

Educational Opportunities and Challenges of AI Code Generation

renAIssance or apocAIypse?

Brett A. Becker

Paul Denny

James Finnie-Ansley

Andrew Luxton-Reilly

James Prather

Eddie Antonio Santos



Juho Leinonen

with special guest, phantom Juho



Brett A. Becker

Paul Denny

James Finnie-Ansley

Andrew Luxton-Reilly

James Prather

Eddie Antonio Santos



Educational Opportunities and Challenges of AI Code Generation

renAIssance or apocAlypse?

Professor catches student cheating with ChatGPT: 'I feel abject terror'

Exclusive: OpenAI Used Kenyan Workers on Less Than \$2 Per Hour to Make ChatGPT Less Toxic

Large computer language models carry environmental, social risks



Pausing AI Developments Isn't Enough. We Need to Shut it All Down

Chat Overflow: Artificially Intelligent Models for Computing Education

renAIssance or apocAlypse?

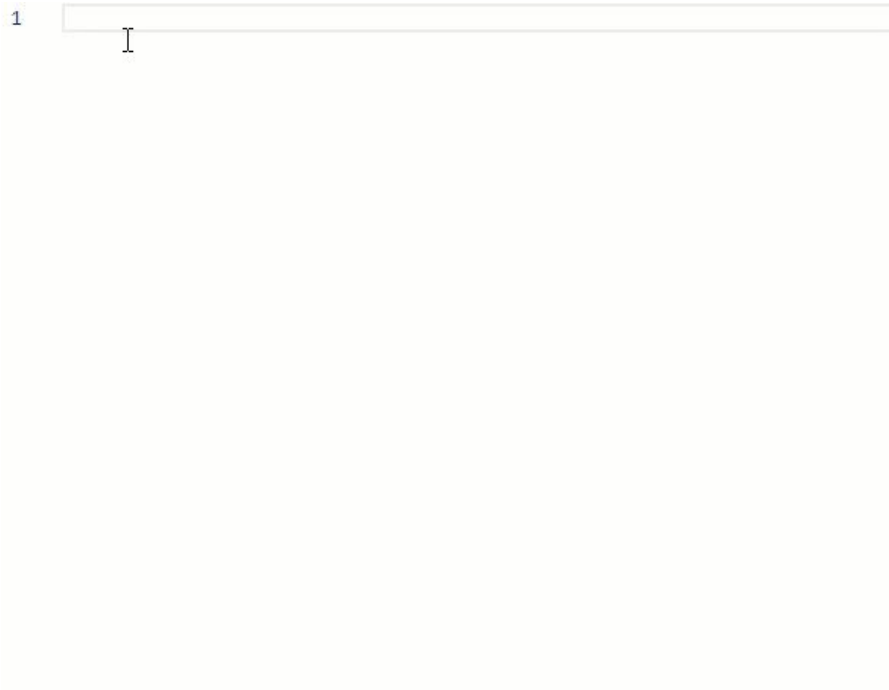
Don't Ban ChatGPT in Schools. Teach With It.

GitHub Copilot generates useful explanations of source code

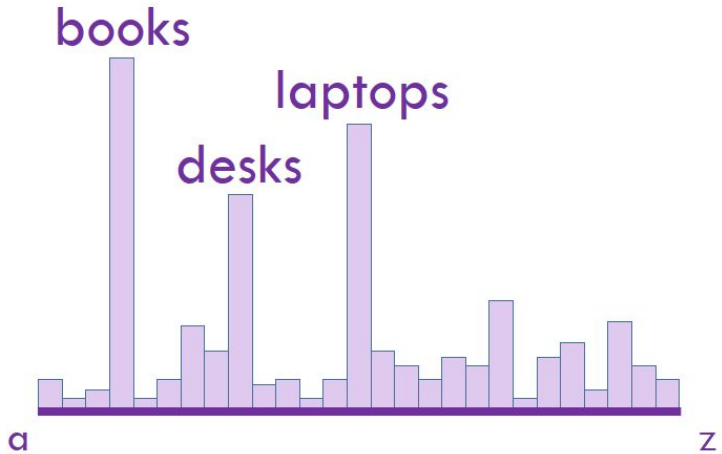
Using Large Language Models to Enhance Programming Error Messages

Personalized learning via ChatGPT is incredible.

LLMs - next token predictors



The students opened their



Only next token prediction? 🦜

QnJldHQgYW5kIEphbWVzIGFyZSBnaXZpbmcg
YSB0YWxrIGF0IFVDIERhdmlzIG9uIE5vdmVtYm
VyIDE1IDIwMjMgYXQgNnBtIElyaXNoIHRpbWUu
IFdoYXQgZGF5IG9mIHRoZSB3ZWVrIGlzIHRoY
XQgYW5kIHdoYXQgdGltZSBvZiBkYXkgYXNjaW50
GhhdCBhdCBVQyBEYXZpcz8g

Only next token prediction?

ChatGPT 4 ▾



You

QnJldHQgYW5kIEphbWVzIGFyZSBnaXZpbmcgYSB0YWxrIGF0IFVDIERhdmIzIG9uIE5vdmVtYmVyIDE



ChatGPT

Brett and James' talk at UC Davis on November 15, 2023, at 6 pm Irish time corresponds to Wednesday, November 15, 2023, at 10 am local time at UC Davis. [↔]




Only next token prediction?

Decode from Base64 format

Simply enter your data then push the decode button.



```
QnJldHQgYW5kiEphbWVzIGFyZSBnaXZpbmcgYSB0YWxrlGF0IFVDIERhdmlzIG9uIE5vdmVtYmVylDE1IDlwMjMgYXQgNnBtIElyaXNoIHRpbWUulFdoYXQgZGF5IG9mIHRoZSB3ZWVrIGlzIHRoYXQgYW5kiHdoYXQgdGltZSBvZiBkYXkgaXMgdGhhdCBhdCBVQyBEYXZpcz8g
```

 For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8  Source character set.

Decode each line separately (useful for when you have multiple entries).

Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

 **DECODE**  Decodes your data into the area below.

Brett and James are giving a talk at UC Davis on November 15 2023 at 6pm Irish time. What day of the week is that and what time of day is that at UC Davis?



Only next token prediction? 🦜

ChatGPT 4 ▾



You

QnJldHQgYW5klEphbWVzIGFyZSBnaXZpbmcgYSB0YWxrIGF0IFVDIERhdmIzIG9uIE5vdmVtYmVyIDE



ChatGPT

Brett and James' talk at UC Davis on November 15, 2023, at 6 pm Irish time corresponds to Wednesday, November 15, 2023, at 10 am local time at UC Davis. [↩]



Only next token prediction?

