユーザステータスを閲覧する
- Chapter 3-2 ランキングを閲覧する
- Chapter 3-3 問題ごとのランキングを閲覧する

### 4. ソースコードを閲覧・管理する

- Chapter 4-1 公開されているソースコードを閲覧する
- Chapter 4-2 自分のソースコードを公開・非公開設定する
- Chapter 4-3 ソースコードの管理

### 5. コース

- Chapter 5-1 コースの選択
- Chapter 5-2 トピックの進め方

## 6. 掲示板 (Web board)

Chapter 6-1 メッセージを閲覧・投稿する

Chapter 6-2 スレッドを投稿する

Chapter 6-3 メッセージ・スレッドを削除する

## 7. コンテスト

Chapter 7-1 コンテストに参加する

Chapter 7-2 問題を解く

Chapter 7-3 ランキング

Chapter 7-4 質問

Chapter 7-5 統計情報

## 8. タグ

Chapter 8-1 カテゴリーの分類

Chapter 8-2 タグを設定する

## 9. ツール

Chapter 9-1 オンラインエディタ

Chapter 9-2 ツールの紹介

## 10. 開発 (API)

# Welcome to Aizu Online Judge (AOJ)

AOJ は提出されたプログラムの正しさ・効率の自動判定を行うオンラインジャッジシステム（WEB アプリケーション）です。24時間インターネットからサイトにアクセスし自由に自分のペースで学習し・遊ぶことができます。

各問題に対するプログラムのソースコードを提出すると、「問題の仕様に基いた入力データに対して正しい出力を行っているか」、「指定された資源（CPU 時間、メモリ使用量）内で処理を行っているか」を審判データと検証器で自動判定を行い、即座に結果をフィードバックします。

AOJ には SNS の要素も含まれています。他のユーザと正解問題数や処理速度で競ったり、公開されているソースコードや、掲示板での質問・議論を通して効率的に学習を進めることができます。

それでは、下記の URL から AOJ のページにアクセスしましょう：

AOJ URL <http://judge.u-aizu.ac.jp>

**AIZU ONLINE JUDGE**

HOME -

AOJ: Online Programming Challenge

**A contest is coming!**  
Ritsumeikan University Competitive Programming Camp,  
Day 3, with problem set from Ritsumeikan University teams  
**2013/03/13 10:00-13:00 JST**

**Recent Activity**

Author	Problem	Status	Lang	Time
e2300x0	0025: Hit and Blow	WA	JAVA	Mon Apr 08 09:57:10
e2300x0	0024: Physical Experiments	AC	JAVA	Mon Apr 08 09:48:19
berlysta	0008: Sum of 4 Integers	AC	Python	Mon Apr 08 09:40:12
core2duoe6320	0009: Prime Number	TL	C++	Mon Apr 08 09:35:29
div9851	0207: Block	AC	C++	Mon Apr 08 09:32:13
div9851	0207: Block	CE	C++	Mon Apr 08 09:31:37
div9851	0207: Block	WA	C++	Mon Apr 08 09:14:23
core2duoe6320	0008: Sum of 4 Integers	AC	C++	Mon Apr 08 08:41:46
Mis877	0201: Wrought Gold Master	AC	C	Mon Apr 08 08:40:35
core2duoe6320	0007: Debt Hell	AC	C++	Mon Apr 08 08:36:34
core2duoe6320	0007: Debt Hell	WA	C++	Mon Apr 08 08:35:24
ltzz	0588: Simple Calculator	AC	D	Mon Apr 08 08:32:51

more information judge replies

**Announcements and Bug reports**

Welcome to AOJ!

For beginners

**Recent messages on the web board**

- 0099 2013-04-03 11:08:48.0
- 0099 2013-04-02 11:58:04.0
- 0097 2013-03-31 23:47:46.0
- 0097 2013-03-31 23:38:09.0
- 10000 2013-03-27 11:06:54.0
- 2504 2013-03-22 21:53:47.0
- 2504 2013-03-21 22:45:32.0
- 2500 2013-03-14 21:13:55.0
- 2500 2013-03-14 20:47:46.0
- 10001 2013-03-10 10:00:30.0

まずは、ページ右上の国旗のアイコンをクリックして、サイトで使用される言語を設定してください。日本語または英語を選択することができます。AOJ のホームには、開催予定のイベント（コンテスト等）、最近の提出状況、お知らせ、掲示板の最新書き込み情報などが表示されています。

次章から AOJ の登録方法から使い方を説明していきます。



# 1 イン트로ダクション

Introduction to AOJ

AOJにプログラムを提出するためには、ユーザ登録を行う必要があります。この章では、AOJを利用するためのユーザ登録について説明します。

- Chapter 1-1 ユーザ登録をする
- Chapter 1-2 ユーザ情報を更新する
- Chapter 1-3 ログインする

## 1-1 ユーザ登録をする

自分の提出したプログラムと他の人が提出したプログラムを区別するために、AOJ ではユーザ登録を行う必要があります。

ユーザ登録はページ上部のヘッダメニュー右上の「登録・設定」ボタンから行います。ボタンを押すと小さい窓が現れるのでさらに「登録・設定」リンクへ進んで下さい。以下のような新規ユーザ登録ページに移ります。



「登録・設定」を選択すると、以下のような新規ユーザ登録ページに移ります。

HOME - Register

### 新規ユーザ登録

登録前に、リンクに記載の説明と注意をお読み下さい。

プログラムの提出に関する注意事項: [Click here](#)

ユーザ ID:	<input type="text"/>	*半角アルファベット、半角数字、半角記号
パスワード:	<input type="password"/>	*4文字以上
パスワード(確認):	<input type="password"/>	
名前:	<input type="text"/>	*半角アルファベット、半角数字、半角記号、空白
所属:	<input type="text"/>	*半角アルファベット、半角数字、半角記号、空白
メールアドレス:	<input type="text"/>	*オプション
URL:	<input type="text"/>	*あなたのホームページやBLOG
提出ソースコードの公開:	<input type="text" value="private"/>	*public:全ての 人に公開します、private:非公開
	<input type="button" value="送信"/>	

ユーザ登録を行う前に、右上リンクより、プログラムの提出に関する注意事項を確認して下さい。必要事項を入力し、送信ボタンを押します。正しい入力が行われ「ご登録ありがとうございます」というメッセージが表示されると登録完了となります。各項目の詳細は以下の通りです：



## ユーザ ID

AOJ でユーザを識別するための ID です。既に登録されている他のユーザと重複することはできません。ニックネームや学籍番号などを使用しましょう。半角アルファベット、半角数字、半角記号を使用することができます。全角の日本語や特殊記号を使用することはできません。

## パスワード

ログインユーザを認証するためのパスワードです。半角アルファベット、半角数字で構成された 4 文字以上の文字列を使用して下さい。パスワードに特殊文字を含めることはできません。

## パスワード（確認）

確認のためにもう一度パスワードを入力します。

User Registration

## 名前

ユーザの名前です。ユーザ ID とは別の文字列を設定することができます。他のユーザと重複することもできます。本名やニックネームなどを登録して下さい。半角アルファベット、半角数字、半角記号、空白を使用することができます。全角の日本語や特殊記号を使用することはできません。

## 所属

ユーザの所属です。学校名や会社名などを入力して下さい。半角アルファベット、半角数字、半角記号、空白を使用することができます。全角の日本語や特殊記号を使用することはできません。

## メールアドレス

メールアドレスは必須項目ではありません。AOJ からのお知らせや、パスワードを忘れてしまった場合に使用される可能性があります。

## URL

ホームページやブログなどの URL を登録することができます。自分のプロフィール等にリンクが表示されます。

## 提出ソースコードの公開

自分が AOJ に提出したソースコード（過去のものを含む）を他のユーザに公開するかを設定します。private は非公開で、自分のみ自分が提出したソースコードを閲覧することができます。public は公開で、すべての人に自分が提出したソースコードが公開されます。

### 1-2 ユーザ情報を更新する

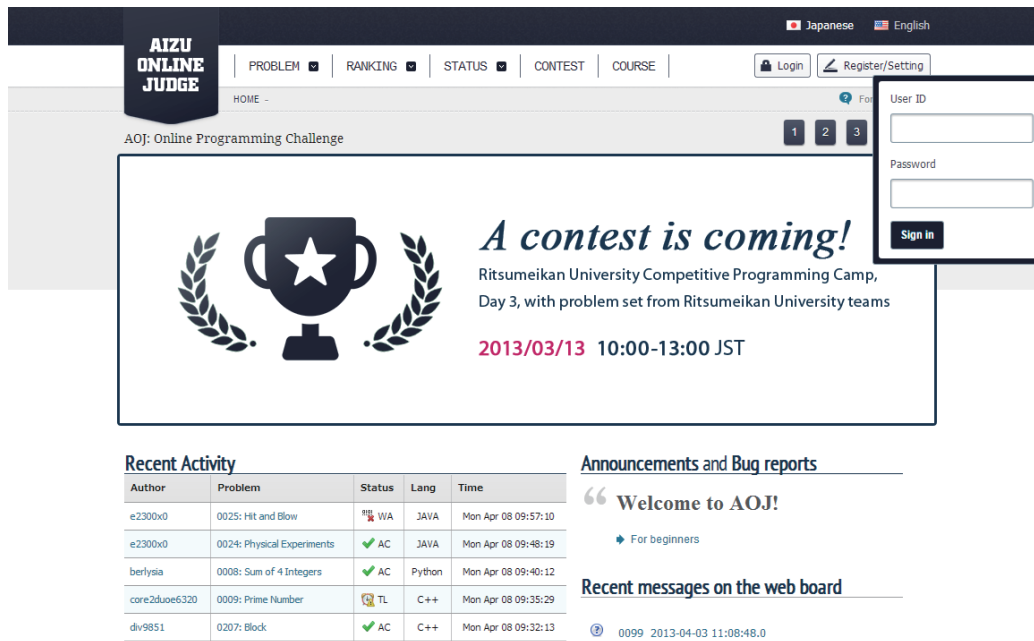
登録後は、必要ときにユーザ情報を更新することができます。

ユーザ情報の更新は、ページ上部ヘッダメニュー右上の「登録・設定」ボタンから行います。ボタンを押すと小さい窓が現れるのでさらに「登録情報更新」リンクへ進んで下さい。ユーザ登録情報更新画面に移ります。

## 1-3 ログインする

この章では、AOJ にログインする方法とその利点を確認します。

ヘッダメニューのログインボタンを押すとユーザ ID とパスワードを入力する小さい窓が現れます。入力したら「Sign in」（ログイン） ボタンをクリックしてログインします。



AOJ では、プログラムの提出時には登録したユーザ ID とパスワードの入力が必要ですが、問題の観覧などの基本的な機能を利用するために必ずしもログインする必要はありません。ログインすると以下の機能面で便利に AOJ を利用することができます。

- プログラムの提出時にユーザ ID とパスワードを毎回入力する必要がありません。
- 問題リストのページで、自分が既に解いた問題をチェックマークで把握することができます。
- コースの問題に解答することができます。
- 問題にアノテーション（カテゴリとタグ）を付けることができます。
- 非公開の自分のソースコードにアクセスすることができます。
- 掲示板にメッセージを投稿することができます。



## 2 問題を解く

Solving problems

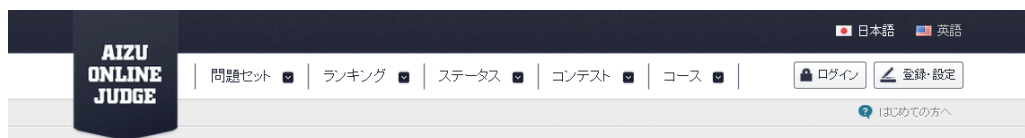
この章では、問題の閲覧やプログラムの提出方法など、AOJ を使用するための基本的な機能を説明します。

