<codeclash>2019

Presented by The Gongoliers

WHEN

November 25, 2019
1 PM – 5 PM

WHERE

Ponaganset High School
137 Anan Wade Road
North Scituate, RI 02857

CONTACT

teams@codeclash.org
(401) 710 – 7500 Ext. 3374

Get involved in the 2019 Code Clash!

Teams of four high-school-aged students compete in a challenging, time-limited tournament to put their computer science skills to the test. Competitors will be tasked with a series of algorithmic challenges of varying difficulty, while having fewer than three hours to come up with their solutions. Programmers of all skill levels are invited and encouraged to compete, as there is something for everyone.

Sign up and spread the word!

High schoolers will compete against one another in this exciting event as they strengthen their computer science, problem solving, and teamwork skills.

Java, C++, C, and Python!

No matter what your favorite language is, you can compete in Code Clash!

OFFICIAL MANUAL

Languages

The contest supports Java, C++, C, Python 2, and Python 3. Teams may use any combination of these languages to solve the challenges.

Software and IDEs

Each team will receive one computer to use in the contest. The Windows 10 computer will be pre-installed with IntelliJ, PyCharm, CLion, Eclipse, Visual Studio Code, and Notepad++. The Microsoft Office 2016 suite will also be available in the event teams would like to type their ideas, create a flowchart, etc. Teams may not access the internet from these computers nor install any additional software.

Challenges

The contest will be composed of a series of challenges, no less than ten challenges. Each challenge will have equal point value, although they may vary in difficulty. Each challenge will have a problem statement, where the challenge is explained in detail. The format of inputs and expected outputs will be detailed in each problem statement. At least one sample test case will be included with each problem statement. Teams must work together to develop a single solution to each challenge that will produce the desired result.

Reading and writing

Each challenge in the contest will require the program to process an input and return an output. Inputs must be read from “standard in” and outputs must be written to “standard out” (also known as the console). Each problem statement will detail the specific inputs and outputs necessary for each challenge.

Submitting solutions

A submission system will already be open in a web browser when teams reach their computers. Teams will press the green submit button to upload their uncompiled source code to this submission system. The source code file name must only contain alphanumeric characters, underscores, and hyphens. Once the submission is confirmed, the submission list page will be shown with the results.

Viewing the results of submissions

The left column of your team web page on the submission system shows an overview of your submissions. It contains all relevant information: submission time, programming language, challenge, and status.

CORRECT	The submission passed all tests: you solved this challenge!
COMPILER-ERROR	There was an error when compiling your program. On the submission details page, you can inspect the exact error. Note that when compilation takes more than 30 seconds, it is aborted and this counts as a compilation error.
TIMELIMIT	Your program took longer than the maximum allowed time for this challenge. Therefore, it has been aborted. This might indicate that your program hangs in a loop or that your solution is not efficient enough.
RUN-ERROR	There was an error during the execution of your program. This can have a lot of different causes like division by zero, incorrectly addressing memory (e.g. by indexing arrays out of bounds), trying to use more memory than the limit, etc. Also check that your program exits with exit code 0!
NO-OUTPUT	Your program did not generate any output. Check that you write to standard out.
OUTPUT-LIMIT	Your program generated more output than the allowed limit. The output was truncated and considered incorrect.
WRONG-ANSWER	The output of your program was incorrect. This can happen simply because your solution is not correct but remember that your out-put must comply exactly with the specifications of the judges.
TOO-LATE	Bummer, you submitted after the contest ended! Your submission is stored but will not be processed anymore.

Viewing the scoreboard

The top of your team page on the submission system shows your team's row in the scoreboard: your position and which challenges you attempted and solved. Via the menu you can view the public scoreboard page with the scores of all teams. The public scoreboard will be frozen one hour into the contest, after which you will only be able to see your team's information and you will not be able to see your team's current rank.

Clarifications

All communication with the judges is to be done through clarifications. These can be found in the right column on your team page on the submission system. Clarification reply broadcasts from the judges and requests sent by you are displayed there. There is also a button to submit a new clarification request to the judges; you can associate a specific challenge or one of the general categories to a request. This clarification request is only readable for the judges. The judges will send a reply to this clarification to all teams. Contest officials will be able to assist teams with technical or non-challenge-related questions in person.