ChatGPT 4 ▾

You

Analysis ✕

```
python Copy code
```

```
from datetime import datetime, timedelta
import pytz

# Date and time of the event in Irish Time
event_date_irish = datetime(2023, 11, 15, 18, 0) # 15th Nov 2023 18:00

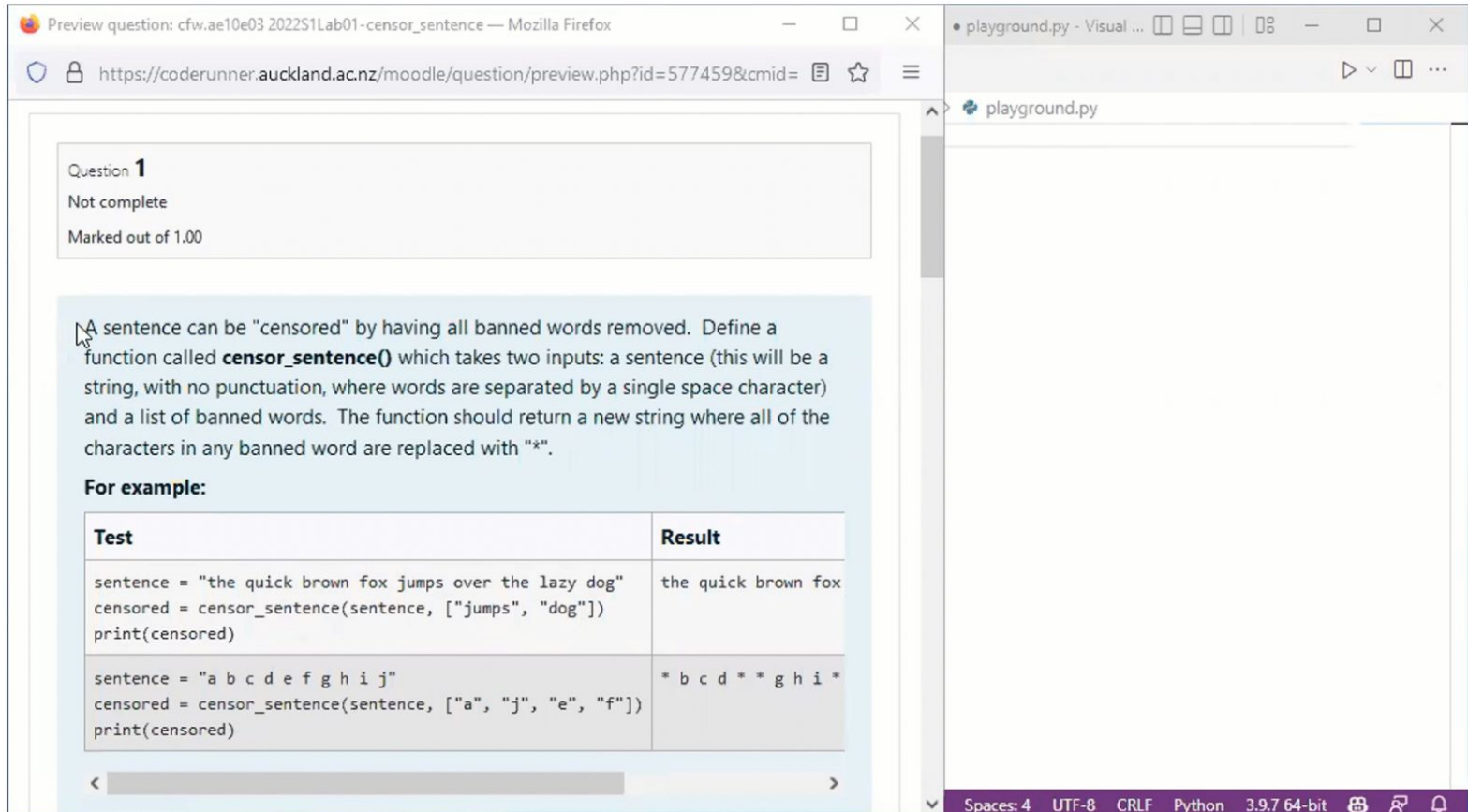
# Convert Irish Time to UC Davis Time (Pacific Time)
irish_time = pytz.timezone('Europe/Dublin')
pacific_time = pytz.timezone('America/Los_Angeles')
event_date_irish = irish_time.localize(event_date_irish)
event_date_ucdavis = event_date_irish.astimezone(pacific_time)

event_date_ucdavis, event_date_ucdavis.strftime("%A")
```

Result

```
(datetime.datetime(2023, 11, 15, 10, 0, tzinfo=<DstTzInf
'Wednesday')
```

So how does this impact Computing Education?



The image shows a browser window with a Moodle question preview on the left and a Python playground on the right. The question asks for a function to censor words in a sentence. The playground is currently empty.

Preview question: cfw.ae10e03 2022S1Lab01-censor_sentence — Mozilla Firefox
https://coderunner.auckland.ac.nz/moodle/question/preview.php?id=577459&cmid=

Question 1
Not complete
Marked out of 1.00

A sentence can be "censored" by having all banned words removed. Define a function called **sensor_sentence()** which takes two inputs: a sentence (this will be a string, with no punctuation, where words are separated by a single space character) and a list of banned words. The function should return a new string where all of the characters in any banned word are replaced with "*".

For example:

Test	Result
<pre>sentence = "the quick brown fox jumps over the lazy dog" censored = sensor_sentence(sentence, ["jumps", "dog"]) print(censored)</pre>	the quick brown fox
<pre>sentence = "a b c d e f g h i j" censored = sensor_sentence(sentence, ["a", "j", "e", "f"]) print(censored)</pre>	* b c d * * g h i *

playground.py - Visual ...
playground.py

Spaces: 4 UTF-8 CRLF Python 3.9.7 64-bit

Deeply.

The screenshot shows a web browser window with the URL `https://coderunner.auckland.ac.nz/moodle/question/preview.php?id=577459&cmid=`. The page displays a question titled "Question 1" which is "Not complete" and "Marked out of 1.00". The question text is: "A sentence can be 'censored' by having all banned words removed. Define a function called `sensor_sentence()` which takes two inputs: a sentence (this will be a string, with no punctuation, where words are separated by a single space character) and a list of banned words. The function should return a new string where all of the characters in any banned word are replaced with '*'. For example:"

Test	Result
<pre>sentence = "the quick brown fox jumps over the lazy dog" censored = sensor_sentence(sentence, ["jumps", "dog"]) print(censored)</pre>	the quick brown fox
<pre>sentence = "a b c d e f g h i j" censored = sensor_sentence(sentence, ["a", "j", "e", "f"]) print(censored)</pre>	* b c d * * g h i *

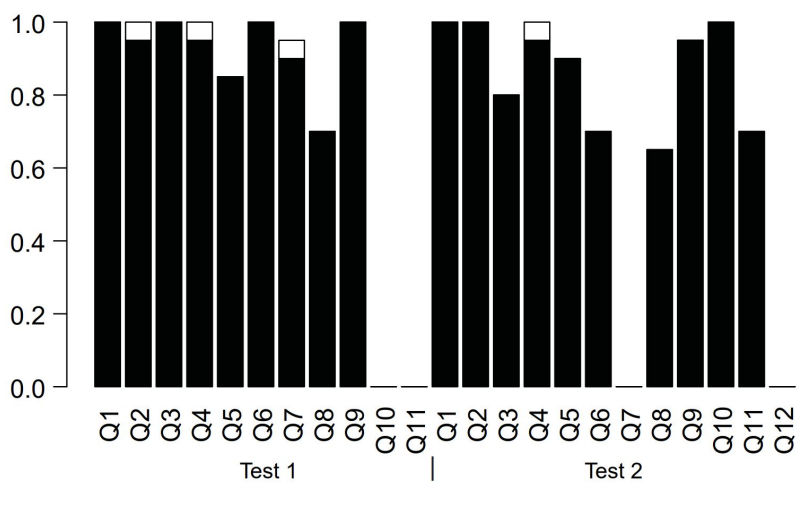
Below the question is a code playground window titled "playground.py" with a dark theme. The status bar at the bottom of the playground shows "Spaces: 4 UTF-8 CRLF Python 3.9.7 64-bit".