- Chapter 2-1 基本メニュー
- Chapter 2-2 問題の一覧を見る
- Chapter 2-3 問題を閲覧する
- Chapter 2-4 プログラムを提出する
- Chapter 2-5 審査結果を閲覧する
- Chapter 2-6 審査結果の詳細を確認する
- Chapter 2-7 コンパイラ・システムの詳細を確認する

## 2-1 基本メニュー

この章では、目的ごとに分けられた AOJ の基本メニューを確認します。

ページ上部のヘッダリンクから各メニューを選択することができます。



各メニューは以下のページへのリンクになっています。メニューにマウスをのせるとサブメニューが開きます。

### 問題セット

問題がまとめられた Volume のリストです。詳しくは 2-2 章を参考にしてください。

### ランキング

サイト内ユーザのランキング（順位）ページです。詳しくは第 3 章を参考にしてください。

### ステータス

提出プログラムの判定結果一覧です。詳しくは 2-5 章を参考にしてください。

### コンテスト

AOJ で過去に開催されたプログラミングコンテスト、開催予定のコンテスト、開催中のコンテストの情報を閲覧することができます。詳しくは第 7 章を参考にしてください。

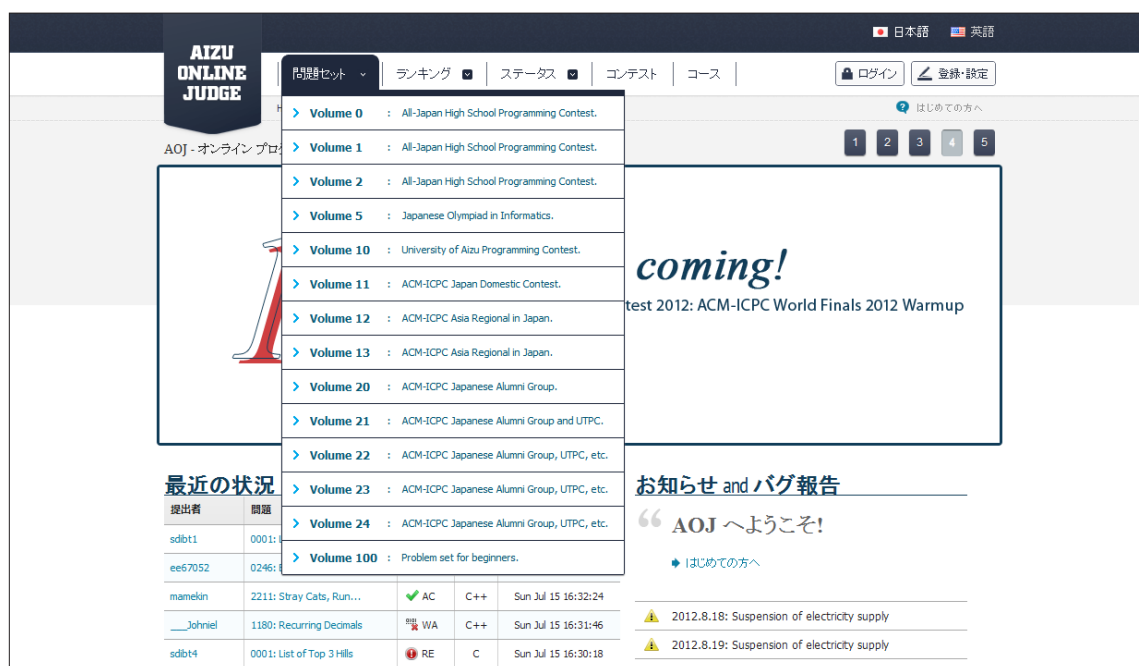
### コース

アルゴリズムやプログラミングの導入として、基本的な問題をステップ b y ステップで解いていくためのコースです。また、特定トピックの問題セットも準備されています。詳しくは第 5 章を参考にしてください。

## 2-2 問題の一覧を見る

問題の一覧から問題を選択する方法を確認します。

メニューの「問題セット」にマウスポインタを合わせると、出典やレベルによって問題がまとめられているボリュームのリストが表示されます。Volume をクリックすると問題一覧のページへ移動することができます。



AOJ では、プログラミングコンテストの過去問等を利用して頂いております：

- Volume 0 - 3 には全校高等学校パソコンコンクール・プログラミング部門の過去問を掲載しています。
- Volume 5, 6 には、日本情報オリンピックの過去問を掲載させて頂いています。
- Volume 10, 15 には AOJ で開催されたコンテストの過去問が掲載されています。
- Volume 11-13 には ACM-ICPC の過去問を掲載させて頂いています。
- Volume 20 - には、ACM-ICPC OB-OG 会が主催したコンテストや、大学が開催したコンテストの過去問を掲載させて頂いています。
- Volume 100 には公開講座や AOJ のコースで使用されたプログラミング入門者・アルゴリズム入門者向けの問題が掲載されています。初めての方は、コースのトピック問題にチャレンジして下さい。

例えば、Volume 10 をクリックすると問題番号が 1000 番台の問題リストが表示されます。

問題集 from Volume 10

Volume : Source :

ID Solved

< prev | 1 / 7 | next > per-page : 10 20 30 50 100

		タイトル	時間	メモリ	Sub	成功率	Source	
		1009: Greatest Common Divisor	1 sec	64 MB	618	<div><div></div></div> 69.42 %  x 373	UAPC 2003	
		1010: Dominoes Arrangement	1 sec	64 MB	293	<div><div></div></div> 29.01 %  x 77	UAPC 2003	
		1011: Finding the Largest Carbon Compo...	1 sec	64 MB	188	<div><div></div></div> 44.15 %  x 73	UAPC 2003	
		1012: Operations with Finite Sets	1 sec	64 MB	284	<div><div></div></div> 29.23 %  x 65	UAPC 2003	
		1013: Combinatorial Topology	5 sec	64 MB	61	<div><div></div></div> 01.64 %  x 1	UAPC 2003	
		1014: Computation of Minimum Length of...	1 sec	64 MB	164	<div><div></div></div> 62.20 %  x 77	UAPC 2003	✓
		1015: Dominating Set	5 sec	64 MB	203	<div><div></div></div> 28.57 %  x 45	UAPC 2003	
		1016: Fibonacci Sets	1 sec	64 MB	344	<div><div></div></div> 47.09 %  x 130	UAPC 2003	✓
		1017: Riffle Shuffle	1 sec	64 MB	162	<div><div></div></div> 75.93 %  x 114	UAPC 2003++	✓
		1018: Cheating on ICPC	1 sec	64 MB	294	<div><div></div></div> 80.27 %  x 194	UAPC 2003++	

< prev | 1 / 7 | next >

また、ページ右上の Source、Volume プルダウンメニューから出典と Volume を指定して問題を絞り込むことができます。例えば、Source から "JOI" を選択すると情報オリンピックの過去問リストが表示されます。

問題集 from JOI

Volume : Source :

ID Solved

< prev | 1 / 10 | next > per-page : 10 20 30 50 100






		タイトル	時間	メモリ	Sub	成功率	Source	
		0500: Card Game	1 sec	64 MB	1930	<div><div></div></div> 55.54 %  x 895	JOI 2005	✓
		0501: Data Conversion	1 sec	64 MB	1910	<div><div></div></div> 42.72 %  x 615	JOI 2005	✓
		0502: Dice	1 sec	64 MB	1309	<div><div></div></div> 64.71 %  x 583	JOI 2005	
		0503: Cup	5 sec	64 MB	613	<div><div></div></div> 34.91 %  x 152	JOI 2005	
		0504: Card Game II	5 sec	64 MB	161	<div><div></div></div> 38.51 %  x 49	JOI 2005	
		0505: Questionnaire	1 sec	64 MB	1145	<div><div></div></div> 44.54 %  x 425	JOI 2005	
		0506: String	1 sec	64 MB	924	<div><div></div></div> 43.72 %  x 342	JOI 2005	
		0507: Square	1 sec	64 MB	673	<div><div></div></div> 47.55 %  x 252	JOI 2005	✓
		0508: String With Rings	8 sec	64 MB	598	<div><div></div></div> 19.90 %  x 76	JOI 2005	
		0509: Sheets	5 sec	64 MB	352	<div><div></div></div> 19.60 %  x 52	JOI 2005	

< prev | 1 / 10 | next >

Volume テーブルは以下のカラムから構成されています：

### 審判のタイプ

このカラムには、自動審判の種類が示されています。各 Volume テーブル上部の旗で示されているように、AOJ では以下の 5 つのタイプの自動判定を行います。

色	名称	説明
	通常	ユーザプログラムの出力結果と審判データを比べます。出力が完全に一致しなければなりません。
	許容誤差	出力に小さな誤差を許す判定を行います。許容される誤差の範囲は問題文に記載されています。
	検査器	問題ごとに特別に用意された検査器で検証されます。入力に対する正しい出力が 1 通りではない場合に採用されます。
	リアクティブ	ユーザプログラムと対話的に反応する判定プログラムが実行されます。
	準備中	準備中のため判定不可能な問題です。

### 問題文の言語

このカラムには、問題文で使用されている言語が表示されます。色付きの地球マークは英語、日の丸マークは日本語がサポートされていることを示します。

### タイトル

このカラムには、問題 ID と問題名が表示されます。固有の問題 ID によって AOJ の問題が識別されます。タイトルをクリックすると問題ページに移動します。

### 時間制限

このカラムには、問題ごとに設定された制限時間が表示されます。プログラムの CPU 時間がこの値を超えてしまうと不正解となります。

### メモリ制限

このカラムには、問題ごとに設定された制限メモリ使用量が表示されます。プログラムの使用メモリ量がこの値を超えてしまうと不正解となります。

### 成功率

このカラムには、チャートで表された正解率と正解者数が表示されます。

### 状態

このカラムには、自分がその問題を正解しているか否かが表示されます。この機能を利用するためには AOJ にログインする必要があります。既に解いた問題にはチェックマークが表示されます。

## 2-3 問題を閲覧する

各問題には制限事項などの様々な情報が付随します。この章では、問題文とその詳細情報を確認します。

問題ページはヘッダとボディから構成されています。ヘッダには問題に関する情報とメニューが含まれ、ボディに問題の本文が記載されています。

The screenshot shows the AIZU ONLINE JUDGE interface. At the top, there's a navigation bar with links for PROBLEM, RANKING, STATUS, CONTEST, and COURSE. The current problem is 'Squid Multiplication' (Volume 10 - 1069). The page includes a header with the problem title, time limit (2 sec), memory limit (32768 KB), and a link to the English version. The main content area contains the problem description in Japanese, which is a variation of a classic 'Squid' problem. It defines two sequences:  $a = \{a_0, \dots, a_n\}$  where  $a_i$  is 0 if  $i$  is even and 1 if  $i$  is odd, and  $b = \{b_i \mid i < j\}$  generated from  $a$ . The task is to find the sequence  $a$  given  $b$ . The input format is specified as  $n$  followed by  $b_0, b_1, \dots, b_{(n+1)/2-1}$ . Constraints include  $2 \leq n \leq 250$  and  $1 \leq b_i \leq 2^{63}-1$ .

