
Compiler Breakout
Cloudflight Coding Contest
5th November 2021

Compiler Breakout › Your task

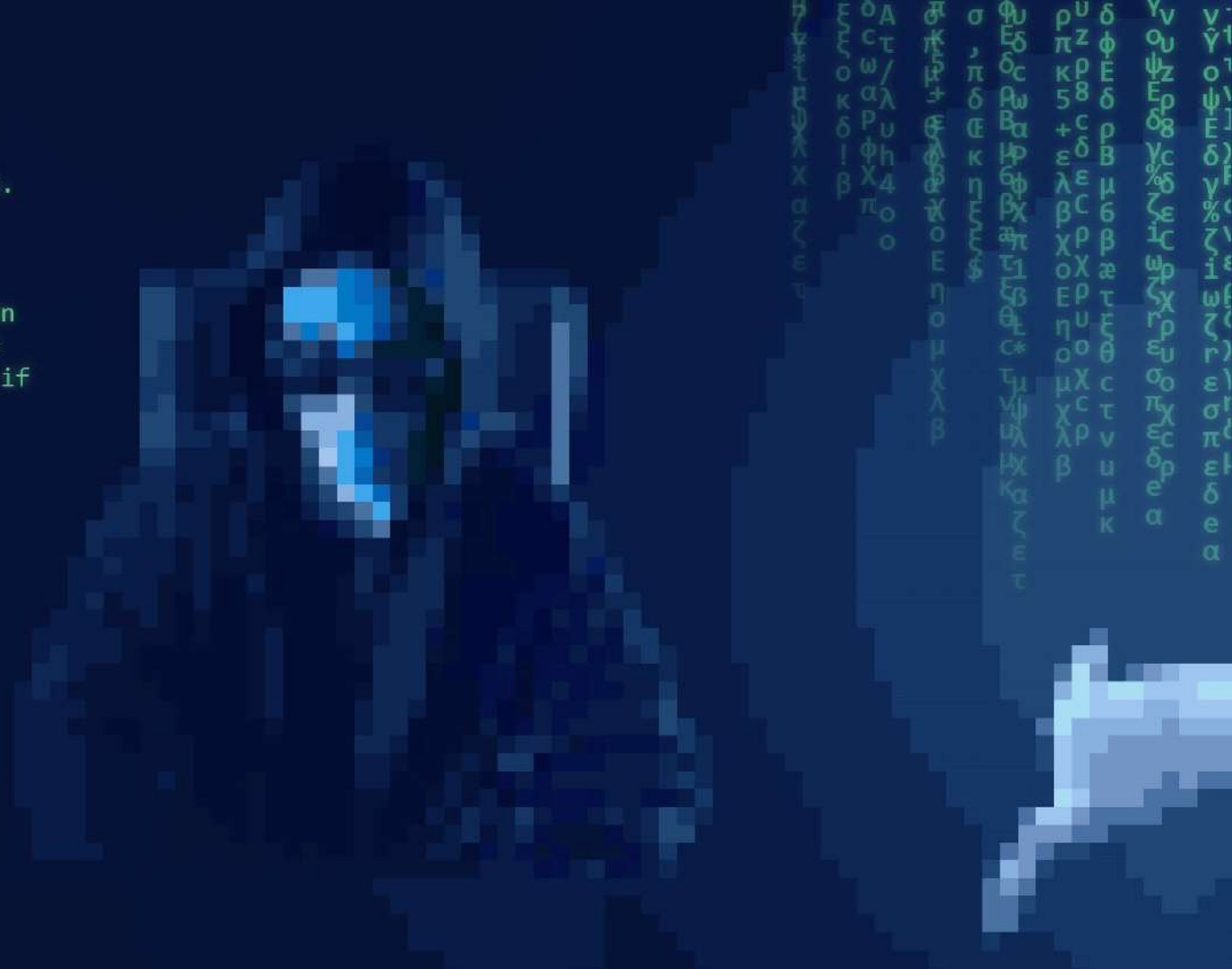
- You are stuck in a computer simulation and will need to interpret some instructions in order to find your way out
- You will write a compiler that can parse code in a new programming language with similar features to existing languages but also some new ones
- Follow the white rabbit to find your way out

It’s almost completely dark.

Empty.

You distinguish some written text floating around but it all feels very robotic, as if you were in a simulation.

The last thing you remember was falling asleep at your computer screen while chasing a white rabbit in a video-game.




LEVEL 1

You are given an input file that contains program code that is written in a new programming language. In this programming language, the code is a stream of tokens separated by some white spaces (line breaks or spaces). A token is a concatenation of characters that are not separated by white spaces or new lines.

Code statements are written in blocks called functions. Those function blocks start with the token **start** and end with the token **end**. Each function contains zero or more code statements. If we refer to a valid code statement as <**statement**> then code statements can be any of the following stream of tokens.

- **print** <**Boolean** | **Integer** | **String**>, this statement prints its value to the screen without extra spaces before or after, which means that the output multiple print statements will always appear concatenated on the screen

Where

Boolean means the token **true** or the token **false**

Integer means a token that represents a non negative integer

String means a token that is not a preserved token or Integer or Boolean

Preserved tokens up till now are:

start, **end**, **print**

More preserved tokens will be added in future levels

- *In this level* the code will only contain one function
- In next levels more statements and definitions will be added

Given a code as described above, output what will be printed on the screen after executing its only function.

	Input	Output
Format	N lineOfCode (repeated N times)	functionOutput
Types	**N (int) number of code lines that follow** **lineOfCode (string) one or more space separated tokens**	**functionOutput (string) the output that print statements in the the given code produce**
Example	6 start print is print this print the print matrix end	isthisthematrix

For easier readability, every statement is shown in one line in the example. This will not be the case in the real input files.


GOOD
LUCK!