

সিএমএস এবং আইওআই টাস্কটাইপ পরিচিতি

রেজওয়ান আরিফিন

২২ ফেব্রুয়ারী ২০১৯

সিএমএস, বা কনটেস্ট ম্যানেজমেন্ট সিস্টেম, একটি প্রোগ্রামিং প্রতিযোগিতা আয়োজন করার সিস্টেম। সিএমএসটি সাধারণ হতে এবং বিভিন্ন আন্তর্জাতিক মানের প্রোগ্রামিং প্রতিযোগিতার জাজিং, স্কোরিং ইত্যাদি পরিচালনার জন্য ডিজাইন করা হয়েছে। তবে, ২০১২ সালের সেপ্টেম্বরে ইতালিতে অনুষ্ঠিত আন্তর্জাতিক অলিম্পিয়াড ইন ইনফরম্যাটিক্সের জন্যই সিএমএস প্রথম তৈরি করা হয়, এবং সেই থেকেই এটি আইওআই এবং বিভিন্ন রিজিওনাল অলিম্পিয়াড ইন ইনফরম্যাটিক্স (যেমন: Central-European Olympiad in Informatics, Baltic Olympiad in Informatics, Japanese Olympiad in Informatics, Asia Pacific Olympiad in Informatics ইত্যাদি)-এর প্ল্যাটফর্ম হিসাবে ব্যবহৃত হয়ে আসছে।

এই ডকুমেন্টে সিএমএস এর কন্টেন্ট ওয়েব সার্ভারের ব্যবহারবিধি এবং আইওআই এর বিভিন্ন টাস্কটাইপ ও তাদের কম্পাইল ও সাবমিট করার নিয়ম আলোচনা করা হবে। ডকুমেন্টটি একটু বড় হয়ে যেতে পারে, তাই তুমি যদি নতুন হও তাহলে সম্পূর্ণ না পরলেও Scoring এবং IOI Task Types সেকশনটি অবশ্যই পড়বে।

1. ‘Overview’ Tab

সিএমএস এ লগইন করার পরেই তুমি এই ট্যাবটি দেখতে পাবে। এটি দেখতে নিচের মত হতে পারে:

Testing Contest

Logged in as Rezwan Arefin (RezwanArefin01) Logout Automatic

Server time: 01:30:02
Time left: 95212:29:57

Overview

Communication

A Plus B

Statement

Submissions

SEARCH

Statement

Submissions

FRACTION

Statement

Submissions

GAME

Statement

Submissions

Documentation

Testing

Overview

General information

The contest is currently running.
The contest started at Jan 1, 2000, 6:00:00 AM and will end at Jan 1, 2030, 6:00:00 AM.

Task overview

Task	Name	Time limit	Memory limit	Type	Files
aplusb	A Plus B	1.000 second	256 MiB	Batch	aplusb[.cpp .c .pas]
search	Interactive Search	1.000 second	256 MiB	Communication	search[.cpp .c .pas]
fraction	Simplify the Fraction	1.000 second	256 MiB	Output only	output_01.txt output_02.txt output_03.txt output_04.txt output_05.txt
game	Alice and Bob Plays a Game	1.000 second	256 MiB	Two steps	alice[.cpp .c .pas] bob[.cpp .c .pas]

Contest Management System is released under the GNU Affero General Public License.

এখানে ‘Task Overview’ সেকশনে তুমি টাস্কগুলার একটা সারসংক্ষেপ দেখতে পাবে। এখানে কলামগুলোতে যথাক্রমে টাস্ক এর কোডনেম, টাস্ক এর পুরা নাম, টাইম লিমিট, মেমরি লিমিট, টাস্কটাইপ এবং কি কি ফাইল সাবমিট করা যাবে সেটা লেখা থাকবে। এখানে টাস্ক টাইপ এর মানে কি সেটায় পরে আসছি। (‘IOI Task Types’ সেকশন দেখ)

2. ‘Communication’ Tab

‘Communication’ ট্যাব এ গেলে তুমি দুইটি সেকশন দেখতে পাবে:

Communication

Announcements

Updated Constraint for Task 'aplusb'

1:36:28 AM

Please re-download the problem statement again for problem 'aplusb'. We've fixed some problems in the statement.

