

[← main page](#) — COMP 526 Applied Algorithmics

Programming Puzzle: Exam-Cheating Codes

This continuous-assessment exercise consists of a small applied project with algorithmic and programming components, including a real-time **leaderboard** of the competition.

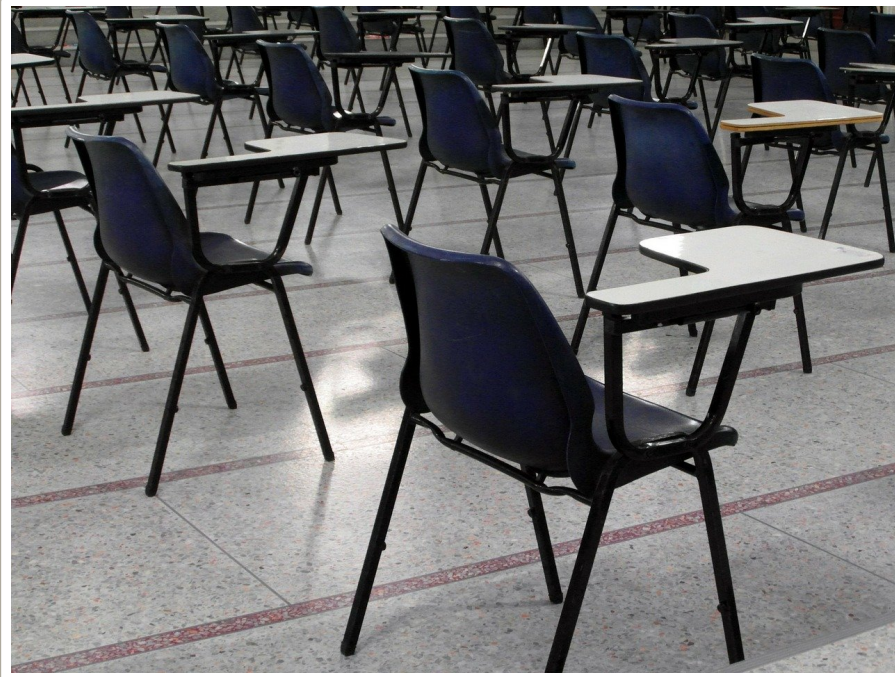
Will you be able to beat your classmates, or even your professor?

As for the Bamboo Trimming Problem, this assignment requires more thinking than coding, and the focus is as much on the *process* of finding solutions as on the actual outcome. It is also an open research problem for which the ultimate best solution is as of yet unknown!

Exam Cheating Codes

Final exams are fast approaching. While you are perfectly well prepared and are eagerly awaiting examination day, your significant other is much less relaxed about an upcoming exam. Actually, *desperate* is more the appropriate term. You decide to help.

The most frightening test, COMP666 🐱, is a multiple-choice exam with 20 questions, and each question requires a single Yes/No answer. You are confident that you can perfectly answer any possible question on behalf of your better half, but of course, the main challenge is how to *communicate* the answers



to your loved one after you left the exam. (To your great relief, the instructor chose not to randomize questions or shuffle answers...)

Clearly, you are not able to enter the exam hall once you have left, but you arrange with your significant other the following code: Honking once (on your moped parked in the side street next to the building) will mean Yes, quickly honking twice means No. Unfortunately, as you figured out from painful experimentation, you have only time to send 10 Yes/No signals before campus police politely removes you (and your irritating source of noise source) from university premises.

*What is the best mark you can **guarantee** for your sweetheart to achieve irrespective of the actual sequence of correct answers?*

Formalization

Clearly, the correct answers to the 20 Yes/No questions of the exam can be given as an array of 20 bits $B[0..20)$, where $B[i] = 1$ (for $i = 0, \dots, 19$) means that the correct answer to the i th question is Yes, whereas $B[i] = 0$ means the answer is No.

Likewise, your honking “message” can be encoded as an array of 10 bits $H[0..10)$.

The task now consists in finding an encoding function $e : \{0, 1\}^{20} \rightarrow \{0, 1\}^{10}$ and a decoding function $d : \{0, 1\}^{10} \rightarrow \{0, 1\}^{20}$ for you respectively your significant other to memorize. Your goal is to get the best *guaranteed* mark for your sweetie, i.e., to find d and e so that $\max_B d_H(d(e(B)), B)$ is as small as possible. Here, d_H denotes the Hamming distance, and the maximum is taken over all possible bit arrays of length 20.

Inputs

Your protocol has to work for any possible set of correct answers B , and it will be tested on all of them.

Design as good a protocol as you can find!