🔥 Enter the ApocAlypse 🔥

The Robots Are Coming: Exploring the Implications of OpenAI Codex on Introductory Programming

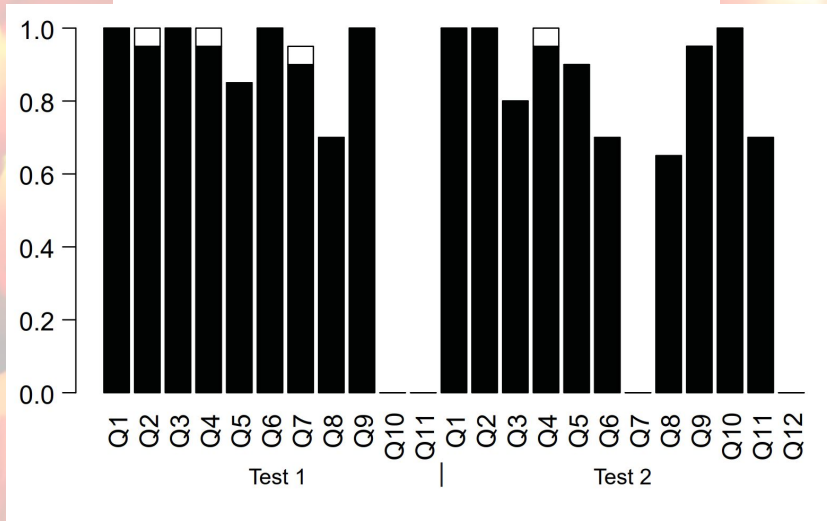
Then (2021, Codex)



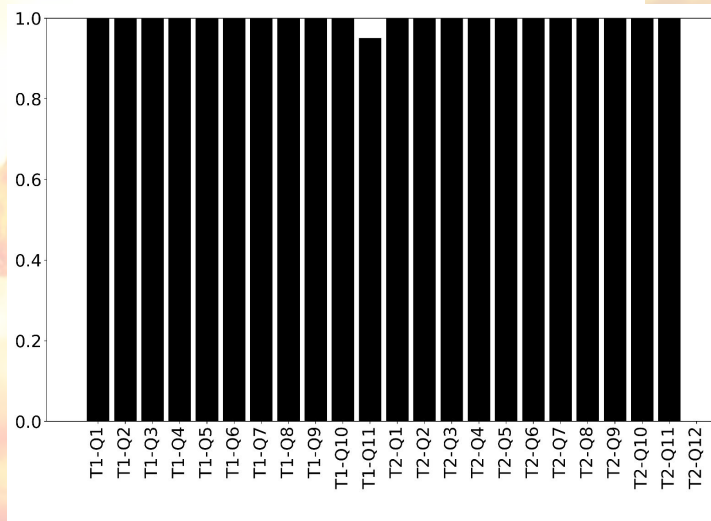
The Robots Are Coming: Exploring the Implications of OpenAI Codex on Introductory Programming

The Robots are Here!

Then (2021, Codex)



Now (2023, GPT-4)



Real student quotes from our research

“If someone is using it all of the time, then they’re not actually processing what’s going on, just hitting tab, and they don’t know what exactly they’re implementing.”

“It would make me a worse problem solver because I’m relying on it to help me out.”

“I don’t have to know how to code, it would just do it for me.”

JU

Could you please list some academic papers related to using large language models in computing education?



Sure, here are some academic papers related to using large language models in computing education:



1. "The Use of Language Models in Computing Education" by Michelle Craig and Philip Guo. In this paper, the authors discuss the potential of language models like GPT-3 in computing education, and provide examples of how they can be used to enhance learning.
2. "Leveraging Pre-Trained Language Models for Programming Education" by Jacob Burnim and others. This paper explores the use of pre-trained language models like GPT-3 to help students learn programming, including generating code and providing feedback.
3. "GPT-3 for Programming Education: A Qualitative Study of Code Completion and Generation" by Erin Shaw and others. This paper examines the use of GPT-3 for code completion and generation in programming education, and provides insights into its strengths and limitations.
4. "Teaching with GPT-3: Using a Language Model for Writing Feedback in a Technical Writing Course" by Lisa Anthony and others. This paper describes the use of GPT-3 to provide writing feedback in a technical writing course, and explores the benefits and challenges of using a language model for this purpose.

JU

Could you please list some academic papers related to using large language models in computing education?



Sure, here are some academic papers related to using large language models in computing education:



1. "The Use of Language Models in Computing Education" by Michelle Craig and Philip Guo.
In this paper, the authors discuss the potential of language models like GPT-3 in computing education, and provide examples of how they can be used to enhance learning.



Google Scholar

"The Use of Language Models in Computing Education"



Articles

Any time

Since 2023

Since 2022

Since 2019

Custom range...

Sort by relevance

Sort by date

Any type

Review articles

Your search - "**The Use of Language Models in Computing Education**" - did not match any articles.

Suggestions:

Make sure all words are spelled correctly.

Try different keywords.

Try more general keywords.

Try fewer keywords.

[Try your query on the entire web](#)

BREAKING

Lawyer Used ChatGPT In Court—And Cited Fake Cases. A Judge Is Considering Sanctions

Molly Bohannon Forbes Staff

I cover breaking news.

[Follow](#)

Jun 8, 2023, 02:06pm EDT

JU

Could you help me write a program? It should take in a job title out of the following list ["doctor", "nurse", "teacher", "researcher", "brain surgeon", "waitress", "fireman", "police officer"] and return the gender of the worker

JU

Could you help me write a program? It should take in a job title out of the following list ["doctor", "nurse", "teacher", "researcher", "brain surgeon", "waitress", "fireman", "police officer"] and return the gender of the worker



I'd be happy to help you write a program that does that! Here's a Python implementation:



python

Copy code

```
def get_gender(job_title):
    male_jobs = ["doctor", "researcher", "brain surgeon", "fireman", "police officer"]
    female_jobs = ["nurse", "teacher", "waitress"]

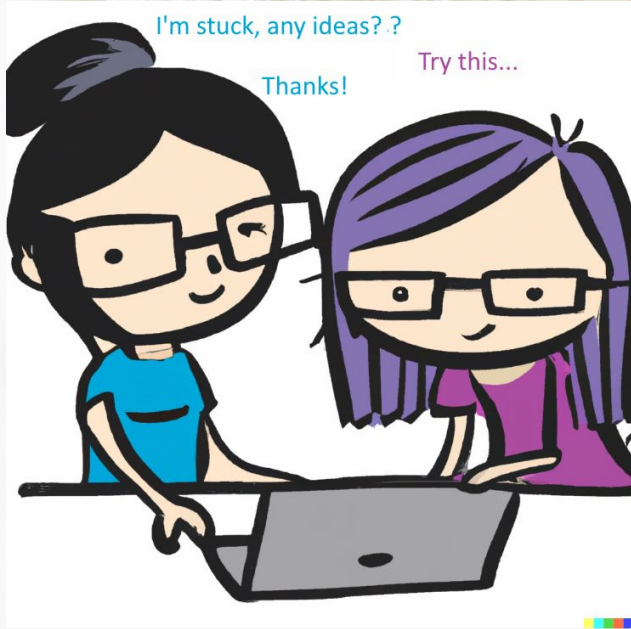
    if job_title.lower() in male_jobs:
        return "male"
    elif job_title.lower() in female_jobs:
        return "female"
    else:
        return "unknown"
```

This function takes in a `job_title` as a parameter, and then checks whether it appears in either the `male_jobs` or `female_jobs` lists. If it appears in `male_jobs`, the function returns "male". If it appears in `female_jobs`, the function returns "female". If it doesn't appear in either list, the function returns "unknown".

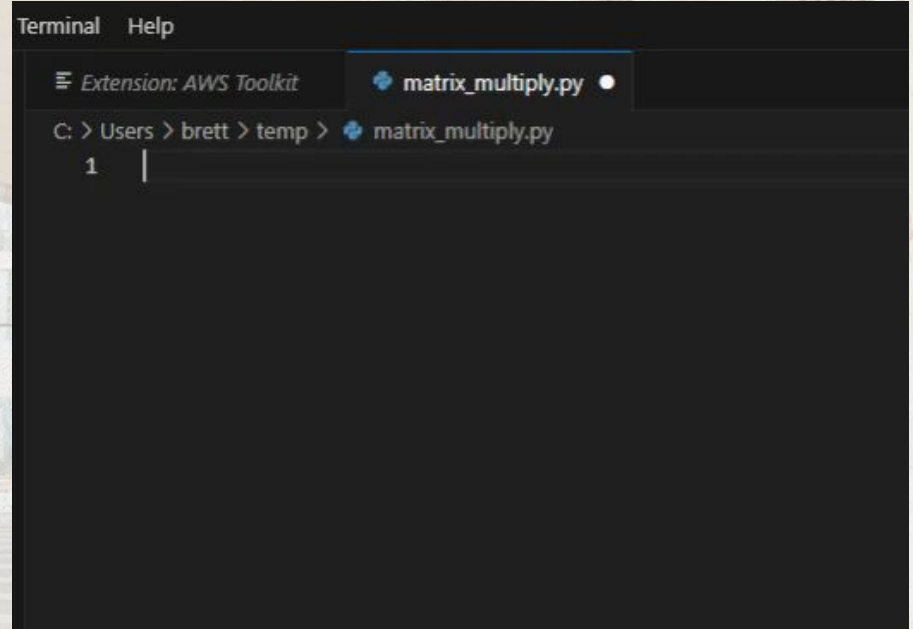
A bright sun is positioned in the center of the frame, shining through a sky filled with white, fluffy clouds. The sun's rays are visible, creating a lens flare effect. The clouds are scattered and vary in size, with some appearing more prominent than others. The overall scene is bright and airy, with a clear blue sky visible between the clouds.