Questions

Subject

Text

Ask question

Reset

‘Announcement’ সেকশনে তুমি কন্টেস্ট এডমিনদের সকল প্রতিযোগীদের উদ্দেশ্যে কোন মেসেজ থাকলে সেগুলো দেখতে পাবে।

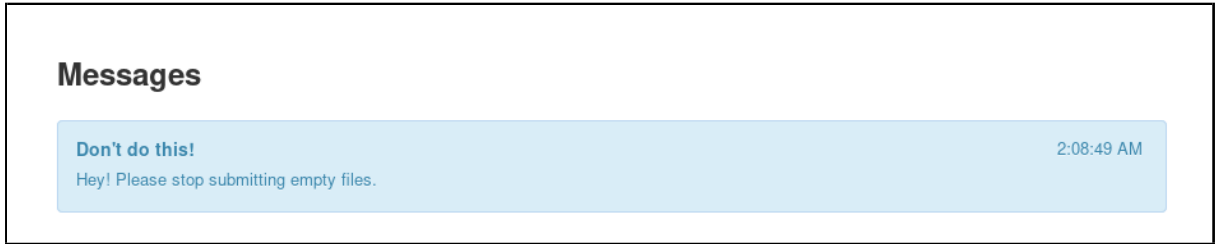
এছাড়াও এখানে একটা ‘Questions’ সেকশন দেখতে পাবে। এটার মাধ্যমে তুমি কন্টেস্ট এডমিনদের কাছে কোন প্রশ্ন থাকলে সেটি জিজ্ঞাসা করতে পারবে। তবে এক্ষেত্রে তোমাকে কিছু বিষয় মাথায় রাখতে হবে:

১. তোমার প্রশ্নের অবশ্যই একটা সুস্পষ্ট মানে থাকতে হবে। এরকম কিছু বলা যাবে না যে: “Give me test 1”, “Please check my submission”, “Why am I getting WA”।
২. তোমার প্রশ্ন এমন হতে হবে যাতে হ্যাঁ/না দিয়ে একটা সুস্পষ্ট উত্তর দেওয়া যায়। যেমন তুমি যদি এরকম প্রশ্ন কর: “Isn’t it true that ...?” – এর উত্তরে হ্যাঁ/না বললে কনফিউসন তৈরি হতে পারে। তাই তোমাকে সব সময় একটা পসেটিভ প্রশ্ন করতে হবে। প্রশ্ন সব সময় “Is it true that ...?” দিয়ে লেখার চেষ্টা করবে।
৩. তোমার প্রশ্নে কেবল মাত্র এই পাঁচ ধরনের উত্তর ছাড়া অন্য কোন উত্তর দেওয়া হবে না:^১
 - “YES”
 - “NO”
 - “ANSWERED IN TASK DESCRIPTION (EXPLICITLY OR IMPLICITLY)”
 - এটার মানে টাস্ক স্টেটমেন্টে তোমার প্রশ্নের উত্তর আছে। তাই তোমাকে টাস্ক স্টেটমেন্ট আবার ভাল করে পরতে হবে।

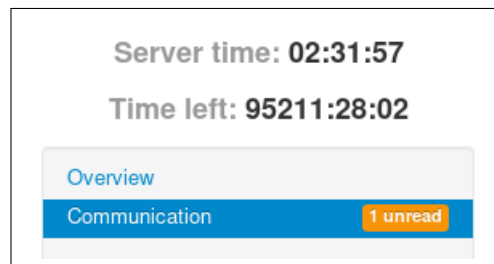
^১Source: “Clarification Requests” Section of IOI Rules

- “INVALID QUESTION” – এটার মানে তোমার প্রশ্নে শুধু YES/NO দিয়ে উত্তর দেওয়া যাচ্ছে না। এক্ষেত্রে তুমি প্রশ্নটিকে পুনরায় ভাল করে লিখে পাঠাতে পার।
- “NO COMMENT” – মানে তুমি এমন কিছু জানতে চেয়েছ যেটা আসলে কোন প্রতিযোগীকে দেওয়া যাবে না।

এছাড়াও সিএমএস এ কোন এডমিন চাইলে শুধু মাত্র তোমাকে উদ্দেশ্য করে কোন মেসেজ পাঠাতে পারবেন। তাহলে ‘Communication’ ট্যাব এ তুমি ‘Messages’ নাম এ আরেকটি সেকশন দেখতে পাবে এরকম –

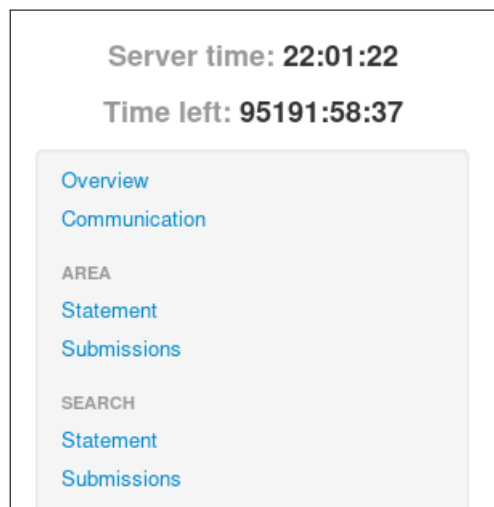


যখন তোমার কোন প্রশ্নের উত্তর দেওয়া হবে, বা কোন Announcement আসবে, অথবা তোমার কাছে কোন মেসেজ দেওয়া হবে তখন ‘Communication’ ট্যাব এর পাশে তুমি এরকম ‘unread’ লেখা দেখতে পাবে:



3. Task ‘Statement’ and ‘Submissions’ Tab

প্রতিটি প্রবলেমের জন্য তুমি একটি ‘Statement’ এবং ‘Submissions’ ট্যাব দেখতে পাবে:



‘Statement’ ট্যাব এ তুমি প্রবলেমটির স্টেটমেন্ট ডাউনলোড করতে পারবে:

Find the Area (area) description

Statement

The statement for this task is available in multiple versions, in different languages. You can see (and download) all of them using the list on the right. Some suggested translations follow.

Statement in Bengali

Statement in English

- Bengali
- English

Some details

Type	Batch	
Time limit	1.000 second	
Memory limit	64.0 MiB	
Compilation commands	C++11 / g++	/usr/bin/g++ -DEVAL -std=gnu++11 -O2 -pipe -static -s -o area grader.cpp area.cpp
	C11 / gcc	/usr/bin/gcc -DEVAL -std=gnu11 -O2 -pipe -static -s -o area grader.c area.c -lm
	Pascal / fpc	/usr/bin/fpc -dEVAL -O2 -XSs -oarea grader.pas

Attachments



‘Some details’ সেকশনে তুমি প্রবলেমটির টাইম লিমিত, মেমরি লিমিট দেখে নিতে পার।
নিচের দিকে ‘Attachments’ নামে একটি সেকশন দেখতে পাবে। এখানে প্রবলেমটি সমাধান করার জন্য প্রয়োজনীয় কিছু ফাইল দেওয়া থাকবে। যেমন, Output Only Task এর জন্য কিছু Input File, প্রবলেমের Grader ইত্যাদি।
প্রবলেমের ‘Submissions’ ট্যাব এ তুমি প্রবলেমটির জন্য তোমার সল্যুশন সাবমিট করতে পারবে। এই ট্যাব এ গেলে উপরেই তোমার এই প্রবলেমের জন্য মোট স্কোর কত হয়েছে এপর্যন্ত তা দেখানো হবে (বিস্তারিত Scoring সেকশনে)।
সল্যুশন সাবমিট করার জন্য ‘Submit a solution’ সেকশনে সঠিক ফাইলের নামের পাশে ‘Browse’ বাটনে ক্লিক করে তোমাকে সঠিক ফাইলটি আপলোড করতে হবে (একটি প্রবলেমে একাধিক ফাইল ও আপলোড করতে হতে পারে, IOI Task Types সেকশন দেখ)। এরপর নিচে ড্রপ-ডাউন লিস্ট থেকে কি ল্যাঙ্গুয়েজ ব্যবহার করেছ সেটি সিলেক্ট করে সাবমিট বাটনে ক্লিক করবে।

Triplets (triplets) submissions

Score:
50 / 100

Submit a solution

triplets:

Browse...
No file selected.

C++11 / g++

Submit
Reset

Previous submissions

Date and time	Status		Score	Files
Feb 18, 2019, 6:03:05 AM	Evaluated	details	50 / 100	Download
Feb 18, 2019, 5:47:54 AM	Evaluated	details	50 / 100	Download

“Previous submissions” সেকশনে তুমি তোমার আগের সব ‘Submission’ এর লিস্ট এবং প্রাপ্ত স্কোর দেখতে পাবে। প্রতিটি সাবমিশনের পাশে তুমি ‘Details’ নামে একটি লিঙ্ক দেখতে পাবে। সেটিতে ক্লিক করে তুমি সেই সাবমিশন এর সম্পূর্ণ ফলাফল দেখতে পাবে। এখানে তোমাকে প্রতিটি সাবটাস্ক এর প্রতিটি টেস্ট এর Verdict দেখানো হবে:

Submission details

> Subtask 1

(10 / 10)

> Subtask 2

(15 / 15)

> Subtask 3

(20 / 20)

> Subtask 4

(20 / 20)

> Subtask 5

(35 / 35)

Compilation output

Compilation outcome:	Compilation succeeded
Compilation time:	0.332 sec
Memory used:	16.9 MiB

Standard output

Standard error

Close

যেকোনো একটি সাবটাস্ক এর উপরে ক্লিক করে তুমি ওই সাবটাস্ক এর সমস্ত টেস্ট এর আলাদা আলাদা Verdict দেখতে পাবে।