Judging submissions

The contest control system uses a fully automated judging system. Source code submissions will automatically be run through a series of test cases that will determine if the solution is correct. These test cases differ from the publicly visible test cases to prevent hard-coding.

After source code is submitted, it enters a queue. The program will be compiled, executed, and tested on one of the auto judge computers running Linux. If the program takes too long to compile, the compilation will be aborted, and the submission will result in COMPILER-ERROR.

Programs will be restricted in CPU time and memory utilization. Programs will be run in a virtual sandbox, where the program will have access to a single CPU core. After execution of the program, the exit status will be checked. Any non-zero exit code will be considered a runtime error, and your submission will result in RUN-ERROR.

The program output will be compared to the test case output. When comparing program output, it must match the format detailed in the problem statement exactly, except for extra whitespace. The auto judge is very strict.

LANGUAGE REFERENCE

Below is an example challenge that demonstrates standard in/out usage.

The first line of input contains the number of testcases. Each testcase consists of a line containing a name (a single word) of at most 99 characters. For each testcase, output the string “Hello <name>!” on a separate line.

Sample test case:

Input	Output
3 world Jan SantaClaus	Hello world! Hello Jan! Hello SantaClaus!

A solution for this challenge in C++:

```cpp
#include <iostream>
#include <string>

using namespace std;

int main() {
        int ntests;
        string name;

        cin >> ntests;
        for (int i = 0; i < ntests; i++) {
                cin >> name;
                cout << "Hello " << name << "!" << endl;
        }
}
```

A solution for this challenge in Java:

```java
import java.util.*;

class Main {
        public static void main(String[] args) {
                Scanner scanner = new Scanner(System.in);
                int nTests = scanner.nextInt();

                for (int i = 0; i < nTests; i++) {
                        String name = scanner.next();
                        System.out.println("Hello " + name + "!");
                }
        }
}
```

A solution for this challenge in C:

```c
#include <stdio.h>

int main() {
        int i, ntests;
        char name[100];

        scanf("%d\n", &ntests);

        for (i = 0; i < ntests; i++) {
                scanf("%s\n", name);
                printf("Hello %s!\n", name);
        }
}
```

A solution for this challenge in Python:

```
n = int(input())
for i in range(n):
    name = input()
    print('Hello ' + name + '!')
```

The above demonstrated format for reading from standard in and writing to standard out will be the default procedure for all the challenges at Code Clash. The challenges at the competition will be more challenging, but they will all follow this same basic format.

OFFICIAL RULES

1. Team Eligibility

A participating team may consist of no more than four (4) pre-college students ages 13 to 18 (“contestants”).
Each contestant must have a media release waiver signed by a legal guardian.
Each unique school or organization must receive prior permission if they wish to have more than two (2) participating teams.

2. Conduct Rules

Mentors and/or spectators may accompany the contestants to the event, but they will be held in a separate room for the duration of the contest.
Only contestants and contest officials may be allowed in the contest room during the contest, and contestants may not have any communication with mentors or those not on their team. An attempt to collaborate with someone outside of one’s team may result in disqualification.
Contestants are expected to be respectful to all teams and officials.
Contestants should be respectful of the equipment and hardware provided to them. Misbehavior will be investigated and may result in disqualification.
Food and drink must be kept in the dedicated eating area in the contest room and may not be kept near the computers. Contestants may move freely around the contest room as they wish throughout the time of the contest, although they may not move their computer. Restrooms will be available throughout the contest.

3. Allowed Materials

Only one (1) computational device is allowed per team. Calculators are an exception to this rule, and we allow any calculator allowed on the SAT®.
Each team may bring as many paper resources, books, and references as they can fit into one (1) standard pocket folder.
Food, snacks, and/or beverages may be brought, but must be left in the dedicated eating area as per section 2. Light snacks will also be provided to contestants and mentors.
One (1) Windows 10 PC will be provided to each team with the necessary software pre-installed, and no other computational devices will be allowed.
Phones, smartwatches, laptops, pagers, and other computers other than the one provided will not be allowed during the competition. Any devices brought by

the contestants must be stowed or given to a mentor prior to the start of the contest. Any attempt to use another device may result in disqualification.

A keyboard and mouse will be provided with the computer. All teams will use identical equipment and may not bring their own peripherals. Contact the contest coordinators if a hardware substitution is necessary for accessibility purposes.