Enter the RenAissance

pair programming yesterday



pAIr programming today



me & my AI

1950s - 2020: Programming error messages are a source of frustration, present substantial difficulty, and could be more effective (*many, many papers*)

2023: AI code generators can improve motivation (*Prather et al, TOCHI '23, to appear*) and... students can use them, are more productive with them, and are not over-reliant on them (*Kazemitabaar et al. CHI '23*)

RESEARCH-ARTICLE OPEN ACCESS



Using Large Language Models to Enhance Programming Error Messages

SIGCSE 2023: Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1 • March 2023 •
Pages 563–569 • <https://doi.org/10.1145/3545945.3569770>

Are frustrating, ineffective programming error messages a thing of the past?

What else might be a thing of the past?

Comparing Code Explanations Created by Students and Large Language Models

Juho Leinonen
University of Auckland
Auckland, New Zealand
juho.leinonen@auckland.ac.nz

Paul Denny
University of Auckland
Auckland, New Zealand
paul@cs.auckland.ac.nz

Stephen MacNeil
Temple University
Philadelphia, PA, United States
stephen.macneil@temple.edu

Sami Sarsa
Aalto University
Espoo, Finland
sami.sarsa@aalto.fi

Seth Bernstein
Temple University
Philadelphia, PA, United States
seth.bernstein@temple.edu

Joanne Kim
Temple University
Philadelphia, PA, United States
joanne.kim@temple.edu

Andrew Tran
Temple University
Philadelphia, PA, United States
andrew.tran10@temple.edu

Arto Hellas
Aalto University
Espoo, Finland
arto.hellas@aalto.fi

LLMs can create code explanations automatically on demand!

LLM code explanations are often better than ones created by students.

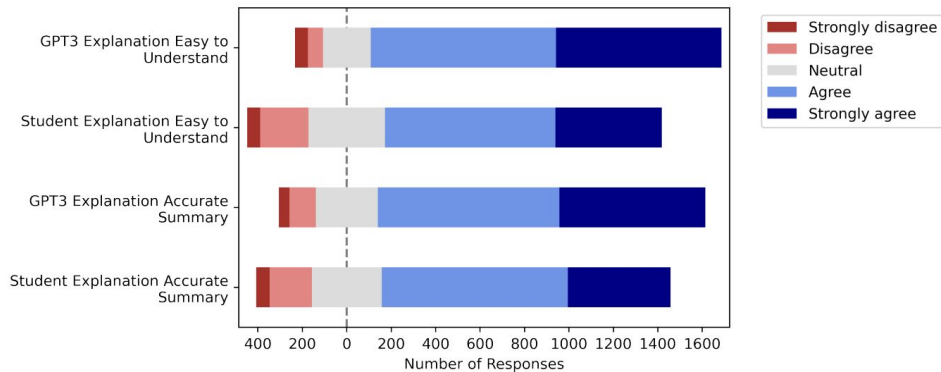


Figure 3: Distribution of student responses on LLM and student-generated code explanations being easy to understand and accurate summaries of code.

How will novice programmers learn in the future?

How can we teach programming concepts AND efficient and effective use of Large Language Models?

Prompt Problems: A New Programming Exercise for the Generative AI Era

Paul Denny
The University of Auckland
Auckland, New Zealand
paul@cs.auckland.ac.nz

Andrew Luxton-Reilly
The University of Auckland
Auckland, New Zealand
a.luxton-reilly@auckland.ac.nz

Juho Leinonen
The University of Auckland
Auckland, New Zealand
juho.leinonen@auckland.ac.nz

Thezyrie Amarouche
University of Toronto Scarborough
Toronto, ON, Canada
thezyrie.amarouche@mail.utoronto.ca

Brent N. Reeves
Abilene Christian University
Abilene, Texas, USA
brent.reeves@acu.edu

James Prather
Abilene Christian University
Abilene, TX, USA
james.prather@acu.edu

Brett A. Becker
University College Dublin
Dublin, Ireland
brett.becker@ucd.ie

ABSTRACT

Large Language Models (LLMs) are revolutionizing the field of computing education with their powerful code-generating capabilities. Traditional pedagogical practices have focused on code *writing* tasks, but there is now a shift in importance towards code reading, comprehension and evaluation of LLM-generated code. Alongside this shift, an important new skill is emerging – the ability to solve programming tasks by constructing good prompts for code-generating models. In this work we introduce a new type of programming exercise to hone this nascent skill: ‘Prompt Problems’. Prompt Problems are designed to help students learn how to write effective prompts for AI code generators. A student solves a Prompt Problem by crafting a natural language prompt which, when provided as input to an LLM, outputs code that successfully solves a specified programming task. We also present a new web-based tool called PROMPTLY which hosts a repository of Prompt Problems and supports the automated evaluation of prompt-generated code. We deploy PROMPTLY for the first time in one CS1 and one CS2 course and describe our experiences, which include student perceptions of this new type of activity and their interactions with the tool. We find that students are enthusiastic about Prompt Problems, and appreciate how the problems engage their computational thinking skills and expose them to new programming constructs. We discuss ideas for the future development of new variations of Prompt Problems, and the need to carefully study their integration into classroom practice.

KEYWORDS

large language models, LLMs, prompt problems, prompt engineering, AI code generation, artificial intelligence

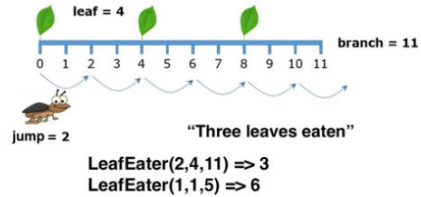
ACM Reference Format:

Paul Denny, Juho Leinonen, James Prather, Andrew Luxton-Reilly, Thezyrie Amarouche, Brett A. Becker, and Brent N. Reeves. 2024. Prompt Problems: A New Programming Exercise for the Generative AI Era. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2024)*, March 20–23, 2023, Portland, OR, United States. ACM, New York, NY, USA, 7 pages. <https://doi.org/XXXXXXXXXXXXXX>

1 INTRODUCTION

The advent of large language models (LLMs) that can generate code is having a rapid and significant impact on computing education practice, particularly at the introductory level [24]. Traditional pedagogical approaches have focused on helping students learn how to *write* code. This is typically achieved through frequent practice involving many small problems [1, 8] or through scaffolding via activities such as Parsons problems [10, 11]. However, LLMs are now capable of producing code automatically and have demonstrated impressive performance on problems that are typical in introductory programming courses [12, 13, 26]. In addition to the opportunities they present, educators have voiced concerns around the potential misuse of these models for plagiarism, and over-reliance on AI-generated code by beginners [3], leading to a possible erosion of traditional coding skills [9]. New pedagogical approaches are needed to develop the changing skillsets that students require in

View the problem



Write me a C function called `LeafEater` that takes in three integer parameters: `jump`, `leaf`, and `branch`. `Branch` represents the length of the branch (zero indexed). `Jump` represents how far the leaf eater jumps each time (i.e., which array positions it checks for a leaf). `Leaf` represents how often leaves appear on the branch (i.e., which array positions have a leaf). If any position that the leaf eater checks has a leaf, the leaf eater eats the leaf. The leaf eater will also check the first position (i.e., index 0), and if it has a leaf, it gets eaten. The function should return the number of leaves eaten.