ヘッダは主に以下の項目から構成されています。

### タイトル

問題のタイトルが記述されています。

### 制限

この問題を解答するために利用できる CPU の時間とメモリ量が記載されています。提出プログラムの実行時間やメモリ使用量がこれらの値を超えると不正解となります（※プログラミング言語によって緩和されます。詳しくは 2-8 章を確認してください）。







### 問題文の言語切り替えリンク

英語がサポートされている問題はまず英語の問題文が表示されます。日本語の問題を読む場合は「Japanese version is here」リンクをクリックして下さい。



## メニューリンク

問題ヘッダの右上には、以下のページ（フォーム）へのリンクボタンが表示されています。

	提出	プログラムの提出フォームを開きます。詳しくは 2-4 章を参照して下さい。
	統計	問題に対する統計情報を確認します。詳しくは 3-3 章を参照して下さい。
	掲示板	問題に関する議論が行われている掲示板のスレッドを確認します。詳しくは第 6 章を参照して下さい。
	解答	問題に正解した公開されているソースコードを観覧します。詳しくは第 4 章を参照して下さい。
	タグ	問題に関連付けられたアノテーション（カテゴリとタグ）を観覧します。
	印刷	印刷用のページを開きます（準備中です）

ボディには以下の内容が記載されています。

### 問題概要

問題の内容が記述されています。問題文に定義された仕様を満たすプログラムを作成しなければなりません。

### 入力の説明 (Input)

問題の入力に関する説明が書かれています。プログラムはここで定義されている入力形式で入力データを読み込まなければなりません。

### 出力の説明 (Output)

問題の出力に関する説明が書かれています。プログラムはここで定義されている形式で出力を行わなければなりません。余分な空白・改行を出力する、あるいは必要な空白・改行が欠如していると不正解と判定されます。

### 入力例 (Sample Input)

審判データとして与えられる入力の「例」です。入力の説明で定義された形式に従っています。

### 出力例 (Sample Output)

入力例に対する正しい出力を示します。

#### Point ★

入力例・出力例で示されるデータは、問題の入出力形式を説明するためのものであって、簡単な「例」を示します。提出されたプログラムはより厳格でサイズが大きいデータを用いて判定されることに留意して下さい。

### フッター

問題の出典や URL が記載されています。

## 2-4 プログラムを提出する

この章では、完成したプログラムを提出する方法を確認します。

問題ヘッダ右上の提出アイコンをクリックすると提出フォームが表示されます。



### Judge

Submit SolutionLast Update: 2015-10-01 14:35

問題 ID: ALDS1\_1\_A

言語: C++

ソースコード:

戻る

リセット

提出

提出フォームには以下の項目を入力します：

### 問題番号

これから提出するプログラムが解く問題の番号を入力します。問題番号は自動的に入力されます。

### 使用プログラミング言語

提出するプログラムの言語を指定します。AOJ では C,C++,Java,C++11,C#,D, Ruby, Python, Python3, Scala, Haskell, OCaml, PHP, JavaScript で書かれたプログラムを提出することができます。コンパイラの詳細は 2-8 章を参照して下さい。

## ソースコード

プログラムのソースコードを貼り付けます。プログラムを作成したエディタ等からソースコードをコピー＆ペーストして貼り付けて下さい。

以上の入力項目を埋めたら、提出フォーム下部の提出（Submit）ボタンでプログラムを提出します。リセット（Reset）ボタンでソースをクリアし、キャンセル（Cancel）ボタンでフォームを閉じることができます。

提出フォームに正しく必要事項を入力し、提出ボタンを押すと自動的に判定結果ページへ遷移します。

### Point ★

AOJ では WEB 上でのエディタを未だサポートしていません。解答プログラムは直接提出フォームに記述するのではなく、普段使用しているお好みのエディタ等でコーディングし、動作を確認した上で貼り付けましょう。

第8章では、便利なツールとして、ブラウザ上でコーディングを行うことができる、オンラインエディタを紹介します。

## 2-5 審判結果を閲覧する

AOJ は様々な種類の審判結果を返します。この章では、プログラムの判定結果の内容を確認します。

プログラムの判定結果は、最近のジャッジステータスを管理するテーブル内に表示されます。

審判結果

Last 200 submissions

Mon Apr 08 11:51:28 JST 2013

< prev | 1 / 20 | next >

per-page : 10 20 30 50 100

Run#	提出者	問題	結果	%	言語	時間	メモリ	コード長	提出日時
617746	atetubou	1504: Connect	: Wrong Answer	10/30	C++	00.04 s	12312 KB	2374 B	Mon Apr 08 11:40:50
617745	atetubou	1504: Connect	: Wrong Answer	10/30	C++	00.04 s	12312 KB	2339 B	Mon Apr 08 11:37:56
617744	cafelate911	0123: Speed Skating Badge Test	: Accepted	1/1	C	00.00 s	608 KB	432 B	Mon Apr 08 11:21:39
617743	spark6251	0078: Magic Square	: Wrong Answer	0/1	C++	00.00 s	1192 KB	821 B	Mon Apr 08 11:16:23
617742	spark6251	0078: Magic Square	: Wrong Answer	0/1	C++	00.00 s	1192 KB	818 B	Mon Apr 08 11:12:10
617741	atetubou	1507: Dungeon (II)	: Accepted	122/122	C++	00.19 s	8800 KB	1599 B	Mon Apr 08 10:55:45
617740	atetubou	1507: Dungeon (II)	: Wrong Answer	48/122	C++	00.00 s	1232 KB	1601 B	Mon Apr 08 10:54:23
617739	taninohito	0014: Integral	: Accepted	1/1	JAVA	00.05 s	17244 KB	459 B	Mon Apr 08 10:53:21
617738	taninohito	0014: Integral	: Wrong Answer	0/1	JAVA	00.05 s	17208 KB	448 B	Mon Apr 08 10:52:05
617737	taninohito	0014: Integral	: Wrong Answer	0/1	JAVA	00.05 s	17184 KB	351 B	Mon Apr 08 10:48:56

< prev | 1 / 20 | next >

ほとんどの場合、今提出したプログラムの判定結果は、テーブルの最上部に表示されます。ステータスのテーブルは以下のカラムから構成されています：

### Run#

各提出に割り当てられる固有のジャッジ ID です。審査結果や各ソースコードはこの ID によって管理されます。Run# をクリックすると、審査結果の詳細ページへ移動します。

### 提出者

提出者のユーザ ID です。自分の ID を確認しましょう。ユーザ ID をクリックするとユーザ情報のページへ移動します。

### 問題

プログラムに対応した問題番号です。問題番号をクリックすると問題ページへ移動します。

## 結果

提出プログラムの判定結果です。判定は以下の順番で決定されます：

画像	状態	意味
	In Queue	提出プログラムがキューに追加されました。ジャッジサーバに送られるまで待機中です。
	Waiting Judge	プログラムを実行中です。判定結果待ちです。しばらく待つと、判定結果のアイコンに変わります。
	Judge Not Available	一時的に判定を行うことができません。データが準備中かシステムによって制限された場合等に判定不可となります。
	Compile Error	提出されたプログラムのコンパイルに失敗しました。Compile Error のリンク先を確認して下さい。
	Runtime Error	提出されたプログラムの実行中にエラーが発生しました。不正なメモリアクセス、スタックオーバーフロー、ゼロによる割り算など多くの原因が考えられます。また、main 関数は必ず 0 を返すようにして下さい。
	Time Limit Exceeded	制限時間を超えました。不正解です。問題に指定された制限時間内にプログラムが終了しませんでした。問題の制限時間を越えてもジャッジは最大で 20 秒間プログラムを実行し続けます。20 秒以内であれば実際のプログラムの実行時間を知ることができます。
	Memory Limit Exceeded	制限メモリ使用量を超えました。不正解です。提出されたプログラムは、問題に指定された以上のメモリを使用しました。
	Output Limit Exceeded	提出されたプログラムは、制限を越えたサイズの出力を行いました。
	Wrong Answer	不正解です。提出されたプログラムは審判データと異なる出力データを生成しました。特別なジャッジ (special judge) については、問題に指定された精度等を考慮します。
	WA:Presentation Error	出力の形式が誤っています。提出されたプログラムは、正しい計算結果を出力していますが、余計な空白や改行を行っていたり、あるいは必要な空白や改行を出力していません。なお、これは誤差を許す special judge (緑のシンボル) 及び検証器を用いる (黄色のシンボル) 問題には適用されません。
	Partial Points	上記不正解の状態において、部分点が得られている場合に付加されます。部分点は主にトピックの問題に対する解答に適用されます。
	Accepted	正解です。提出されたプログラムは上記すべての審査において拒否されなかったため、" 受理 " されました。審査には提出プログラムの出力を判定するために厳格なデータを用いていますが、提出されたプログラムが完全に正しいことを保証するものではありません。

## 言語

提出されたプログラムの言語です。

## 時間

提出プログラムが該当問題の審判データに対する結果を出力するのに要した秒単位の時間です。AOJでは最大20秒(※言語によって緩和されます)プログラムを走らせます。従って、10秒を超える表示については測定不能となります。複数の審判データ(テストケース)がある場合は、その結果の中で最大のものが表示されます。

## メモリ

提出プログラムが該当問題の審判データに対する結果を出力するのに要したキロバイト単位のメモリ使用量です。複数の審判データ(テストケース)がある場合は、その結果の中で最大のものが表示されます。

## コード長

提出プログラムのバイト単位の文字数です。

## 提出日時

プログラムが提出された日時です。

### Point ★

AOJでは審判結果にかかわらず何度でも再提出を行うことができます。正解するまで、あるいは納得のいく性能が得られるまで、何度でもチャレンジしましょう。

2-6 審判結果の詳細を確認する

審判結果の詳細をレビューページで確認することができます。

ジャッジステータスのRun#をクリックすると、レビューページに移動します。

#617900Solution for 10008: A / B Problem by zukky

ソースコード

ステータス

テストケース

アクセス: public 閲覧数: 7 コメント: 0

Compile Error Logs:

Runtime Error Logs:

ステータス

Judge: 8/10

Python

CPU: 00.02 sec

Memory: 4212 [KB]

Length: 64 [B]

2013-04-08 15:58

2013-04-08 15:58

テストケースの判定結果

Case #1:	✓: Accepted	00.01 sec	4212 [KB]	in1.txt	6 B	out1.txt	16 B
Case #2:	✓: Accepted	00.01 sec	4212 [KB]	in2.txt	4 B	out2.txt	15 B
Case #3:	✓: Accepted	00.01 sec	4208 [KB]	in3.txt	11 B	out3.txt	18 B
Case #4:	✓: Accepted	00.01 sec	4212 [KB]	in4.txt	4 B	out4.txt	15 B
Case #5:	✓: Accepted	00.01 sec	4212 [KB]	in5.txt	6 B	out5.txt	19 B
Case #6:	✓: Accepted	00.02 sec	4212 [KB]	in6.txt	9 B	out6.txt	20 B
Case #7:	✓: Accepted	00.01 sec	4208 [KB]	in7.txt	13 B	out7.txt	33 B
Case #8:	✓: Accepted	00.01 sec	4212 [KB]	in8.txt	12 B	out8.txt	29 B
Case #9:	✗: Wrong Answer	00.01 sec	4212 [KB]	in9.txt	12 B	out9.txt	15 B
Case #10:	-	-	-	-	NA	-	NA

ジャッジからのメッセージ

コンパイルエラーメッセージとランタイムエラーメッセージを閲覧することができます。

ステータス

ステータスの概要です。テストケースの正解率、言語、最大 CPU 時間、最大メモリ使用量、コードサイズ、日時が含まれます。

テストケースの判定結果

各テストケース(ジャッジデータ)ごとの判定結果の詳細がリストアップされています。テストケースごとのCPU時間、メモリ使用量を確認することができます。また、審判データに使用された入出力データを閲覧することができます(※多くのデータが準備中です)。

## 2-7 テストケースを確認する

審判データとして使用されたテストケースを確認することができます。

審判結果詳細の入出力ファイルへのリンク、または「詳細」タブをクリックすると、ジャッジデータテストケースを確認するページへ移ります。

ページ左側にジャッジで使用された各ケースの入力データ、右側に対応するケースの出力ジャッジデータ（正しい出力結果）が表示されます。※提出したプログラムの出力ではないことに注意して下さい。

The screenshot shows a web interface for a programming competition solution. At the top, it says "#665697 Solution for 10010: Simple Calculator by admin1". Below this are tabs for "ソースコード", "ステータス", and "テストケース", with "テストケース" being the active tab. To the right of the tabs, it says "アクセス: private", "閲覧数: 0", and "コメント: 0". Below the tabs, there is a status bar showing "<| 3 / 5 |>", a red flag icon with "Wrong Answer", "00.00 sec", and "1028 [KB]". The main content area is divided into two columns: "入力 #3 ( in3.txt )" on the left and "出力 #3 ( out3.txt )" on the right. The input column contains 20 lines of arithmetic expressions, and the output column contains 20 lines of results. The last line of input is "100 ? 100" and the last line of output is "16".