4. ‘Documentation’ Tab

এখানে তুমি C/C++ বা অন্যান্য ল্যাঙ্গুয়েজ এর Documentation পাবে। এছাড়াও এখানে বিভিন্ন Verdict এর মিনিং লেখা থাকবে। প্রয়োজনে সেগুলো দেখে নিতে পার।

5. Scoring

IOI তে স্কোরিং সাধারণ Online Judge এর থেকে একটু ভিন্ন ভাবে হয়। একটা উদাহরণ দিয়ে এটা বুঝানোর চেষ্টা করি। মনে কর তোমাকে একটা প্রবলেমে একটা অ্যারে ইনপুট দেওয়া হবে। আর সাবটাস্ক দেওয়া আছে তিনটা:

১. অ্যারেটির যেকোনো সংখ্যা সর্বোচ্চ ১০ (২০ পয়েন্ট)।
২. অ্যারেটির সকল সংখ্যা সমান (৩০ পয়েন্ট)।
৩. অ্যারেটির যেকোনো সংখ্যা সর্বোচ্চ ১০০০০ (৫০ পয়েন্ট)।

এখন প্রথম সাবটাস্কের ২০ পয়েন্ট হয়ত তুমি কোন ব্রুটফোর্স সল্যুশন লিখে নিয়ে নিতে পারবে। কিন্তু এর পরের সাবটাস্ক এর জন্য কি করবে? তুমি মনে কর কোন একটা বিশেষ সল্যুশন খুঁজে পেলে যেইটা শুধুমাত্র সব সংখ্যা সমান হলেই কাজ করে, সেটা যদি সাবমিট দাও তাহলে তো প্রথম সাবটাস্ক আবার ভুল হয়ে যাবে!

এই সমস্যা দূর করার জন্য সিএমএস-এ প্রত্যেকটি সাবটাস্ক আলাদা আলাদা করে জাজ করা হবে! অর্থাৎ তুমি যদি প্রথমবার সাবমিট করে ২০ পয়েন্ট পাও প্রথম সাবটাস্ক এ কিন্তু বাকিগুলোতে ০ পাও, আবার আরেকবার সাবমিট করে দ্বিতীয় সাবটাস্ক এ ৩০ পয়েন্ট পাও কিন্তু প্রথমটাতে ০ পাও, তাহলে তোমার মোট স্কোর সাধারণ জাজ এর মত ৩০ না হয়ে এখানে ৫০ হবে। অর্থাৎ, তোমার মোট স্কোর হবে, প্রত্যেকটি সাবটাস্ক এ তোমার যেকোনো সল্যুশন সর্বোচ্চ যে স্কোর পেয়েছে, সেগুলোর যোগফল।

তার মানে, যদি তোমাকে ৬-৭ টা সাবটাস্ক ও দেওয়া হয়, আর প্রত্যেকটাতে কোন কঠিন শর্ত দেওয়া থাকে, তোমার ওইসকল শর্ত চেক করার দরকার নাই তোমার সল্যুশনের মধ্যে। তুমি যদি আলাদা আলাদা সাবটাস্ক এর জন্য সল্যুশন বের করতে পার তাহলে আলাদা আলাদা করেই সাবমিট করতে পার। তোমার মোট স্কোর এ আলাদা আলাদা সাবটাস্ক এ পাওয়া স্কোর যোগ হয়ে যাবে।

6. IOI Task Types

IOI তে মোট চার ধরনের টাস্ক আসে। এগুলো সম্পর্কে বিস্তারিত আলোচনা করা যাক (এখানে শুধুমাত্র C/C++ Language এর জন্য আলোচনা করা হল; Java/Pascal এর জন্য একই ভাবে করা যাবে):

6.1. Batch Task

এইধরনের টাস্ক তুমি সকল Online Judge এই দেখে থাকবে। এই ধরনের টাস্ক এ সাধারণ ভাবেই তোমাকে কিছু ইনপুট দেওয়া থাকবে আর তোমাকে কিছু জিনিস হিসাব করে একটা আউটপুট দিতে হবে। এইধরনের টাস্ক এ ইনপুট আউটপুট ২ ভাবে হতে পারে –

6.1.1. Without Grader

এখানে তোমাকে standard input stream থেকে অথবা কোন বলে দেওয়া Input File থেকে ইনপুট নিয়ে standard output stream এ বা কোন Output File এ আউটপুট দিতে হবে। এই ধরনের প্রবলেম তুমি সকল Online Judge-এই দেখে থাকবে। তাই এটা নিয়ে বেশি কিছু বলার নাই।