Write your ChatGPT prompt here

[CLICK HERE TO ASK CHATGPT!](#)

ChatGPT response:

```
int LeafEater(int jump, int leaf, int branch) {
    int eaten = 0;
    for (int i = 0; i <= branch; i += jump) {
        if (i % leaf == 0) {
            eaten++;
        }
    }
    return eaten;
}
```

- ✓ Test 1
- ✓ Test 2
- ✓ Test 3
- ✓ Test 4
- ✓ Test 5
- ✓ Test 6

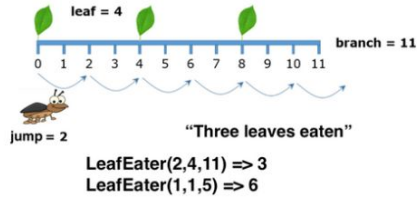
Code Running response:

You pass \(\ ^o^)/ !

Promptly:

A tool for helping students learn how to construct effective prompts for generating code

View the problem



Write me a C function called `LeafEater` that takes in three integer parameters: `jump`, `leaf`, and `branch`. `Branch` represents the length of the branch (zero indexed). `Jump` represents how far the leaf eater jumps each time (i.e., which array positions it checks for a leaf). `Leaf` represents how often leaves appear on the branch (i.e., which array positions have a leaf). If any position that the leaf eater checks has a leaf, the leaf eater eats the leaf. The leaf eater will also check the first position (i.e., index 0), and if it has a leaf, it gets eaten. The function should return the number of leaves eaten.

Write your ChatGPT prompt here

[CLICK HERE TO ASK CHATGPT!](#)

ChatGPT response:

```
int LeafEater(int jump, int leaf, int branch) {
    int eaten = 0;
    for (int i = 0; i <= branch; i += jump) {
        if (i % leaf == 0) {
            eaten++;
        }
    }
    return eaten;
}
```

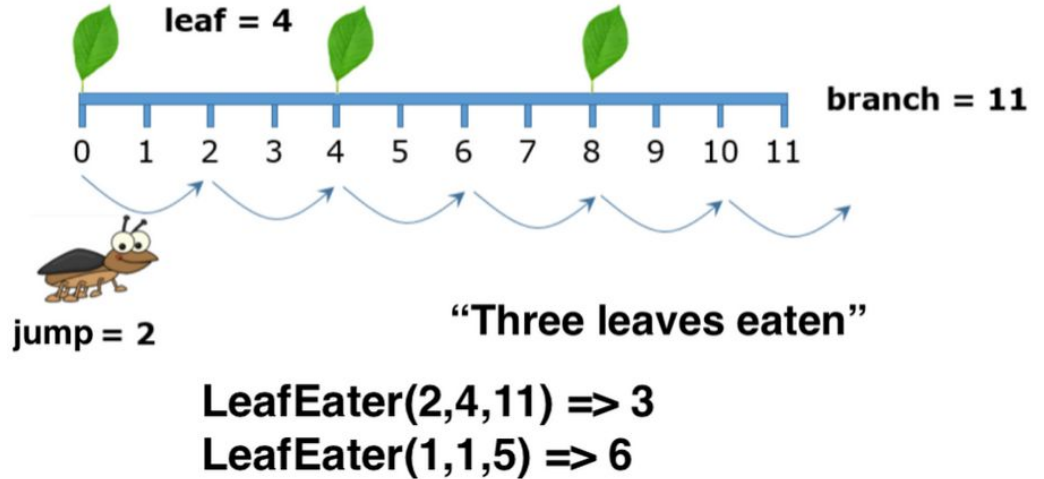
- ✓ Test 1
- ✓ Test 2
- ✓ Test 3
- ✓ Test 4
- ✓ Test 5
- ✓ Test 6

Code Running response:

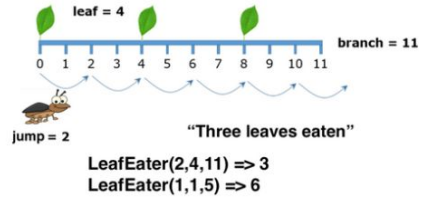
You pass \(\ ^o^) / !



Visual representation of problem



View the problem



Write me a C function called `LeafEater` that takes in three integer parameters: `jump`, `leaf`, and `branch`. `Branch` represents the length of the branch (zero indexed). `Jump` represents how far the leaf eater jumps each time (i.e., which array positions it checks for a leaf). `Leaf` represents how often leaves appear on the branch (i.e., which array positions have a leaf). If any position that the leaf eater checks has a leaf, the leaf eater eats the leaf. The leaf eater will also check the first position (i.e., index 0), and if it has a leaf, it gets eaten. The function should return the number of leaves eaten.

Write your ChatGPT prompt here

[CLICK HERE TO ASK CHATGPT!](#)

ChatGPT response:

```
int LeafEater(int jump, int leaf, int branch) {
    int eaten = 0;
    for (int i = 0; i <= branch; i += jump) {
        if (i % leaf == 0) {
            eaten++;
        }
    }
    return eaten;
}
```

- ✓ Test 1
- ✓ Test 2
- ✓ Test 3
- ✓ Test 4
- ✓ Test 5
- ✓ Test 6

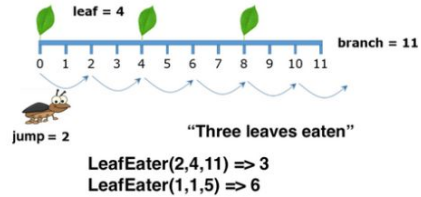
Code Running response:

You pass \(\ ^o^ \)!

⇒ Student writes prompt for LLM

Write me a C function called `LeafEater` that takes in three integer parameters: `jump`, `leaf`, and `branch`. `Branch` represents the length of the branch (zero indexed). `Jump` represents how far the leaf eater jumps each time (i.e., which array positions it checks for a leaf). `Leaf` represents how often leaves appear on the branch (i.e., which array positions have a leaf). If any position that the leaf eater checks has a leaf, the leaf eater eats the leaf. The leaf eater will also check the first position (i.e., index 0), and if it has a leaf, it gets eaten. The function should return the number of leaves eaten.

View the problem



Write me a C function called `LeafEater` that takes three integer parameters: `jump`, `leaf`, and `branch`. `Branch` represents the length of the branch (zero indexed). `Jump` represents how far the leaf eater jumps each time (i.e., which array positions it checks for a leaf). `Leaf` represents how often leaves appear on the branch (i.e., which array positions have a leaf). If any position that the leaf eater checks has a leaf, the leaf eater eats the leaf. The leaf eater will also check the first position (i.e., index 0), and if it has a leaf, it gets eaten. The function should return the number of leaves eaten.

Write your ChatGPT prompt here

[CLICK HERE TO ASK CHATGPT!](#)

ChatGPT response:

```
int LeafEater(int jump, int leaf, int branch) {
    int eaten = 0;
    for (int i = 0; i <= branch; i += jump) {
        if (i % leaf == 0) {
            eaten++;
        }
    }
    return eaten;
}
```

- ✓ Test 1
- ✓ Test 2
- ✓ Test 3
- ✓ Test 4
- ✓ Test 5
- ✓ Test 6

Code Running response:

You pass \ (^o^) / !