入力 #3 ( in3.txt )	出力 #3 ( out3.txt )
20000 + 20000	40000
20000 / 20000	1
20000 * 20000	400000000
20000 - 20000	0
0 + 20000	20000
0 / 20000	0
0 * 20000	0
0 - 20000	-20000
10009 * 997	9978973
10009 - 997	9012
10009 + 997	11006
10009 / 997	10
8 * 8	64
8 / 8	1
8 - 8	0
8 + 8	16
100 ? 100	

矢印ボタン（リンク）でテストケースを変更することができます。

容量が大きいケースについては表示されません。



## 2-8 コンパイラとシステムの詳細を確認する

自分の環境と AOJ の環境は異なるため、その詳細を把握しておく必要があります。

現在 AOJ では以下の言語をサポートしています。下記の表に示すように、言語によって各問題に設定された CPU の制限時間とメモリ使用量が緩和されます。

言語	コンパイラ	時間緩和	メモリ緩和	マシン
C	gcc 4.6.3 20120306 (Red Hat 4.6.3-2) (GCC)	1.0	1.0	#4
C++	gcc 4.6.3 20120306 (Red Hat 4.6.3-2) (GCC)	1.0	1.0	#4
JAVA	opneJDK version 1.8.0_45	2.0	5.0	#7
C++11	gcc 4.6.3 20120306 (Red Hat 4.6.3-2) (GCC)	1.0	1.0	#5
C#	C# mono 2.10.8	2.0	2.0	#7
D	DMD64 D Compiler v2.067	1.0	1.0	#7
Ruby	Ruby 2.2.2	5.0	5.0	#6
Python	Python 2.7.10	5.0	5.0	#6
Python3	Python 3.4.2	5.0	5.0	#6
Scala	Scala 2.11.6	2.0	5.0	#7
Haskell	GHC 7.8.4	2.0	2.0	#7
OCaml	OCaml 4.02.0	2.0	2.0	#7
PHP	PHP 5.6.10	5.0	5.0	#6
JavaScript	node.js 0.10.36	5.0	5.0	#6

※パラメタは変更される可能性があります。予めご了承ください。

C/C++ については緩和はされず、問題文に記載されている制限事項が適用されます。一方、例えば Java の場合は問題文に記載されている CPU 制限値は 2 倍、メモリ制限値は 5 倍の制限がそれぞれ適用されます。



# ユーザステータス

User status

問題の正解状況やタグ等から、様々な統計情報が記録され、ユーザのステータスやランキングの決定に用いられています。この章では、ユーザのステータスやランキングの詳細について確認します。

※一部の機能は現在開発中です。

**Chapter 2-1** ユーザステータスを閲覧する

**Chapter 2-2** ランキングを閲覧する

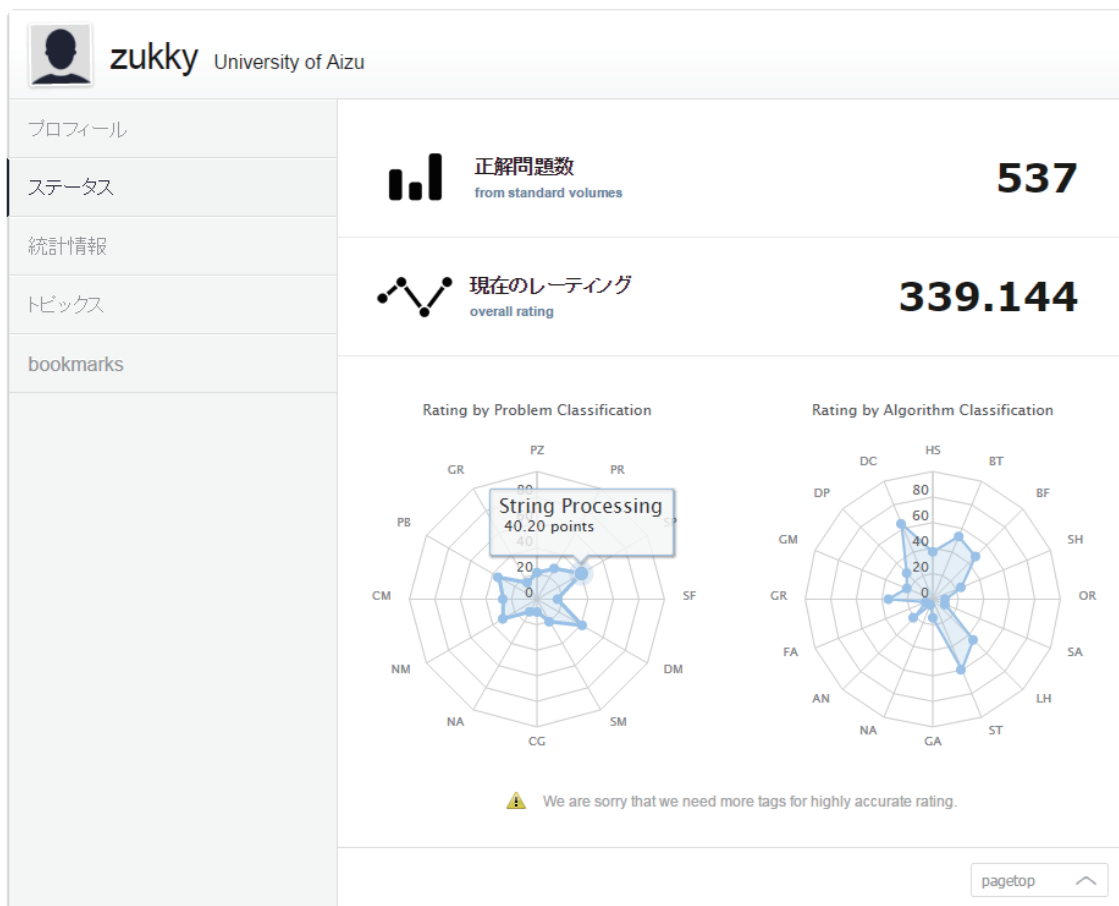
**Chapter 2-3** 問題ごとのランキングを閲覧する

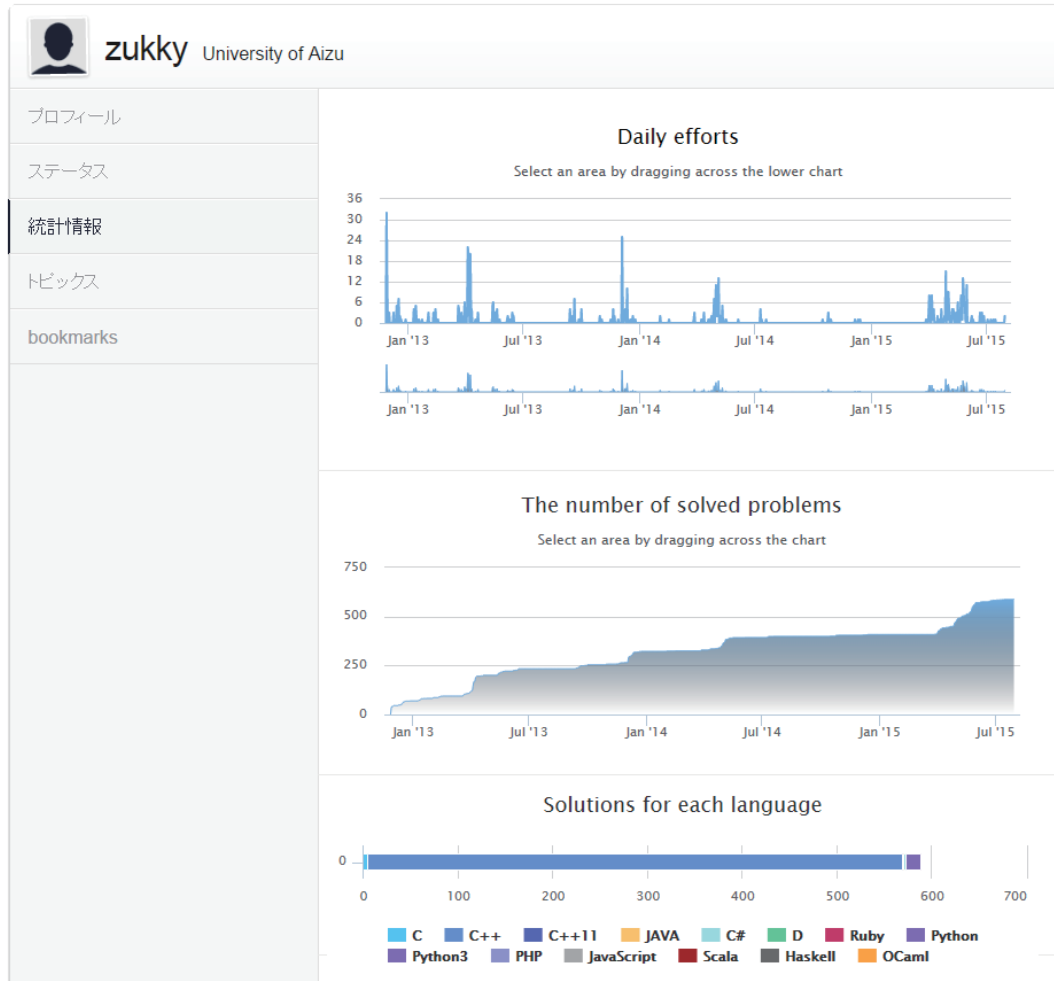
### 3-1 ユーザステータスを閲覧する

自分や他のユーザのステータスを閲覧することができます。

ページヘッダ右上の「My Status」をクリックすると自分のステータス（成績）を閲覧することができます。ランキングページの「Link」や Status ページのユーザ ID リンクからも、各ユーザのステータスを閲覧することができます。

このページでは、各分野のポイント（各問題に分野ごとのポイントが割り当てられています）や、提出状況・統計、正解した問題リストを確認することができます。





## 3-2 ランキングを閲覧する

AOJ では問題数やレーティングで他のユーザと競争することができます。

ヘッダメニューの「ランキング」リンクより、AOJ ユーザのランキングを閲覧することができます。ランキングにマウスを合わせると以下に示す基準ごとのランキングへのリンクが表示され、それぞれ閲覧することができます。

ユーザランキング 1 - 100: 解いた問題数順 Select section:

Solved Rating Diff

< prev | 1 / 5 | next > per page: 10 20 30 50 100

順位	ID	名前	所属	問題数	レート	差分	平均	提出	
1	orpheus	rook	Kihoku Technical High School	1018	2176.69	72.27	2.14	2433	
2	K_Operafan	Chihaya-chan Love!!	841 Production	979	2255.91	-19.74	2.30	3790	
3	fura2	@fura_2	Osaka University	945	1811.18	-12.92	1.92	2214	
4	mamekin	mamekin	-	905	1385.71	19.02	1.53	1768	
5	sune2	@sune232002	Kyoto University	889	1264.68	-10.94	1.42	1718	
6	ishikado	Ziro Saburo Motonobu	Country of Mikawa	812	817.04	4.85	1.01	3281	
7	ichyo	ichyo	Japan	780	971.35	96.81	1.25	1895	
8	s1150092	retired	retired	772	1265.86	-5.40	1.64	4816	
9	ee67052	nanikaka	Meiji University	761	762.12	-4.97	1.00	1855	
10	Respect2D	Yayoi-chan Love!!	765 Production	730	609.67	4.18	0.84	1638	
11	yukim	Sharo-chan Love!!	Holmes Detective Academy	729	892.17	45.32	1.22	1751	
12	shiohiota	Tetsuya Shiota	The University of Tokyo (OB Aizu)	680	578.13	-0.10	0.85	2426	
13	lyrically	hos.lyric	The University of Tokyo	667	1568.62	-4.81	2.35	1380	
14	Darsein	Azu-nyan Love!!	Wakaba Girls	663	488.17	21.15	0.74	2042	
15	kyuridenamida	kyuridenamida	Osaka Prefecture University Co ...	659	580.15	-13.77	0.88	2568	
16	ustimaw	Kawamitsu Naohiro	Osaka University	629	375.44	-1.91	0.60	1154	
17	s1180132	WinField95	University of Aizu	619	517.97	74.53	0.84	2168	
18	sinapusu2002	sin-horie	job hunting	595	433.82	9.50	0.73	1577	
19	s1180133	purple_jewel928	University of Aizu	592	344.07	18.47	0.58	1478	
20	rankalee	Seiya Takahara	The University of Tokyo	588	463.46	20.59	0.79	1455	

< prev | 1 / 5 | next >

### 問題数順

AOJ に掲載されている問題の正解問題数を基準にしたランキングです。

### レーティング順

各問題に正解すると、その問題の正解者数から算出されたポイントを得ることができます。ポイントの合計がユーザのレーティングとなります。レーティングを基準にしたランキングでは、より難易度の高い問題を多く解いているユーザが上位となります。

### 差分順

レーティングの変動がプラスに大きい順にランキングされます。

※現在、レーティングの更新は不定期に行われています。

### 3-3 問題ごとのランキングを閲覧する

各問題ごとに、CPU 使用時間やコードの長さで競うことができます。



問題ページヘッダの「統計」アイコンをクリックすると、問題の統計情報ページに移動します。ここでは、各問題の統計情報を確認することができます。また、CPU 使用時間やコード長によるランキングを閲覧することができます。

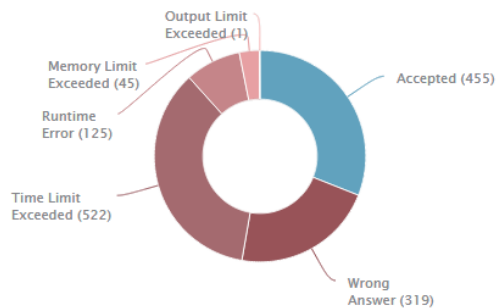
#### Status of 0529 : Darts

Summary

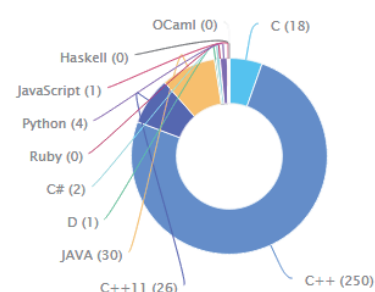
Ranking (Top 325 Users)

Users : 325 Accepted rate : 31 %

##### Users submissions (Total: 1467)

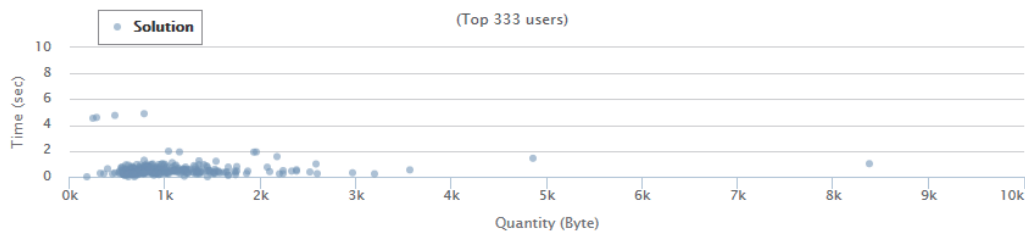


##### Submissions for each language

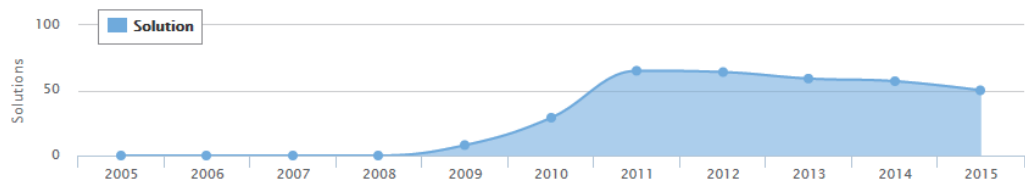


##### CPU Time versus Code Quantity

(Top 333 users)



##### Yearly Statistics







# 4 ソースコードを閲覧・管理する

Managing source code

4

ソースコードを閲覧・管理する

他のユーザが作成したソースコードを閲覧して学習に役立てることができます。また、自分のソースコードを公開することができます。

- Chapter 4-1** 公開されているソースコードを閲覧する
- Chapter 4-2** ソースコードのレビュー
- Chapter 4-3** ソースコードを管理する



4-1 公開されているソースコードを閲覧する

公開されている他のユーザの解答プログラムは言語ごとにリストアップされます。

🔍 ヘッダメニューの虫眼鏡アイコンをクリックすると、公開されたソースコード一覧のページへ移動します。  
テーブル上部のリンクより、言語を選択することができます。

Public Solutions for 0587

C (3) C++ (17) JAVA (2) C++11 (1) C# (0) D (0) Ruby (0) Python (1) PHP (0) JavaScript (0)

Solutions in C++

#	Author	Submission Date	Language	Version	Server	Time	Code	Reviewed
584524	s1190170	2013-02-24 18:01	C++	4.6.3	#4	00.00 sec	308 byte	7
586509	taichan	2013-02-27 15:53	C++	4.6.3	#4	00.00 sec	457 byte	5
587112	kachan	2013-02-28 11:57	C++	4.6.3	#4	00.00 sec	460 byte	2
587416	ichyo	2013-02-28 17:39	C++	4.6.3	#4	00.00 sec	1143 byte	9
591789	kusohako	2013-03-05 23:45	C++	4.6.3	#4	00.00 sec	485 byte	0
594583	daiota	2013-03-09 18:04	C++	4.6.3	#4	00.00 sec	653 byte	3
615218	ltzz	2013-04-04 16:15	C++	4.6.3	#4	00.00 sec	232 byte	1
582589	nox	2013-02-21 13:11	C++	4.6.3	#4	00.00 sec	346 byte	8
580444	kyuridenamida	2013-02-17 23:06	C++	4.6.3	#4	00.00 sec	2054 byte	9
580705	mamekin	2013-02-18 14:00	C++	4.6.3	#4	00.00 sec	655 byte	6
581398	id	2013-02-19 15:04	C++	4.6.3	#4	00.00 sec	446 byte	4
581977	sh19910711	2013-02-20 13:44	C++	4.6.3	#4	00.00 sec	1662 byte	4
582044	ishikado	2013-02-20 15:15	C++	4.6.3	#4	00.00 sec	2039 byte	5
582134	s1180132	2013-02-20 17:31	C++	4.6.3	#4	00.00 sec	398 byte	3
582188	otaks21	2013-02-20 18:39	C++	4.6.3	#4	00.00 sec	294 byte	6
582588	Respect2D	2013-02-21 13:10	C++	4.6.3	#4	00.00 sec	385 byte	8
617667	ichigo_o_re	2013-04-08 03:46	C++	4.6.3	#4	00.00 sec	637 byte	0

## 4-2 ソースコードのレビュー

各問題について、公開されている他のユーザの解答プログラムを閲覧することができます。

リストの左列の#（ジャッジ ID）リンクをクリックすると、ソースコードを閲覧することができます。

#582134 Solution for 0587: Tunnel by s1180132

ソースコード ステータス 詳細 コメント



アクセス: public 閲覧数: 3 コメント: 0

00.00 sec 1160 KB 30 lines 398 bytes 2013-02-20 17:31

```
1 #include<iostream>
2 #include<algorithm>
3 using namespace std;
4
5 int N,M;
6
7 void solve(){
8     cin >> N >> M;
9     int s = M;
10    int res = 0;
11    for(int i = 0 ; i < N ; i++){
12        int in,out;
13        cin >> in >> out;
14        res = max(res,s);
15        s += in;
16        s -= out;
17        if(s < 0){
18            cout << 0 << endl;
19            return;
20        }
21    }
22    res = max(res,s);
23    cout << res << endl;
24 }
25
26 int main(){
27     solve();
28     return 0;
29 }
```

## 4-3 ソースコードを管理する

これまでにチャレンジした問題のリストから、提出履歴やソースコードを確認することができます。プロフィールまたは統計情報ページの Repository リンクから、以下のような管理ページへ移動することができます。

Repository		
Category	Problems	Status
 Volumes	Submissions to Problems in Volumes >	
 Library	Submissions to Problems in Libraries >	


Volume または Library をクリックすると、チャレンジした問題を閲覧することができます。

Repository

CategoryProblemsStatus

< prev | 4 / 4 | next >


<

 GRL\_2\_A

Minimum Spanning Tree

submissions >


<

 GRL\_3\_A

Articulation Points

submissions >


<

 GRL\_4\_B

Topological Sort

submissions >


<

 GRL\_5\_A

Diameter of a Tree

submissions >


<

 GRL\_5\_B

Height of a Tree

submissions >


<

 GRL\_5\_C

Lowest Common Ancestor

submissions >


<

 GRL\_6\_A

Maximum Flow

submissions >


<

 GRL\_7\_A

Bipartite Matching

submissions >


<

 ITP1\_1\_B

X Cubic

submissions >


<

 ITP1\_1\_D

Watch

submissions >


<

 NTL\_1\_A

Prime Factorize

submissions >





















<

 NTL\_1\_E

Extended Euclid Algorithm

submissions >

右側のリンクから、その問題に対する提出履歴を確認することができます。

Repository									
Category	Problems	Status							
<	 792525	1321	 : Memory Limit Exceeded	0/1	C++	1653 sec	820164 B	2743 B	Sun Oct 20 18:05:39 JST 2013 >
<	 792526	1321	 : Memory Limit Exceeded	0/1	C++	1649 sec	820164 B	2744 B	Sun Oct 20 18:07:03 JST 2013 >
<	 792533	1321	 : Compile Error	0/1	C++	0 sec	0 B	2893 B	Sun Oct 20 18:20:22 JST 2013 >
<	 792534	1321	 : Memory Limit Exceeded	0/1	C++	693 sec	326736 B	2895 B	Sun Oct 20 18:20:54 JST 2013 >
<	 792541	1321	 : Memory Limit Exceeded	0/1	C++	162 sec	74712 B	3149 B	Sun Oct 20 18:34:56 JST 2013 >
<	 792548	1321	 : Memory Limit Exceeded	0/1	C++	185 sec	92936 B	3387 B	Sun Oct 20 18:57:41 JST 2013 >
<	 792550	1321	 : Memory Limit Exceeded	0/1	C++	158 sec	74712 B	3385 B	Sun Oct 20 18:58:08 JST 2013 >
<	 792551	1321	 : Memory Limit Exceeded	0/1	C++	193 sec	92932 B	3315 B	Sun Oct 20 19:00:56 JST 2013 >
<	 792554	1321	 : Accepted	1/1	C++	29 sec	11996 B	3394 B	Sun Oct 20 19:05:16 JST 2013 >
<	 792556	1321	 : Accepted	1/1	C++	29 sec	11988 B	3278 B	Sun Oct 20 19:08:48 JST 2013 >

Judge ID または右側のリンクから、ソースコードを確認することができます。

※「管理」機能については未だ開発中です。



# 5 コース

Explanation of course

コースでは、関連のある問題セットをステップ b y ステップで解くことができます。プログラミングを始めたばかりの人向けの初級問題から、基本的なアルゴリズムや特定領域の問題等をテーマにしたコースが準備されています。