6.1.2. With Grader

কোন কোন টাস্ক এ তোমাকে একটা Grader দেওয়া থাকবে। Grader হল এমন একটা প্রোগ্রাম যে ওই প্রবলেমের সমস্ত ইনপুট-আউটপুট এর কাজ করে দেবে। তোমাকে শুধু মাত্র Core লজিক লিখতে হবে। কোন প্রবলেমের জন্য Grader তুমি ‘Attachment’ সেকশনে ডাউনলোড করতে পারবে। একটা Grader এর ভিতরে সাধারণত এই File গুলো থাকে – code.cpp, grader.cpp, code.h, compile.sh, compile.bat, sample.in, sample.out। এইগুলো কোনটা কি এবার সেটাতে আসা যাক:

- grader.cpp – এইটা হল তোমার প্রোগ্রাম এর মূল অংশ। অর্থাৎ এইখানে প্রোগ্রামের main() function থাকবে। এই ফাইল সঠিকভাবে ইনপুট নিয়ে তোমার লেখা একটা ফাংশনকে কল করে তোমার হিসাব করা আউটপুট নিয়ে নেবে, এবং সঠিক ফরম্যাটে সেটি প্রিন্ট করবে। এই ফাইলটি তোমার এডিট না করাই ভাল।
- code.cpp – এখানে তোমাকে তোমার কোড লিখতে হবে। এইখানে এক বা একাধিক ফাংশনের প্রোটোটাইপ দেওয়া থাকবে, সেগুলো তোমাকে সম্পূর্ণ করতে তবে তোমাকে অবশ্যই code.h হেডার ফাইলটি ইনক্লুড করতে হবে। এই ফাইলে তুমি তোমার ইচ্ছা মত যেকোনো যেকোনো গ্লোবাল ভেরিয়েবল বা সাহায্যকারী ফাংশন লিখে নিতে পারবে। সাবমিট করার সময় শুধুমাত্র এই ফাইলটিই তোমাকে সাবমিট করতে হবে।
- code.h – এই হেডার ফাইলটি grader.cpp এবং code.cpp উভয়ে ব্যবহার করবে। এইখানে ২টা ফাইল এর মধ্যে কমন যেই ফাংশন গুলো ব্যবহার হবে সেইগুলো লেখা থাকবে। এটিও তোমার এডিট করার প্রয়োজন হবে না।
- compile.sh / compile.bat – তুমি যদি Linux ব্যবহারকারী হয়ে থাক তাহলে এই ফাইলটি রান করলে grader.cpp এবং code.cpp একসাথে কম্পাইল হবে। আর Windows ব্যবহারকারী হলে compile.bat ব্যবহার করতে হবে। তবে অনেকক্ষেত্রে compile.bat ফাইলটি নাও দেওয়া থাকতে পারে। কারণ IOI তে শুধুমাত্র Linux Machine ব্যবহার করতে দেওয়া হয়।
- sample.in / sample.out – এটা দেওয়া থাকতেও পারে বা নাও পারে। এখানে ওই প্রবলেম এর স্যাম্পল ইনপুট-আউটপুট দেওয়া থাকে। এইরকম অনেক গুলো ফাইল ও থাকতে পারে।

এখন একটা উদাহরণ দেখা যাক। মনে কর তোমাকে ‘A + B Problem’ Grader এর সাহায্যে সমাধান করতে হবে। স্বাভাবিক ভাবে তুমি কি করতে? এইরকম একটা প্রোগ্রাম লিখতে –

aplusb.cpp

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    int a, b;
    cin >> a >> b;
```

```
cout << a + b << endl;
}
```

এখন Grader এর কাজ হল এই ইনপুট-আউটপুট এর কাজ টা করে দেওয়া। এই প্রবলেমের জন্য grader.cpp এইরকম দেখতে হবে –

grader.cpp

```
#include <bits/stdc++.h>
#include "code.h"
using namespace std;

int main() {
    int a, b;
    cin >> a >> b;
    cout << add(a, b) << endl;
}
```

দেখ এইখানে add(a, b) ফাংশনটি ব্যবহার হয়েছে কিন্তু ইমপ্লিমেন্ট করা হয় নি। এই ফাংশনের প্রোটোটাইপ থাকবে code.h ফাইল এ, আর এটি ইমপ্লিমেন্ট করবে তুমি code.cpp ফাইল এ। যেমন তোমার code.cpp এইরকম হতে পারে:

code.cpp