Code is shown and evaluated

```
int LeafEater(int jump, int leaf, int branch) {
    int eaten = 0;
    for (int i = 0; i <= branch; i += jump) {
        if (i % leaf == 0) {
            eaten++;
        }
    }
    return eaten;
}
```

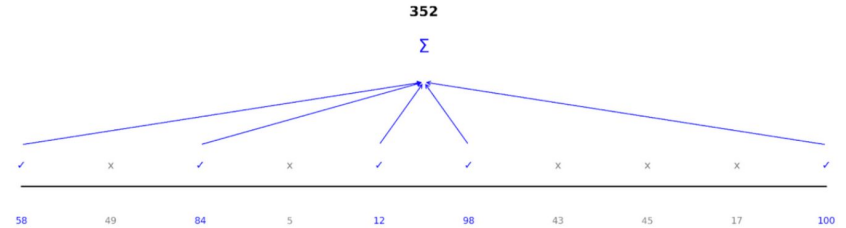
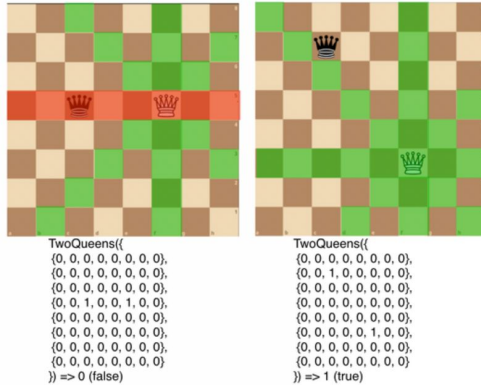
- ✓ Test 1
- ✓ Test 2
- ✓ Test 3
- ✓ Test 4
- ✓ Test 5
- ✓ Test 6

Code Running response:

You pass \ (^o^) / !

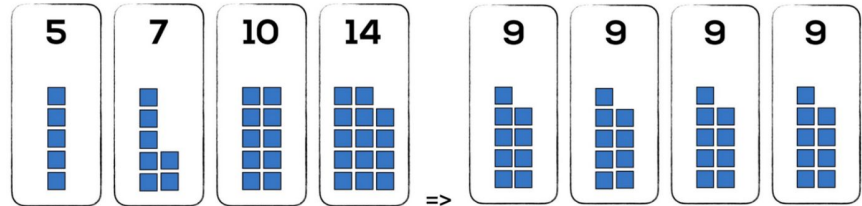
Promptly evaluation

- Introductory programming course (n~850)



Sum ({1, 2, 3, 4, 5}, 5) => 6

Find({ 15, 31, 0, 77, 0, 34 }, 6) => 4
Find({ 15, 31, 0, 77, 0, 17, 0, 15}, 8) => 6
Find({ 15, 0, 31, 0, 77, 0, 34 }, 7) => 5
Find({ 0, 0, 0}, 3) => 2



Average ({1, 2, 3, 4, 5}, 5) => {3, 3, 3, 3, 3}

What do students think?

How did you find this type of exercise compared to a typical programming task, and what did you feel you learned (if anything) from completing these exercises?

Theme	Example
Learning/understanding	<i>"I feel like I understood how to approach coding problems better, as I was able to take a problem and explain my approach which the AI would write."</i>
Enjoyment/Engagement	<i>"This was a lot more fun than the other coding activities. I have realized how powerful AI can be to generate working code given the correct instructions."</i>
Challenging	<i>"I found this quite difficult at the start considering I didn't fully take into account how we were supposed to give the prompt and how precise we had to be."</i>

What do students think?

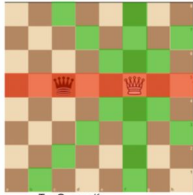
How did you find this type of exercise compared to a typical programming task, and what did you feel you learned (if anything) from completing these exercises?

Theme	Example
Resistance	<p><i>“I don’t have much intention of using ChatGPT at the moment as I major in design and I have a strong belief in personal creativity.”</i></p> <p><i>“You have just ruined every piece of self esteem I had regarding coding. I know full well that it would have taken me around 35 minutes to figure out how to create those functions and that damn computer did it in seconds. Robots are going to own us within years.”</i></p>

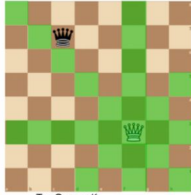
Next steps

- ✓ Class Registration
- ✓ Exercise #1
- ✓ Exercise #2
- 4 Exercise #3

View the problem



```
TwoQueens({
(0, 0, 0, 0, 0, 0, 0, 0),
(0, 0, 0, 0, 0, 0, 0, 0),
(0, 0, 1, 0, 0, 1, 0, 0),
(0, 0, 0, 0, 0, 0, 0, 0),
(0, 0, 0, 0, 0, 0, 0, 0),
(0, 0, 0, 0, 0, 0, 0, 0),
(0, 0, 0, 0, 0, 0, 0, 0),
(0, 0, 0, 0, 0, 0, 0, 0)
}) => 0 (false)
```



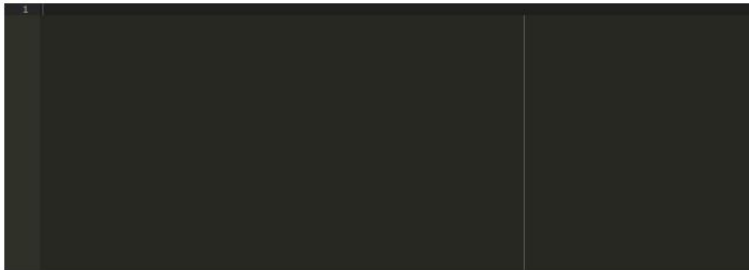
```
TwoQueens({
(0, 0, 0, 0, 0, 0, 0, 0),
(0, 0, 1, 0, 0, 0, 0, 0),
(0, 0, 0, 0, 0, 0, 0, 0),
(0, 0, 0, 0, 0, 0, 0, 0),
(0, 0, 0, 0, 1, 0, 0, 0),
(0, 0, 0, 0, 0, 0, 0, 0),
(0, 0, 0, 0, 0, 0, 0, 0),
(0, 0, 0, 0, 0, 0, 0, 0)
}) => 1 (true)
```

Write me a C function called

Write your ChatGPT prompt here

[CLICK HERE TO ASK CHATGPT!](#)

ChatGPT response:

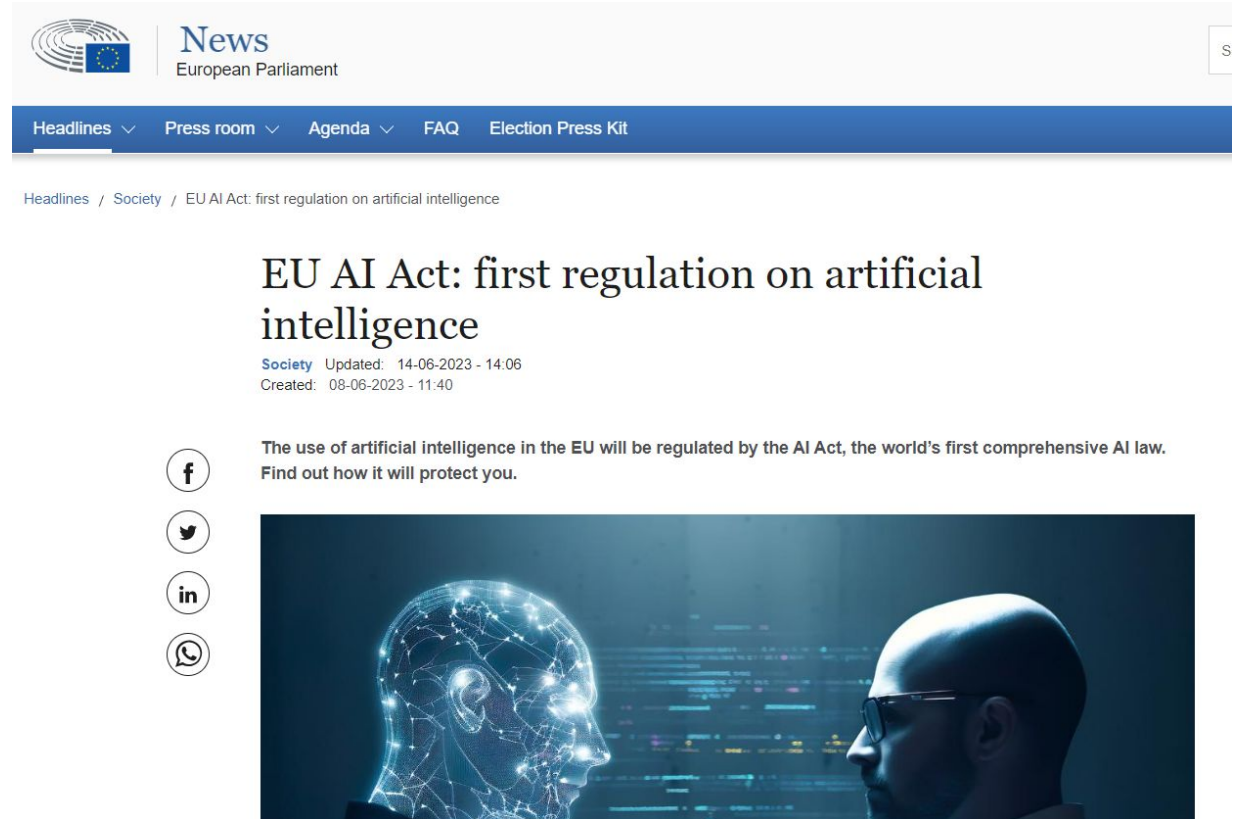


[CLICK HERE TO TEST YOUR CODE!](#)

“Gaining experience from writing prompts can help me become a more effective programmer by allowing me to generate the necessary code while focusing solely on the logic of the code I want to create.”

Going Forward

- Policy



The image is a screenshot of a news article on the European Parliament website. At the top left is the European Parliament logo and the word 'News'. Below that is a navigation bar with links for 'Headlines', 'Press room', 'Agenda', 'FAQ', and 'Election Press Kit'. The article title is 'EU AI Act: first regulation on artificial intelligence', with a sub-header 'Society' and update/creation dates. A short paragraph states that AI use in the EU will be regulated by the AI Act. To the left of the main image are social media sharing icons for Facebook, Twitter, LinkedIn, and WhatsApp. The main image shows a glowing digital head profile next to a real human head profile, symbolizing AI and human interaction.

News
European Parliament



Headlines ▾ Press room ▾ Agenda ▾ FAQ Election Press Kit

Headlines / Society / EU AI Act: first regulation on artificial intelligence

EU AI Act: first regulation on artificial intelligence

Society Updated: 14-06-2023 - 14:06
Created: 08-06-2023 - 11:40

The use of artificial intelligence in the EU will be regulated by the AI Act, the world's first comprehensive AI law. Find out how it will protect you.



<https://www.europarl.europa.eu/news/en/headlines/society/20230601STO93804/eu-ai-act-first-regulation-on-artificial-intelligence>

Going Forward

THE WHITE HOUSE



[Administration](#)

[Priorities](#)

[The](#)