**Chapter 5-1** コースの選択

**Chapter 5-2** トピックの進め方

## 5-1 コースの選択

レベルや目的に応じたコースが準備されています。

ヘッダメニューからコースリンクをクリックするとコースリストのページへ移ります。コース名からコースを選びます。



タブによってコースのタイプ（レッスン、ライブラリ等）を絞り込むことができます。

コースリスト			
<div>すべて レッスン ライブラリ</div>			
<div>&lt; prev   1 / 1   next &gt;</div> <div>per-page : 10 20 30 50 100</div>			
ID	コース名	タイプ	ステータス
ITP1	Introduction to Programming I	lesson	Available
ALDS1	Algorithms and Data Structures I	lesson	Available
DSL	Data Structures	library	now in preparation
CGL	Computational Geometry	library	now in preparation
GRL	Graph Algorithms	library	now in preparation
DPL	Dynamic Programming	library	now in preparation
NTL	Number Theory	library	now in preparation
<div>&lt; prev   1 / 1   next &gt;</div>			

## 5-2 トピックの進め方

この章では、コースを構成するトピックの進め方を説明します。

1つのコースは段階を踏んでチャレンジすることができる複数のトピックから構成されています。コースのページでは、トピック、問題リスト、提出フォーム、ステータス確認を切り替えるタブで構成されています。

各メニューへはタブで移動することもできますが、基本的にはページ内のリンクやボタンで移動します。

ページの最初はコースに含まれるトピックのリストが表示されます。トピック番号、トピック名、達成度、獲得得点が表示されています。リストの下に、トピック全体の進捗率と合計得点が表示されます。

Introduction to Programming I - -						
トピック   問題   提出   ステータス						
#	トピック		達成度		得点	
1	Getting Started	Result	100 %	<div><div></div></div>	300 / 300	>
2	Branch on Condition	Result	66 %	<div><div></div></div>	200 / 300	>
3	Repetitive Processing	Result	0 %	<div><div></div></div>	0 / 300	>
4	Computation	Result	40 %	<div><div></div></div>	120 / 300	>
5	Nested Controls I	Result	0 %	<div><div></div></div>	0 / 300	>
6	Array	Result	0 %	<div><div></div></div>	0 / 300	>
7	Nested Controls II	Result	0 %	<div><div></div></div>	0 / 300	>
8	Character	Result	0 %	<div><div></div></div>	0 / 300	>
9	String	Result	0 %	<div><div></div></div>	0 / 300	>
10	Math	Result	0 %	<div><div></div></div>	0 / 300	>
合計得点			17 %	<div><div></div></div>	620 / 3600 点	

トピック名または右端のボタンをクリックすると各トピックのページへ移動します。

1つのトピックはアルファベットのAから始まる複数の問題から構成されています。各問題の行では、問題ID、問題名、制限実行時間、制限メモリ、正解者数、獲得得点が表示されています。

左端のリンクからトピック一覧に戻ることができます。

Introduction to Programming I - Computation -							
トピック   問題   提出   ステータス							
ID	タイトル		制限時間	制限メモリ	正解者数	得点	
<  A	A / B Problem		 1 sec	64 MB	180	0 / 100	
<  B	Circle		 1 sec	64 MB	174	100 / 100	
<  C	Simple Calculator		 1 sec	64 MB	166	20 / 100	

問題名または右端のボタンをクリックすると、問題文のページへ移動します。問題文の閲覧については 2-3 章を参考にしてください。

AIZU  
ONLINE  
JUDGE

PROBLEM

RANKING

STATUS

CONTEST

COURSE

Logout

Register/Setting

HOME - ALDS1 - ALDS1\_2 - B

For beginnersMy Status

Sort I - Selection Sort

Time Limit : 1 sec, Memory Limit : 65536 KB

English version is here

?

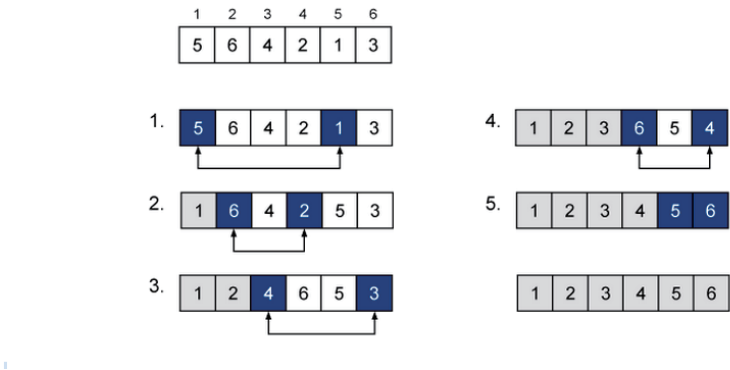
Q

📄

選択ソート

数列Aを読み込み、選択ソートのアルゴリズムで昇順に並び替え出力するプログラムを作成せよ。以下に示す疑似コードに従いアルゴリズムを実装せよ。

```
SelectionSort(A)
1 for i = 1 to length[A]
2   do mini = i
3   do for j = i to length[A]
4     do if A[j] < A[mini]
5       then mini = j
6   swap A[i] and A[mini]
```



問題文ページ右上の提出アイコンまたは問題リスト右端の紙飛行機アイコンをクリックすると、提出フォームへ移動します。フォームで、プログラミング言語を選択し、解答プログラムをソースコード領域に貼り付けて（コピー＆ペースト）下さい。

The screenshot shows a web interface for a programming problem. At the top, the title is "Introduction to Programming I - Computation - Circle". Below the title is a navigation bar with tabs: "トピック", "問題", "提出", and "ステータス". The "提出" (Submit) tab is active. Below the navigation bar is a button labeled "問題リストへ戻る" (Return to problem list). The main form area has a "タイトル:" (Title) field with the value "B: Circle" and a "言語:" (Language) dropdown menu set to "C++". Below these is a large text area for the "ソースコード:" (Source code) with the following C++ code:

```
#include<iostream>
#include<cstdio>
using namespace std;

const double PI = 3.14159265358979;

int main(){
    double n;
    cin >> n;
    printf("%.6f %.6f\n",PI*n*PI,PI*n*PI);
    return 0;
}
```

At the bottom of the form are three buttons: "戻る" (Return), "リセット" (Reset), and "提出" (Submit).

提出ボタンをクリックすると、ステータスページへ移動します。判定結果の詳細については、2-5 章「判定結果を閲覧する」を参考にして下さい。





## 掲示板 (Web borad)

Web borad

問題やシステムについて質問がある場合は、掲示板を活用しましょう。

**Chapter 6-1** メッセージを閲覧・投稿する

**Chapter 6-2** スレッドを投稿する

**Chapter 6-3** メッセージ・スレッドを削除する

## 6-1 メッセージを閲覧する

問題に対しての他のユーザのメッセージを閲覧することができます。

問題ページの右上にある Discussion アイコンを選択すると、その問題専用のスレッドリストを閲覧することができます。



## 3 threads for problem 0022

全てのスレッド 質問 議論 トラブル お知らせ

5

の回答

## ② Problem:0022 Main Thread

My program success about sample input. But judge system return "Time Limit Exceeded". Can you give me any advice?  
Thank you.

Posted by [aoibbs](#) LastPosted: 2012-04-18 13:32:37.0

1

の回答

## ② 連続した項の和とは

項が1個のみでも連続した項の和と考えるのでしょうか？たとえば、 $n=4$ で  $a1=-1$   $a2=-1$   $a3=-1$   $a4=-1$  の場合の答えはどうでしょう？連続した項の和というときは最低でも2項の連続で答えは  $-2$  となるように思いますが、判定データは答えは  $-1$  としているのでしょうか？

Posted by [trsq](#) LastPosted: 2012-08-08 23:43:04.0

1

の回答

## ② 問題分の誤記「Maximum Sum Sequece」

問題文冒頭の小見出しに誤記があるように思います。誤: Maximum Sum Sequece 正: Maximum Sum Sequence、ご確認願います。

Posted by [nogitsune413](#) LastPosted: 2013-04-17 12:55:57.0

スレッドを作成する

見たいスレッドのタイトルを選択します。

スレッドを選択するとメッセージのリストを閲覧することができます。

Thread ID: 23

Problem ID: 0022



My program success about sample input.  
But judge system return "Time Limit Exceeded".  
Can you give me any advice?

Thank you.

Posted by [AIZU](#) Last Posted: 2010-09-13 20:54:22.0: Problem [0022](#)

5 の回答



How about thinking a sum of a contiguous subsequence to be one number?  
And note that Maximum Sum Sequence doesn't include the first number when it's a negative number.

Posted by [burdock](#) Last Posted: 2010-09-14 06:04:17.0: Problem [0022](#)



Thank you.  
I solve it. :-)

Posted by [AIZU](#) Last Posted: 2010-09-14 19:57:25.0: Problem [0022](#)



和の最大値、最小値がどれくらいの範囲になるのかの制約を書いておいた方が良いのでは？

Posted by [snukenin](#) Last Posted: 2010-10-11 23:10:46.0: Problem [0022](#)

スレッドの最下部に、メッセージを入力するフォームがあります。スレッドに対してメッセージを投稿するためには、ログインする必要があります。フォームに入力したメッセージを「コメントを投稿する」ボタンで投稿することができます。



迅速な対応ありがとうございます！

Posted by [snukenin](#) Last Posted: 2010-10-12 18:32:57.0: Problem [0022](#)

メッセージ

コメントを投稿する

メッセージの投稿に成功すると、そのメッセージが一覧に追加されます。

## 6-2 スレッドを投稿する

問題に対してスレッドを投稿することができます。

スレッドリストの最下部にある「スレッドを作成する」ボタンを押すと、新しいスレッドを作成することができます。スレッドを作成するには、ログインが必要です。

The screenshot shows a web form for creating a new thread. At the top, there is a button labeled 'スレッドを作成する' (Create Thread). Below it, there is a text input field for the 'タイトル' (Title). Under the title field, there are four radio buttons for selecting a category: '質問' (Question), '議論' (Discussion), 'トラブル' (Trouble), and 'お知らせ' (Announcement). The '質問' option is selected. Below the category selection, there is a large text area for the 'メッセージ' (Message). At the bottom of the form, there is a '送信' (Send) button.