```
#include "code.h"
int add(int a, int b) {
    return a + b;
}
```

মানে তোমাকে কোন ইনপুট-আউটপুট নিয়ে চিন্তা করতে হল না! শুধু মাত্র এই ফাইলটিই তুমি সাবমিট করবে। তবে ভুল করে grader.cpp ফাইলটা সাবমিট করে ফেল না যেন।

6.1.3. Compiling Grader

এখন আসা যাক তুমি এই প্রোগ্রামটি কম্পাইল করবে কি করে:–

Linux Machine এ কম্পাইল করা খুবই সোজা। তোমাকে grader.cpp আর code.cpp একসাথে কম্পাইল করতে হবে। এইজন্য তুমি যেখানে এইগুলো একসাথে রেখেছ সেখানে Terminal খুলে এই কমান্ড দিবে – `g++ grader.cpp code.cpp -o code`, এটা ওই ডিরেক্টরিতে code নাম এ একটি Executable file বানিয়ে দেবে। এর পরে সেটি তুমি রান করবে `./code` কমান্ড দিয়ে। যদি তুমি c++11 বা c++14 standard ব্যবহার করে থাক আর কম্পাইল হওয়ার সময় এইগুলার সাথে সম্পর্কিত কোন এরর পাও, তাহলে g++ এর পরে একটা স্পেস দিয়ে `-std=c++11` লিখে নেবে।

Windows এ কম্পাইল করার আগে তোমাকে শিওর হতে হবে যে g++ তোমার PATH variable এ আছে কিনা। এইটা কিভাবে চেক করতে হয় সেটা Google করে দেখে নিতে পার। অনসাইট কন্টেন্টে সম্ভবত এইটা ঠিক করেই দেওয়া হবে। এই পর্ব হয়ে গেলে তুমি Linux এর মত করেই কম্পাইল করতে পারবে। যেই ডিরেক্টরি তে কোডগুলো আছে সেখানে যেয়ে Shift + Right Click করে Open Command Prompt here এই ধরনের

একটা লেখা পাবে, সেটা দিয়ে ওই ডিরেক্টরিতে CMD খুলবে। এর পরে Linux এর মত করেই কম্পাইল এবং রান করতে পারবে।

যদি এত ঝামেলা করতে না চাও তাহলে তুমি সরাসরি compile.sh / compile.bat ফাইল গুলোও ব্যবহার করতে পার। মূলত এইগুলোতে ওই লাইনটাই লেখা থাকবে। কিন্তু Windows এ compile.bat ব্যবহার করতে চাইলেও g++ কে PATH ভ্যারিয়েবলে থাকতে হবে।

6.2. Communication Task

এই ধরনের টাস্ক কে Interactive Task ও বলা হয়। এইধরনের টাস্ক তুমি সাধারণত অন্যান্য Online Judge এও দেখে থাকবে। এইধরনের টাস্ক এ কোনও নির্দিষ্ট ইনপুট থাকে না। এখানে তোমার প্রোগ্রাম আরেকটা জাজ প্রোগ্রামের কাছে প্রশ্ন করে কিছু ইনপুট চেয়ে নেয়। আর তুমি সর্বোচ্চ কয়বার প্রশ্ন করতে পারবে সেটার উপরে একটা লিমিট থাকে। এইধরনের প্রবলেম সম্পর্কে এইখানে আরও বিস্তারিত জানতে পার: <https://codeforces.com/blog/entry/45307>।

উপরের টিউটোরিয়াল এ শুধুমাত্র standard input / standard output দিয়েই জাজ প্রোগ্রামের সাথে যোগাযোগের কথা বলা আছে। সাধারণ সকল জাজেই এইরকম প্রবলেম পাবে। কিন্তু সিএমএস এ আরেকভাবে Interaction করা যায়, সেটা হল Grader এর মাধ্যমে Interaction!

সেখানে Grader প্রোগ্রাম তোমাকে কিছু ফাংশন কল করার সুযোগ করে দিবে। তুমি জাজ প্রোগ্রাম এর কাছে যে প্রশ্ন করবে সেটা ওইসকল ফাংশনের মাধ্যমে করে নিবে। ওই ফাংশন গুলো Grader প্রোগ্রামে ইমপ্লিমেন্ট করা থাকবে। যেমন, একটা উদাহরণ দেওয়া যাক:

মনে কর, তোমার সাথে জাজ প্রোগ্রামের এইরকম Interaction হওয়ার কথা, যে তুমি ২টা ইনডেক্স i, j দিবে জাজ কে, আর জাজ তোমাকে বলবে কোন একটা লিস্ট a এর জন্য $a_i < a_j$ সত্য নাকি মিথ্যা, যেখানে a তোমার জানা নাই। তাহলে Codeforces বা অন্যান্য জাজে তোমাকে একটা প্রশ্ন করার জন্য এইরকম একটা প্রোগ্রাম লিখতে হত:

```
//...
int main() {
    //...
    cout << i << " " << j << endl;
    bool reaction;
    cin >> reaction;
    //...
}
```

কিন্তু Grader এর সাহায্যে তুমি এটা আরও সহজে লিখতে পারবে:

```
#include "code.h"

int func() {
    // ...
    bool reaction = ask(i, j);
}
```

– এখানে ask(i, j) ফাংশনটি Grader এর কাছে থাকবে, যেটি ব্যবহার করে তুমি জাজের কাছ থেকে তথ্য

নিতে পারবে।

6.3. Output Only Task

টাস্ক এর নাম শুনেই বুঝা যায় এইধরনের টাস্ক-এ কি করতে হবে! হ্যাঁ, এইধরনের টাস্কে তোমার কাছে শুধু মাত্র Output File চাওয়া হয়। তোমরা যারা Google Code Jam বা Facebook Hacker Cup করেছ তারা এইধরনের টাস্কের সাথে পরিচিত। এখানে তোমার কাছে তোমার কোড চাওয়া হবে না। বরং তোমাকে কিছু ইনপুট ফাইল দিয়ে সেইগুলার জন্য আউটপুট ফাইল চাওয়া হবে। তবে তার মানে এটা না যে তুমি কোড না লিখেই সমস্যাটি সমাধান করে ফেলতে পারবে।

প্রবলেমের জন্য ইনপুট ফাইলগুলো তুমি ‘Attachment’ সেকশনে ডাউনলোড করতে পারবে। একটি .zip ফাইল এর মধ্যে ইনপুটগুলো দেওয়া থাকবে।

এখন কিভাবে এইধরনের প্রবলেমের সমাধান সাবমিট করবে সেটা দেখ। তুমি যখন ‘Submissions’ ট্যাব এ যাবে তখন নিচের মত একটা জায়গা দেখতে পাবে। এখানে তুমি চাইলে একটি একটি করে আউটপুট ফাইল আপলোড করতে পার (বাম পাশে) সঠিক কোডনেমের ফাইলটির পাশে, অথবা একসাথে একটি .zip ফাইল এ করেও আপলোড করতে পার (ডান পাশে)। একটি একটি করে আপলোড করতে হলে তোমাকে ফাইল এর নাম নিয়ে ভাবতে হবে না, কিন্তু ZIP করে আপলোড করতে হলে তোমাকে ফাইল এর নাম ঠিক রাখতে হবে। মানে input_xyz.txt যদি ইনপুট ফাইল হয় তাহলে তোমার আউটপুট ফাইল এর নাম অবশ্যই output_xyz.txt হতে হবে।

Simplify the Fraction (fraction) submissions

Submit a solution

You may submit any subset of outputs in a single submission.

output_01.txt

Browse...

No file selected.

output_02.txt

Browse...

No file selected.

output_03.txt

Browse...

No file selected.

output_04.txt

Browse...

No file selected.

output_05.txt

Browse...

No file selected.

Submit

Reset

submission.zip

Browse...

No file selected.

Submit

Reset

Previous submissions

Time	Status	Score	Files
no submissions			

তবে মনে রেখ, তোমাকে কিন্তু একটা সাবমিশনেই সকল ইনপুট ফাইল সমাধান ‘না’ করলেও হবে। তোমরা যারা উপরে Scoring সেকশন পড়েছ তারা এটা ইতোমধ্যে জান। মনে কর তুমি প্রথম বার শুধুমাত্র ১ম আর ৩য় আউটপুট ফাইল সাবমিট করলে এবং উভয় ফাইলে পয়েন্ট পেলে এখন এর পরে যদি তুমি ২য় আউটপুট ফাইল সাবমিট করবে তখন কিন্তু আর ১ম বা ৩য় ফাইল আপলোড না করলেও চলবে! তুমি আগে ১ম ও ৩য় আউটপুট ফাইলে যেই পয়েন্ট পেয়েছ সেটা থেকেই যাবে সবসময়। মানে প্রত্যেকটা আউটপুট ফাইল একটা আলাদা সাবটাস্ক হিসাবে ধরা হবে এবং সেগুলার আলাদা আলাদা সর্বোচ্চ স্কোর যোগ করা হবে।

এখন, তুমি কিভাবে আউটপুট ফাইল বানাবে? এইটার সবথেকে সহজ উপায় হল Terminal / Command Prompt ব্যবহার করা। মনে কর, তোমার প্রোগ্রামটা কম্পাইল করে code নামে একটা এক্সিকিউট্যাবল ফাইল পেয়েছ। এখন তুমি যদি input.txt ফাইল থেকে ইনপুট নিয়ে তার আউটপুট output.txt ফাইল এ জমা করতে চাও তাহলে code আর input.txt ফাইল দুটিকে একই ডিরেক্টরিতে রেখে সেখান থেকে এই কমান্ডটি রান কর: `./code < input.txt > output.txt`। এটি Linux এবং Windows এর জন্য একই।

অথবা আরেকটা উপায় আছে সরাসরি কোডের মাধ্যমে। তুমি যদি চাও তোমার কোড input.txt ফাইল থেকে ইনপুট নেবে এবং output.txt ফাইল এ আউটপুট দেবে, তাহলে `main()` ফাংশন শুরু করার সাথে সাথে এই দুটি লাইন যোগ করে নাও:

```
// ...
int main() {
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
    // rest of the code here
}
```

এটা মূলত input.txt ও output.txt ফাইল দুটিকে যথাক্রমে stdin এবং stdout এর সাথে ম্যাপ করে নেবে। তুমি যদি অন্য কোন ল্যাপ্রুয়েজ ব্যবহার করে থাক তাহলে সেই ল্যাপ্রুয়েজের ডকুমেন্টেশনে File I/O সেকশন খুঁজে দেখতে পার।

6.4. Two Step Task

এইধরনের টাস্ক “দুই স্টেপ”-এ হয়। অর্থাৎ এইখানে তোমার শুধু মাত্র একটা প্রোগ্রাম লিখলেই হবে না। বরং ২টা বা প্রবলেমের উপরে নির্ভর করে আরও বেশি প্রোগ্রাম লিখতে হতে পারে। যেমন, একটা উদাহরণ দিয়ে দেখানো যাক –

মনে কর এলিস এবং বব স্টেজে জাদু দেখাচ্ছে। এলিস একজন দর্শকের কাছে একটি কাগজ দিবে একটি সংখ্যা লেখার জন্য, এবং বব সেই সংখ্যা না দেখেই বলে দেবে। মূলত এলিস বব কে ইশারায় সংখ্যাটি বলে দিবে। এখন এলিস একই সংখ্যার ইশারা করলে সেটা লোকে ধরে ফেলবে তাই তাকে অন্য সংখ্যার ইশারা করতে হবে।

এখন তোমার কাজ হল এলিস এবং বব ২ জনকেই প্রোগ্রাম করা অর্থাৎ তোমাকে `alice.cpp` এবং `bob.cpp` নাম এ ২টা প্রোগ্রাম লিখতে হবে। `alice.cpp` একটি সংখ্যা ইনপুট নেবে, মনেকর ১ থেকে ১০ এর মধ্যে। এর পরে এই প্রোগ্রাম `bob.cpp` এর আছে অন্য একটি সংখ্যা পাঠাবে, যেটি আবার ১ থেকে ১০ এর মধ্যে হবে, আর সেই সংখ্যা দেখে `bob.cpp` কে মূল সংখ্যাটি বের করতে হবে।

এখানে এই `alice.cpp` আর `bob.cpp` এর মধ্যে Interaction হবে Grader এর মাধ্যমে। অর্থাৎ শুরুতে Grader একটি সংখ্যা ইনপুট নেবে, এর পরে `alice.cpp` এর একটি ফাংশনের কাছে সেই সংখ্যা পাঠাবে। `alice.cpp` এর সেই ফাংশন আরেকটি সংখ্যা রিটার্ন করবে। এর পরে Grader এই সংখ্যাটি `bob.cpp` এর কাছে পাঠাবে এবং সে কি রিপ্লাই দেয় সেটা নেবে এবং মিলিয়ে দেখবে প্রথম সংখ্যা আর এই সংখ্যা সমান কিনা। একই সংখ্যা হলে তোমার প্রোগ্রামটি সঠিক হয়েছে বলে ধরা হবে।

এখানে `alice.cpp` আর `bob.cpp` ফাইল গুলো তোমার লিখতে হবে, উপরে Batch Task এর জন্য যেভাবে বলা হয়েছে ওইভাবে লিখলেই হবে। আর কম্পাইল করার সময় ৩টা ফাইল একসাথে কম্পাইল করতে হবে। মানে এইরকম: `g++ grader.cpp alice.cpp bob.cpp -o magic` – এটি ওই ডিরেক্টরিতে `magic` নামে একটি Executable file বানাবে। সেটতে ইনপুট দিয়ে তোমার প্রোগ্রাম টেস্ট করতে পারবে।

যখন সাবমিট করতে যাবে তখন এরকম জায়গা দেখতে পাবে:

Alice and Bob Plays a Game (game) submissions

Submit a solution

alice: No file selected.

bob: No file selected.

Previous submissions

Time	Status	Score	Files
no submissions			

এখানে এলিসের জায়গায় এলিসের প্রোগ্রাম এবং ববের জায়গায় ববের প্রোগ্রাম আপলোড করে সাবমিট করবে।