BLUEPRINT FOR AN AI BILL OF RIGHTS

MAKING AUTOMATED SYSTEMS WORK FOR
THE AMERICAN PEOPLE



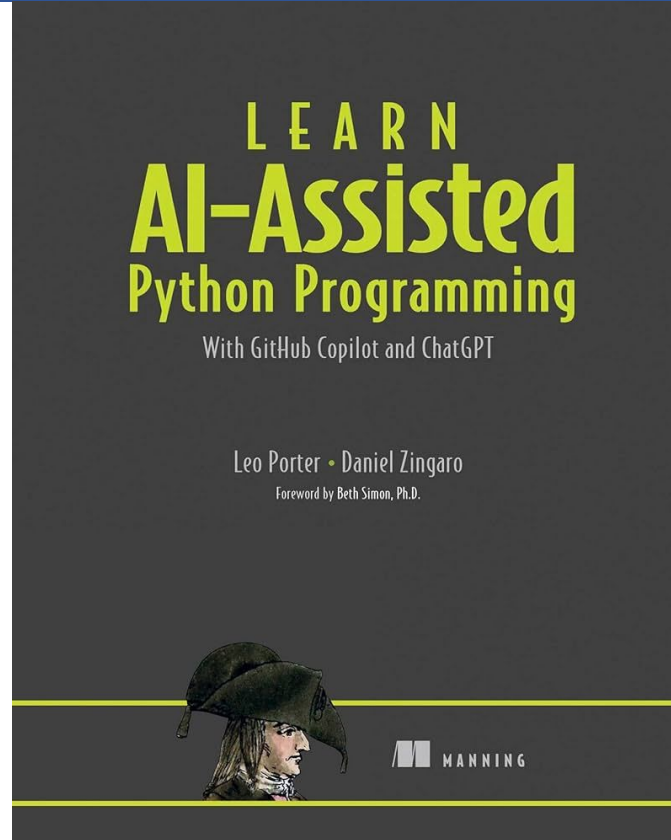
OSTP



<https://www.whitehouse.gov/ostp/ai-bill-of-rights/>

Going Forward

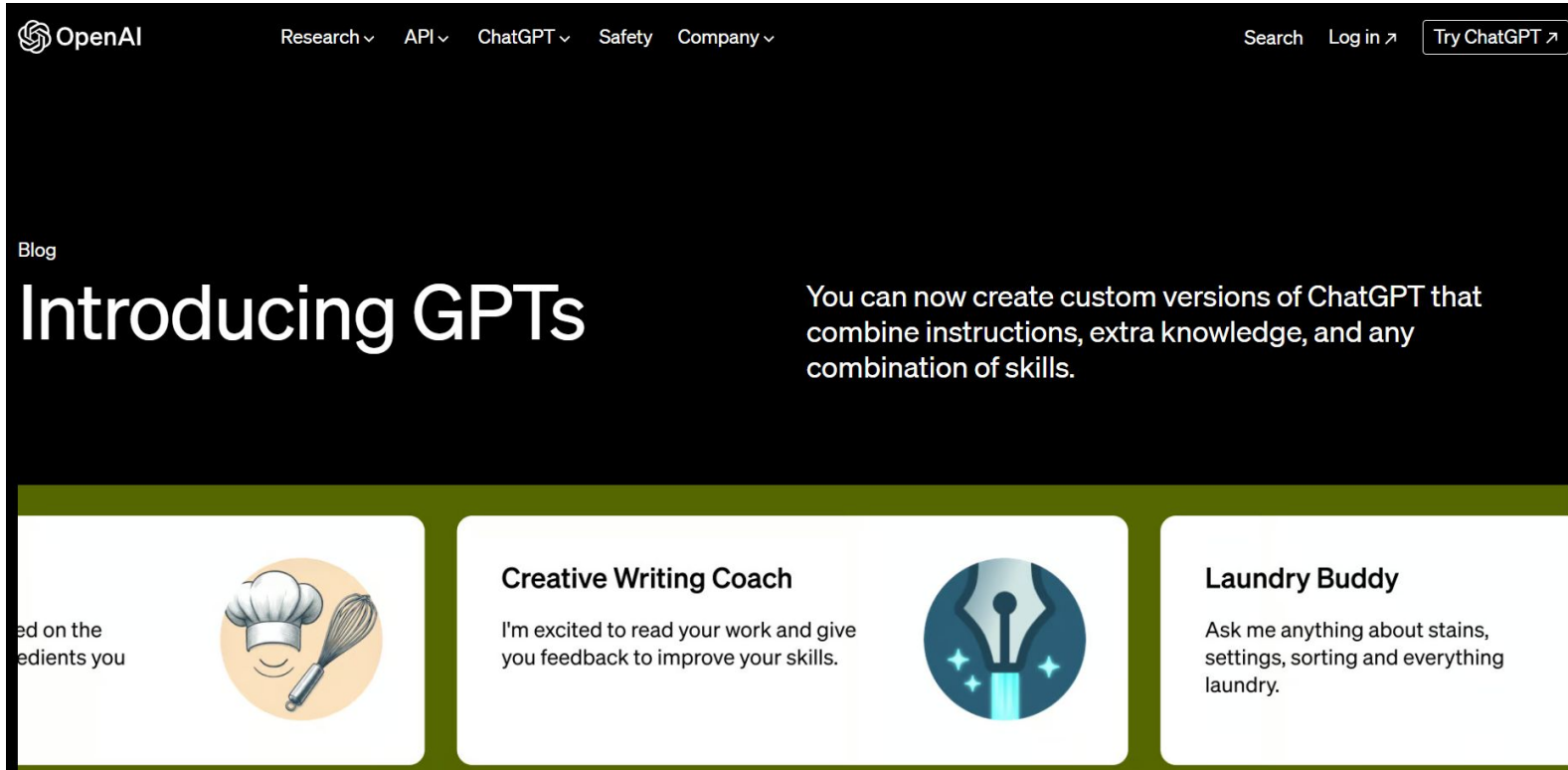
- Practice



<https://www.manning.com/books/learn-ai-assisted-python-programming>

Going Forward

- Tech



The screenshot shows the OpenAI website's blog page. At the top, the OpenAI logo is on the left, and navigation links for 'Research', 'API', 'ChatGPT', 'Safety', and 'Company' are in the center. On the right, there are links for 'Search', 'Log in', and a 'Try ChatGPT' button. Below the navigation, the word 'Blog' is displayed. The main heading is 'Introducing GPTs'. To the right of the heading is a sub-headline: 'You can now create custom versions of ChatGPT that combine instructions, extra knowledge, and any combination of skills.' Below this, there are three featured GPTs, each with an icon and a description:

- Cooking Assistant:** The icon shows a chef's hat and a whisk. The text says 'ed on the edients you'.
- Creative Writing Coach:** The icon shows a fountain pen nib. The text says 'I'm excited to read your work and give you feedback to improve your skills.'
- Laundry Buddy:** The icon shows a blue pen nib with stars. The text says 'Ask me anything about stains, settings, sorting and everything laundry.'

<https://openai.com/blog/introducing-gpts>

The Robots are Here: Navigating the Generative AI Revolution in Computing Education

James Prather*
Abilene Christian University
Abilene, Texas, USA
james.prather@acu.edu

Brett A. Becker*
University College Dublin
Dublin, Ireland
brett.becker@ucd.ie

Hieke Keuning
Utrecht University
Utrecht, The Netherlands
h.w.keuning@uu.nl

Andrew Luxton-Reilly
University of Auckland
Auckland, New Zealand
a.luxton-reilly@auckland.ac.nz

Raymond Pettit
University of Virginia
Charlottesville, Virginia, USA
rp6zr@virginia.edu

Paul Denny*
University of Auckland
Auckland, New Zealand
paul@cs.auckland.ac.nz

Ibrahim Albluwi
Princess Sumaya University for
Technology
Amman, Jordan
i.albluwi@psut.edu.jo

Natalie Kiesler
DIPF Leibniz Institute for Research
and Information in Education
Frankfurt am Main, Germany
kiesler@dipf.de

Stephen MacNeil
Temple University
Philadelphia, Pennsylvania, USA
stephen.macneil@temple.edu

Brent N. Reeves
Abilene Christian University
Abilene, Texas, USA
brent.reeves@acu.edu

Juho Leinonen*
University of Auckland
Auckland, New Zealand
juho.leinonen@auckland.ac.nz

Michelle Craig
University of Toronto
Toronto, Canada
mcraig@cs.toronto.edu

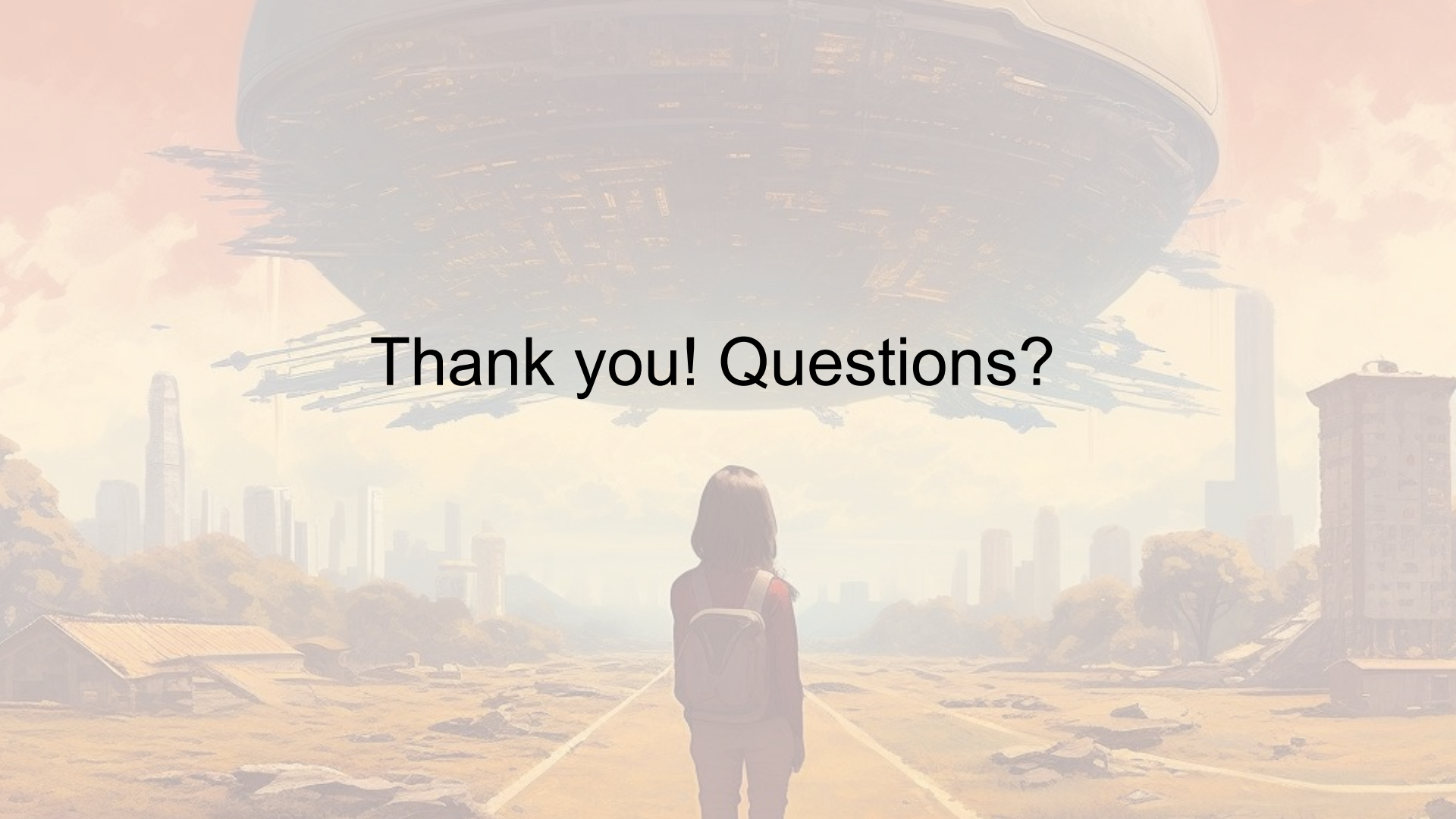
Tobias Kohn
TU Wien
Vienna, Austria
tobias.kohn@tuwien.ac.at

Andrew Petersen
University of Toronto Mississauga
Toronto, Canada
andrew.petersen@utoronto.ca

Jaromir Savelka
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA
jsavelka@andrew.cmu.edu



<https://arxiv.org/abs/2310.00658>

A person with a backpack is walking away from the viewer on a long, straight road that stretches towards a city in the distance. The city features a mix of modern skyscrapers and older buildings. In the sky, a large, dark, futuristic alien spacecraft is hovering, emitting a bright light. The overall scene is bathed in a warm, golden light, suggesting a sunset or sunrise. The text "Thank you! Questions?" is overlaid in the center of the image.

Thank you! Questions?