以下の情報を入力・選択して、送信ボタンを押すと、新しいスレッドが作成されます。

### タイトル

スレッドのタイトルを入力します。内容を要約する適切なタイトルを付けましょう。

### カテゴリ

以下の中から、議論に関連のあるカテゴリを選択します。

Question: 問題の解法等に関する質問

Discussion: 問題や解法に関する議論

Trouble: 問題発生への報告

Announce: 管理者からのメッセージなど

### メッセージ

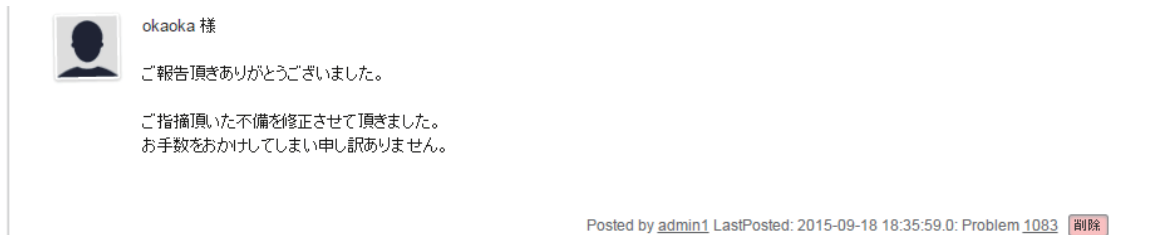
スレッドの最初の投稿メッセージになります。

スレッドの投稿に成功すると、投稿したスレッドが一覧に表示されます。

## 6-3 メッセージ・スレッドを削除する

自分で作成したメッセージ・スレッドは削除することができます。

自分で作成したメッセージとスレッドの右下には赤い Delete ボタンが表示されます。



削除したいメッセージ・スレッドがあれば Delete ボタンを選択し、メッセージ・スレッドを削除します。



# コンテスト

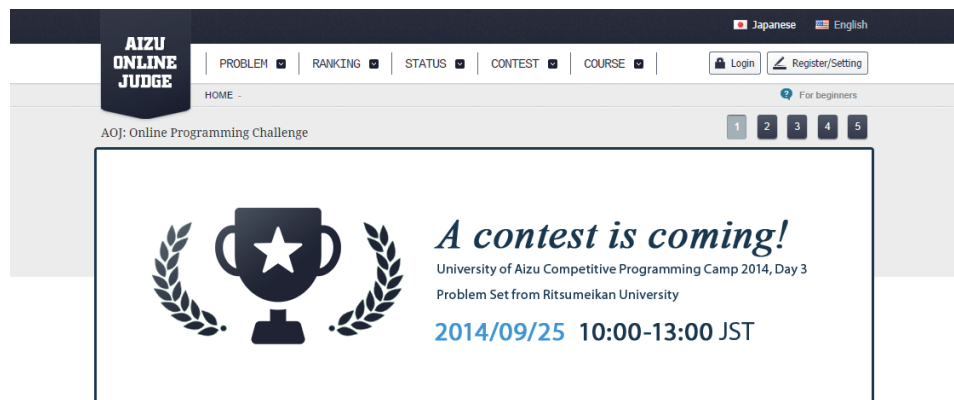
Contest

AOJ では非定期的にコンテストが開催されています。特に制限がなければどなたでも参加することができます。

- [Chapter 7-1](#) コンテストに参加する
- [Chapter 7-2](#) 問題を解く
- [Chapter 7-3](#) ランキングを閲覧する
- [Chapter 7-4](#) 質問する
- [Chapter 7-5](#) 統計情報を閲覧する

## 7-1 コンテストに参加する

トップページのコンテスト案内、またはヘッダの「コンテスト」リンクからコンテストリストのページへ移動します。



開催予定、開催済みのコンテストリストがタイトルや日時とともに表示されます。タイトルから各コンテストのページへ移動することができます。

### Contest List

Public Private

< prev | 1 / 3 | next >

per-page : 10 20 30 50 100

Contest ID	Title	Start	End	Policy	State	Info.
ACPC2014Day3	Aizu Competitive Programming Camp 2014 Day 3	14/09/25 10:00	14/09/25 13:00	public	END	<a href="#">i</a>
ACPC2014Day2	Aizu Competitive Programming Camp 2014 Day 2	14/09/24 11:00	14/09/24 16:00	public	END	<a href="#">i</a>
ACPC2014Day1	Aizu Competitive Programming Camp 2014 Day 1	14/09/23 14:30	14/09/23 17:30	public	END	<a href="#">i</a>
RitsCamp14Day3	Ritsumeikan Competitive Programming Camp 2014 Day 3	14/03/19 10:00	14/03/19 13:00	public	END	<a href="#">i</a>
RitsCamp14Day2	Ritsumeikan Competitive Programming Camp 2014 Day 2	14/03/18 11:00	14/03/18 16:00	public	END	<a href="#">i</a>
RitsCamp14Day1	Ritsumeikan Competitive Programming Camp 2014 Day 1	14/03/17 14:45	14/03/17 17:45	public	END	<a href="#">i</a>
ACAC007	Archived Classic Algorithm Contest 007	13/12/20 20:00	13/12/20 22:00	public	END	<a href="#">i</a>
ACAC006	Archived Classic Algorithm Contest 006	13/12/18 20:00	13/12/18 22:00	public	END	<a href="#">i</a>
ACAC005	Archived Classic Algorithm Contest 005	13/12/16 20:00	13/12/16 22:00	public	END	<a href="#">i</a>
ACAC004	Archived Classic Algorithm Contest 004	13/11/19 20:00	13/11/19 22:00	public	END	<a href="#">i</a>
ACAC003	Archived Classic Algorithm Contest 003	13/11/03 13:30	13/11/03 15:30	public	END	<a href="#">i</a>
ACAC002	Archived Classic Algorithm Contest 002	13/11/02 15:00	13/11/02 17:00	public	END	<a href="#">i</a>
ACAC001	Archived Classic Algorithm Contest 001	13/10/29 20:00	13/10/29 22:00	public	END	<a href="#">i</a>
ACAC000	Archived Classic Algorithm Contest 000	13/10/26 13:30	13/10/26 15:30	public	END	<a href="#">i</a>
ACPC2013Day3	Aizu Competitive Programming Camp 2013 Day 3	13/09/05 10:00	13/09/05 15:00	public	END	<a href="#">i</a>
ACPC2013Day2	Aizu Competitive Programming Camp 2013 Day 2	13/09/04 11:00	13/09/04 16:00	public	END	<a href="#">i</a>
ACPC2013Day1	Aizu Competitive Programming Camp 2013 Day 1	13/09/03 14:30	13/09/03 17:30	public	END	<a href="#">i</a>
RitsCamp13Day3	Ritsumeikan Competitive Programming Camp 2013 Day 3	13/03/13 10:00	13/03/13 13:00	public	END	<a href="#">i</a>
RitsCamp13Day2	Ritsumeikan Competitive Programming Camp 2013 Day 2	13/03/12 11:00	13/03/12 16:00	public	END	<a href="#">i</a>
RitsCamp13Day1	Ritsumeikan Competitive Programming Camp 2013 Day 1	13/03/11 14:30	13/03/11 17:30	public	END	<a href="#">i</a>

< prev | 1 / 3 | next >

各コンテストは、下の図のヘッダに示されるように、いくつかのメニューから構成されています。

ACPC2013Day3 UEC Programming Contest 2013							
Problem	Standing	Judge	Clarification	Gallery	Analysis	My Submissions	Rejudge
Judge	Doc	Problem Title	TL	ML	Success	State	
		A: Register Phase	2 sec	65536 KB	<div><div></div></div> 86.49 %  x 32	challenge	
		B: Time Complexity	2 sec	65536 KB	<div><div></div></div> 57.78 %  x 26	challenge	
		C: Mysterious Operator	2 sec	65536 KB	<div><div></div></div> 88.00 %  x 22	challenge	
		D: Change	2 sec	65536 KB	<div><div></div></div> 36.17 %  x 17	challenge	
		E: Pie Chart is as easy as pie.	2 sec	65536 KB	<div><div></div></div> 71.43 %  x 10	challenge	
		F: MLE	2 sec	65536 KB	<div><div></div></div> 46.15 %  x 6	challenge	
		G: Get Lost	8 sec	262144 KB	<div><div></div></div> 57.14 %  x 4	challenge	
		H: Brainf*ck	2 sec	65536 KB	<div><div></div></div> 19.05 %  x 4	challenge	
		I: Reverse Game	1 sec	65536 KB	<div><div></div></div> 33.33 %  x 1	challenge	
		J: Avant-garde Art	2 sec	65536 KB	<div><div></div></div> 30.43 %  x 7	challenge	
		K: Collector	2 sec	65536 KB	<div><div></div></div> 18.18 %  x 2	challenge	
		L: The Return of FizzBuzz	8 sec	65536 KB	<div><div></div></div> 12.50 %  x 1	challenge	

Problem(問題) リンクから問題リストを閲覧することができます。問題リストには、通常問題のファインダーと同様に、判定方法、タイトル、制限時間・メモリ、正解率等が表示されます。タイトルまたは Challenge リンクから問題文のページへ移動します。



問題文は、通常問題と同じような構成となっています。問題文ページ右上の提出アイコンから提出フォームを開き、解答プログラムを貼り付け、Submit（提出）ボタンを押します。

ACPC2015Day3

問題文
順位
**審判**
質問
ギャラリー
Analysis
My Submissions
Rejudge

問題 ID: A

言語: C++

ソースコード:

戻る
リセット
提出

判定結果が表示されるステータスページ（Judge）に遷移するので、結果を確認してください。また、自分が提出したプログラムの判定結果リストは My Submission から確認することができます。

ACAC005							
Problem	Standing	Judge	Clarification	Gallery	Analysis	My Submissions	Rejudge
< prev   1 / 1   next >				per-page : 10 20 30 50 100			
Run#	Author	Problem	Status	Lang	Time	Memory	Submission Date
836858	st5471	B	✓ : Accepted	C++	00:04 sec	3896 KB	Mon Dec 16 21:39:02
836823	st5471	B	✗ : Wrong Answer	C++	00:00 sec	5264 KB	Mon Dec 16 21:07:06
836820	st5471	B	✗ : Wrong Answer	C++	00:00 sec	5264 KB	Mon Dec 16 21:06:37
836793	st5471	D	⌚ : Time Limit Exceeded	C++	02:22 sec	4936 KB	Mon Dec 16 20:41:24
836773	st5471	A	✓ : Accepted	C++	00:00 sec	1464 KB	Mon Dec 16 20:19:12
< prev   1 / 1   next >							

## 7-3 ランキングを観覧する

Standing( ランキング) リンクから順位表を観覧することができます。

ACPC2013Day3 UEC Programming Contest 2013

Problem

Standing

Judge

Clarification

Gallery

Analysis

My Submissions

Rejudge

Elapsed Time : 05:00:00    Remaining Time : 00:00:00

< prev | 1 / 2 | next >

per-page : 10 20 30 50 100

Rank	User	AC	Time	A	B	C	D	E	F	G	H	I	J	K	L
1	facepalmparty	11	1305	✓ 00:20	✓ 00:57	✓ 00:46	✓ 00:38	✓ 00:54	✓ (3) 02:06	✓ 03:07	✓ (1) 01:38	-	✓ 02:40	✓ 00:14	✓ (7) 04:40
2	bluebull	9	914	✓ 00:03	✓ 00:20	✓ 00:27	✓ 00:35	✓ 00:59	✓ (1) 01:58	-	✓ 02:37	-	✓ 03:26	✓ 04:24	-
3	miterudake	8	785	✓ 00:04	✓ 00:13	✓ 00:24	✓ 00:32	✓ 01:10	-	✓ (1) 03:41	✓ 03:40	-	✓ 02:58	-	-
4	EviField162	8	898	✓ 00:03	✓ 00:25	✓ 00:19	✓ 00:34	✓ (1) 02:17	✓ 03:03	✓ 02:10	-	-	✓ (4) 04:23	(5)	-
5	logicmachine	8	996	✓ 00:02	✓ 00:10	✓ 00:23	✓ 00:32	✓ 02:34	✓ (1) 02:49	✓ (2) 04:17	-	-	✓ 04:47	-	-
6	Darsein	6	635	✓ (1) 00:42	✓ 00:53	✓ 01:12	✓ 00:31	✓ 02:14	-	-	(1)	-	✓ (2) 04:02	(2)	-
7	y3eadgbe	6	857	✓ 01:32	✓ (1) 01:44	✓ 01:58	✓ (1) 02:13	✓ 02:36	✓ 03:30	-	-	-	(3)	-	-
8	IT_z	5	332	✓ 00:06	✓ 00:18	✓ 00:52	✓ (1) 01:35	✓ 02:19	-	-	(1)	-	(5)	-	-
9	rankalee	5	445	✓ 00:03	✓ 00:08	-	-	-	✓ (1) 00:54	-	-	✓ (2) 02:15	✓ (1) 02:43	(2)	-
10	tanondaVetteru	5	526	✓ 00:03	✓ (2) 00:43	✓ 00:36	✓ (3) 02:25	✓ 03:16	-	-	-	-	(1)	-	-
11	hadroni	5	628	✓ 00:32	✓ 01:09	✓ 01:34	✓ 01:54	✓ (2) 04:38	-	-	-	-	-	-	-
12	yamato	5	642	✓ 00:06	✓ (3) 00:35	✓ 00:50	✓ 02:35	-	-	-	✓ (4) 04:14	-	-	-	-
13	tomoki_imai	4	109	✓ 00:04	✓ 00:16	✓ 00:36	✓ 00:52	-	-	-	-	-	-	-	-
14	BattleProgrammerTeam	4	284	✓ 00:07	✓ 00:34	✓ 01:29	✓ (1) 02:13	(1)	-	-	-	-	-	-	-

ランキングは ACM-ICPC の基準に基づいて行われます。Time（ペナルティの合計）はコンテスト開始から各問題を正解した時間（分）の総和です。ただし、正解した問題の誤答 1 つにつき 20 分のペナルティが加算されます。誤答の回数は赤字で表示されます。正解した問題にはチェックマークが付き、正解した時間が表示されます。

## 7-5 質問をする

Clarification には、コンテストや問題に関するお知らせが掲載されます。

また、コンテストの問題に関する質問は Clarification から行います。フォームに問題番号と質問内容を記入し、送信します。

ACPC2013Day3 UEC Programming Contest 2013

Problem Standing Judge Clarification Gallery Analysis My Submissions Rejudge

Clarifications

Problem D

Request M円すべて、R国もしくはD国の通過に両替する必要がありますか

Response

Thu Sep 05 12:08:17 JST 2013 requested by tanondaVetteru

全部両替する必要はありません。例えば、入出力例1では、9333円は両替されません。

Thu Sep 05 12:10:13 JST 2013 responded by todo

Send a Clarification Request

You can send clarification requests in English and Japanese.