Competitors may not use any device that can be used for communication during the contest. The internet may not be used or accessed on the computer during the contest.

Competitors may not bring any flash drives or portable storage devices that could be used to transfer external data onto their working computer.

4. Computer Rules

Each Windows 10 computer will be pre-installed with contest IDEs. IntelliJ, PyCharm, CLion, Eclipse, Visual Studio Code, and Notepad++ will be available. The Microsoft Office 2016 suite (including Word) will also be available.

You are not allowed to access the internet during the contest. Only the submission portal and a selection of official language documentation will be whitelisted for access.

Desktop shortcuts will be shown for all software and documentation that contestants can use during the contest.

A web browser will be opened to the submission portal when teams first get to their assigned computers. When submitting solutions to challenges, contestants must upload their uncompiled source code to this submission portal. The submission portal accepts Java, C++, C, Python 2, and Python 3 source code. Source code must be a single file not exceeding 256 kilobytes. Compilation of your program may take no longer than 30 seconds. During execution of your program, there are 524288 kilobytes of memory available. Programs are limited to no more than 5 seconds of CPU time and 1 thread.

All code will be compiled and tested per the details in the official manual. Do not install any software on the computers or attempt to change any settings without permission from a contest official.

5. Contest Rules

Check-in and registration will be from 1:00 PM until 1:30 PM. During this time, a practice/warm-up session will take place, where teams can become familiar with their computers and the submission system. The opening ceremonies will last until 2:00 PM, at which time the contest will immediately begin.

The contest will commence from 2:00 PM until 4:30 PM. Once the contest begins, contestants will be brought to the contest room, which they will not leave for the duration of the contest. Mentors and spectators remain in a separate area and will be able to watch their teams progress.

The main problem statements will be released to both the contestants and mentors, and the contestants may immediately begin their work on the computer. Contestants will code their solutions to each of the challenges and submit them on the submission portal.

If a contestant has a clarification question related to one of the challenges, they must submit their question through the clarification system on the submission portal. Contest officials will not be able to answer questions related to the challenges, although they may assist contestants to use the submission system or troubleshoot other issues.

When a team submits a solution to a challenge, it will be automatically judged, and the team will receive feedback on whether their solution passed all test cases. If the solution fails, the team will be notified, and they will be able to modify their code and submit an infinite number of attempts. If all test cases are passed, the team will be credited with completing that challenge and the total number of minutes since the start of the competition will be added to their time score.

Teams will be ranked first based on how many challenges they have completed, and second based on how their time score is. The team with the most completed challenges will be ranked first. Time scores are used in the event of a tie, the team with the lowest time score will be ranked higher (think golf scores). 15 minutes of penalty time will be added to the time score for each submitted failed attempt to a challenge, including compile errors, runtime errors, and wrong answers. A leaderboard will be shown to both contestants and observers for the 1 hour and 15 minutes of the contest, until 3:15pm. For the remaining 75 minutes of the contest, the leaderboard will be frozen. The final

leaderboard and standings will be revealed to both contestants and mentors at the closing ceremonies.

At exactly 4:30 PM, the submission portal will be locked, and the contest will be concluded. Contestants will be reunited with their mentors and will go to the closing ceremonies. The final standings will be revealed, and the winners will be announced at the ceremony. This ceremony should be concluded by 5:00 PM.

SCHEDULE

Time	Event
1:00pm - 1:30pm	**Visiting Schools Arrive at Ponaganset High School** 1. Teams will check in and be given visitor passes. 2. Student team members will be escorted to the contest room for systems check and the warm-up session. 3. Mentors and Chaperones will be escorted to the mentor lounge for check-in and exchange of documents. 4. Student team members will be escorted to the mentor lounge for refreshments and the opening ceremonies. Note: Each visiting school will have greeters assigned to escort.
1:30pm - 2:00pm	**Opening Ceremony in the mentor lounge** 1. Briefing on rules and procedures. 2. Teams escorted to the contest room.
2:00pm - 4:30pm	**Code Clash Competition in the contest room** • Mentors and Chaperones remain in the mentor lounge and will have no contact with students.
4:30pm - 5:00pm	**Closing Ceremony in the mentor lounge** • Results will be announced and awards will be presented.