Code template

We prepared a Python implementation of the exam sitting that you will use to evaluate your protocol:

- [Python sources](#)

Obey the comments! Once you have downloaded the code, replace the marked sections in `examCheatingCode.py` with

1. code for your encoding function and
2. code for your decoding function.

To run the simulation, extract the zip archive to a folder and run `python3 sitExam.py` there. The output is the worst mark (minimal number of correctly answered questions) that your loved one could achieve with the help of your protocol.

Deliverables

This is an individual project; each student has to submit their own solution. The submission is on Canvas, split into the following parts; **you have to submit *all* of them**. Note further that only **typed solutions** will be marked, but you can use any format that Canvas accepts.

- ▶ **Part 1: Reflective report describing how you solved the puzzle**
(at most ~300 words)
Briefly describe *how you arrived* at your solution (not yet how it works). Comment on approaches you used and also report on dead ends you tried (what did not work).
- ▶ **Part 2: Explanation of your code** (at most ~300 words)
Explain the *key ideas* behind your code (to be submitted in part 4). How does it achieve to transmit answers to as many questions as claimed in part 4?
A secondary criterion here is *simplicity* of the solution: If you cannot seem to

improve the mark, extra marks can be awarded for clean/simple solution that achieve that guarantee.

► **Part 3: Proof for your impossibility result** (at most ~300 words)

Provide a proof for the impossibility result claimed in part 4, i.e., why a certain mark guarantee cannot be achieved with any thinkable exam-cheat protocol.

► **Part 4: The quality of your solutions and code**

1. your `examCheatingCode.py` source file
2. the guaranteed mark that your code achieves
3. lowest mark that you could prove impossible to achieve (in part 3) as the worst case over all answer vectors with a fixed protocol

Marking Scheme

The overall mark will consist of a weighted average of the four parts:

- 20% for a convincing reflective report
- 30% for a clear and convincing explanation and code simplicity
- 20% for a valid impossibility proof
- 30% for the quality of the bounds (high guarantee and low impossibility result)

Collaboration

This programming puzzle is an individual project, and you have to submit your own solution. In particular, the reports of your solution must be a single-author document. As for all assessments, the University of Liverpool Policy on Academic Integrity applies; please refer to the [Canvas page](#) for details.

Collaboration in small groups (not more than five students) on a *conceptual level* (discussing ideas, not sharing entire solutions) is accepted, *but it must be declared in the description document*, including proper mention of others' contributions. Moreover, the report itself must be written by each student individually.



Substantial similarity between your and other students' solutions is deemed collusion and thus a breach of our academic integrity policy, even if you were unaware of your work being copied. What counts as substantial is decided by markers; Turnitin similarity scores are only one input for that.

Leaderboard

We run a (voluntary, anonymous) *leaderboard* of the current best solutions. When you have a code tried in the simulation, use the below form to share your achievement with the rest of the class!

Exam-cheating code leaderboard

Enter the your achieved guaranteed mark (number of correct questions, as output by the simulation) you were able to achieve.

 **bhattbalkrishna7@gmail.com** (not shared) 
[Switch account](#)

* Required

Your alias *

Your answer

Achieved guaranteed mark *

Your answer

Submit

Clear form

Highscores

The plots below show all answers over time. Recall that higher is better.

New submissions are immediately added at the right end, but might take a few seconds and refreshing before they show up.

Your alias	Achieved goal
The Great One	
KJ	
The One Above All	
Yin	
ඔ^*~*^ඔ	
Alafin Oluwatobi (2022/11/30)	
🌀.🌀?	
🌀.🌀?	
ඔ^*~*^ඔ	
zy	

Sebastian Wild

Contact:

- ▶ [sebawild at gmail](mailto:sebawild@gmail.com) · PGP
- ▶ [wild at liv.ac.uk](mailto:wild@liv.ac.uk) · PGP
- ▶ ~~[wild at uwaterloo.ca](mailto:wild@uwaterloo.ca) · PGP~~
- ▶ ~~[wild at cs.uni-kl.de](mailto:wild@cs.uni-kl.de)~~

Elsewhere:

- ▶ GitHub profile
- ▶ LinkedIn profile
- ▶ ORCID: 0000-0002-6061-9177
- ▶ Google Scholar profile

Quick links:

- ▶ my publications
- ▶ my library

- ▶ Website at UoL
 - ▶ intranet site at UoL
 - ▶ TCS @ Liverpool
 - ▶ (Old) website TU KL
- ▶ DBLP publication list
 - ▶ Semantic Scholar Author Page
 - ▶ arXiv Author ID