Problem ID

Content

submit

## 7-6 統計情報を閲覧する

Analytic リンクから各種統計情報を閲覧することができます。

Statistics：問題ごとの統計詳細情報

### ACPC2013Day3

Problem Standing Judge Clarification Gallery Analysis My Submissions Rejudge

Statistics

First AC

Ranking

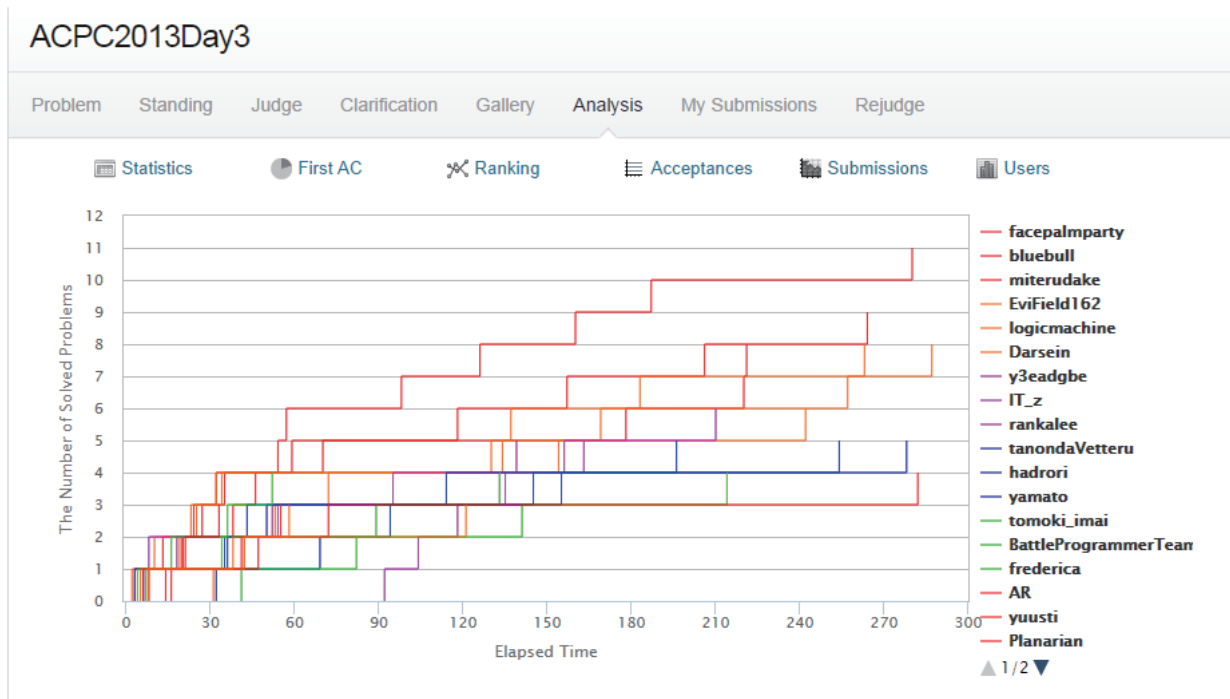
Acceptances

Submissions

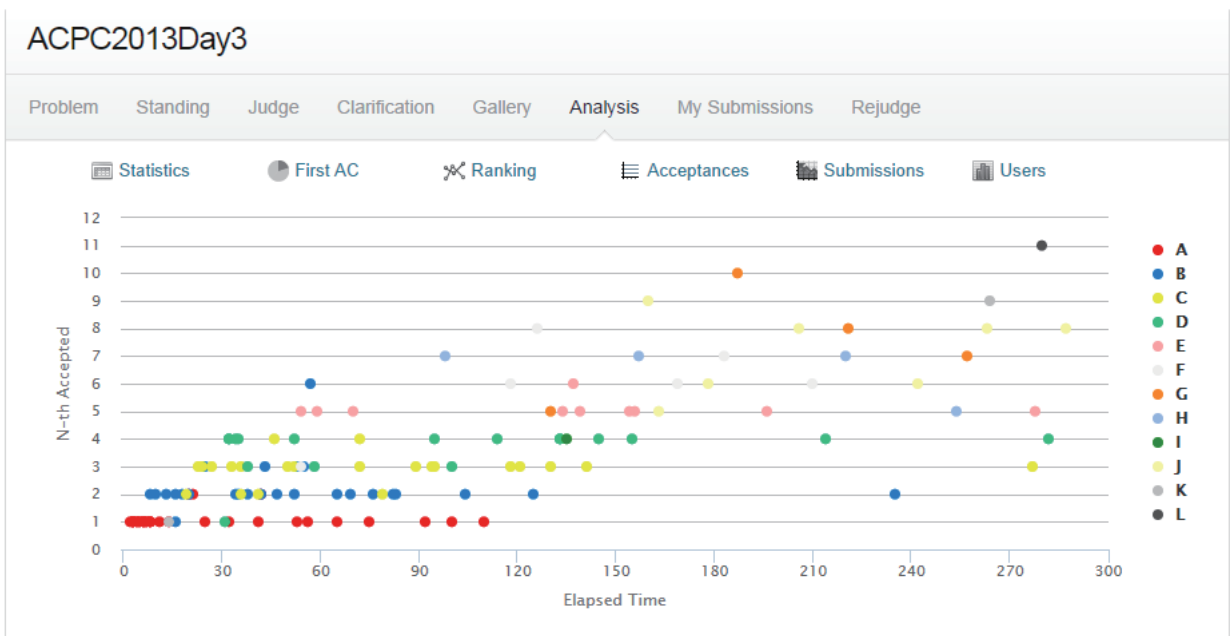
Users

Ttitle	Rate	SB	US	AC	WA	TL	ML	OL	RE
A: Register Phase	86.49%	37	32	32	5	0	0	0	0
B: Time Complexity	57.78%	45	26	26	19	0	0	0	0
C: Mysterious Operator	88.00%	25	22	22	3	0	0	0	0
D: Change	36.17%	47	17	17	30	0	0	0	0
E: Pie Chart is as easy as pie.	71.43%	14	10	10	3	0	0	0	1
F: MLE	46.15%	13	6	6	1	1	2	0	3
G: Get Lost	57.14%	7	4	4	3	0	0	0	0
H: Brainf*ck	19.05%	21	4	4	17	0	0	0	0
I: Reverse Game	33.33%	3	1	1	2	0	0	0	0
J: Avant-garde Art	30.43%	23	7	7	16	0	0	0	0
K: Collector	18.18%	11	2	2	9	0	0	0	0
L: The Return of FizzBuzz	12.50%	8	1	1	7	0	0	0	0
		254	132	132	115	1	2	0	4

Ranking：順位の入れ替わり



Acceptances：問題の正解状況





## タグ

Tools

各問題には付加情報としてタグが付けられています。また、問題に対してタグを設定することもできます。これらのタグは、ユーザの統計情報や検索などに応用されます。

※タグ機能は開発中です。

**Chapter 8-1** カテゴリーの分類

**Chapter 8-2** タグを設定する









# ツール

Tools

AOJ で提供されているツールを紹介します。

5

ツ  
ー  
ル

**Chapter 9-1** オンラインエディタ

**Chapter 9-2** その他のツール

AOJ の提出フォームはプログラミングを行うためのものではありませんでした。プログラミングの方法として、オンラインエディタと呼ばれるサービスを利用することができます。ここでは、数あるオンラインエディタの中から、ProcGarden (<http://sc.yutopp.net/>) を紹介します。

ProcGarden の Web サイトへアクセスすると次のような画面が表示されます。

The screenshot shows the ProcGarden web interface. On the left is a code editor with a C++ program. On the right is a submission control panel with buttons for 'Submit', 'Compile', 'Link', and 'Run'. Below these are status indicators for 'Success', 'Error', and 'Warning'. The bottom right shows the standard output of the program, which is a 5x5 grid of numbers.

```

1 #include <iostream>
2 using namespace std;
3
4 int N;
5 int A[101];
6
7 void print() {
8     for(int i = 1; i <= N; ++i)
9         cout << (i!=1?" ":"") << A[i];
10    cout << endl;
11 }
12
13 int main() {
14     cin >> N;
15     for(int i = 1; i <= N; ++i) cin >> A[i];
16
17     print();
18     for(int j = 2; j <= N; ++j) {
19         int key = A[j];
20         /* insert A[j] into the sorted sequence A[1],...,j-1 */
21         int i = j - 1;
22         while(i > 0 && A[i] > key) {
23             A[i+1] = A[i];
24             i = i - 1;
25         }
26         A[i+1] = key;
27         print();
28     }
29     return 0;
30 }

```

Submission status: C++ (gcc) - 4.8.3, exec? Finished. Compile: Success, 0 (none), 0.2296/5[sec], 24,816/2,097,152[KB]. Link: Success, 0 (none), 0.0478/5[sec], 9,984/2,097,152[KB]. Run: Success, 0 (none), 0.0142/5[sec], 1,668/2,097,152[KB].

runtime command line: ./prog.out

stdout:

```

5 4 3 2 1
4 5 3 2 1
3 4 5 2 1
2 3 4 5 1
1 2 3 4 5

```

5

5 4 3 2 1

画面左側がエディタ、右上がコンパイラ、右下が標準入出力エリアとなっています。

## エディタ

プログラムをエディタ部分に直接記述します。右上の設定アイコンから、キーやタブの詳細を設定することができます。

## コンパイラ

まず、コンパイラ選択部分から適切なコンパイラを選んでください。

## 標準入力

プログラムに入力として必要な数字や文字を標準入力部分に入力します。

## プログラムの実行

右上の青い「Submit」ボタンを押すか、エディタ部分で「Ctrl + エンターキー」を入力すると、プログラムをコンパイルし実行することができます。プログラムの実行結果は、標準出力のエリアに表示されます。プログラムに文法上の間違いがあると、コンパイルエラーが表示されます。

- 2011/08/23 チュートリアルドキュメント公開致しました。
- 2012/07/13 チュートリアルドキュメント改訂致しました。
- 2012/08/10 コンパイラ情報を更新致しました。
- 2013/04/08 ユーザステータス・コースについて追記致しました。
- 2014/10/15 タグ・コンテストについて追記致しました。
- 2015/08/03 チュートリアルドキュメントを修正致しました。

