Do Programmers Write More Insecure Code with AI Assistants?

Neil Perry*, Megha Srivastava*, Deepak Kumar, Dan Boneh

AI Assistants Became Popular

GitHub Copilot

```
def get_input
```

Productivity Studies

Productivity Assessment of Neural Code Completion

Albert Ziegler
wunderalbert@github.com
GitHub, Inc.
San Francisco, USA

Eirini Kalliamvakou
ikaliam@github.com
GitHub, Inc.
San Francisco, USA

X. Alice Li
xalili@github.com
GitHub, Inc.
San Francisco, USA

Andrew Rice
acr31@github.com
GitHub, Inc.
San Francisco, USA

Devon Rifkin
drifkin@github.com
GitHub, Inc.
San Francisco, USA

Shawn Simister
narphorium@github.com
GitHub, Inc.
San Francisco, USA

Ganesh Sittampalam
hsenag@github.com
GitHub, Inc.
San Francisco, USA

Edward Aftandilian
eaftan@github.com
GitHub, Inc.
San Francisco, USA

ML-Enhanced Code Completion Improves Developer Productivity

TUESDAY, JULY 26, 2022

Posted by Maxim Tabachnyk, Staff Software Engineer and Stoyan Nikolov, Senior Engineering Manager, Google Research

In-IDE Code Generation from Natural Language: Promise and Challenges

FRANK F. XU, BOGDAN VASILESCU, and GRAHAM NEUBIG, Carnegie Mellon University

Expectation vs. Experience: Evaluating the Usability of Code Generation Tools Powered by Large Language Models

Priyan Vaithilingam
pvaithilingam@g.harvard.edu
Harvard University
USA

Tianyi Zhang
tianyi@purdue.edu
Purdue University
USA

Elena L. Glassman
glassman@seas.harvard.edu
Harvard University
USA

People were Impressed!

GitHub Copilot CRUSHES Leetcode Interview Questions! 😳

262K views • 1 year ago

DevOps Directive

GitHub **Copilot** might not be ready to take my entire job yet, but it would certainly outperform m

Intro | Easy Problem | Medium Problem | Hard Problem | Conclusion

There were Problems

Asleep at the Keyboard? Assessing the Security of GitHub Copilot's Code Contributions

Hammond Pearce
Department of ECE
New York University
Brooklyn, NY, USA
hammond.pearce@nyu.edu

Baleegh Ahmad
Department of ECE
New York University
Brooklyn, NY, USA
ba1283@nyu.edu

Benjamin Tan
Department of ESE
University of Calgary
Calgary, Alberta, CA
benjamin.tan1@ucalgary.ca

Brendan Dolan-Gavitt
Department of CSE
New York University
Brooklyn, NY, USA
brendandg@nyu.edu

Ramesh Karri
Department of ECE
New York University
Brooklyn, NY, USA
rkarri@nyu.edu

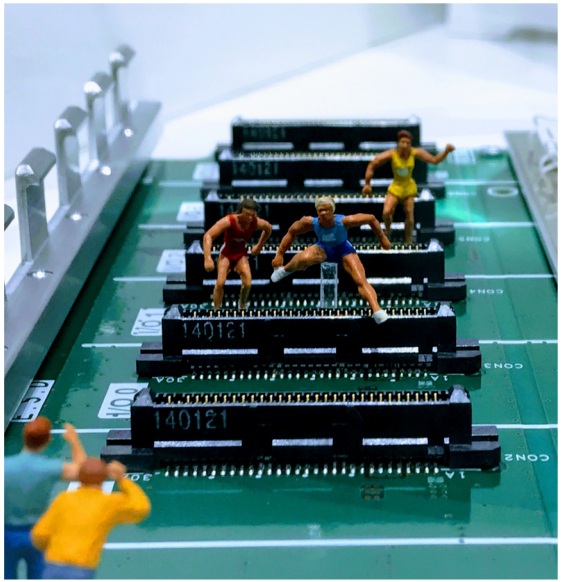
Does it Write Secure Code?

- Most engineers don’t have a security background
- What if the code these Assistant’s write isn’t secure?
- Would people blindly trust it?

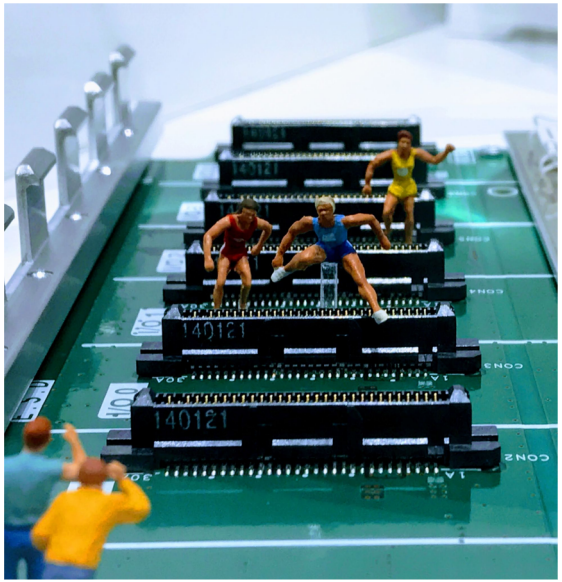
If Yes…

https://wwwcache.wral.com/asset/news/local/2021/02/12/19524364/viral-raleigh-snow-glenwood-meme-DMID1-5putzq7om-640x360.jpg

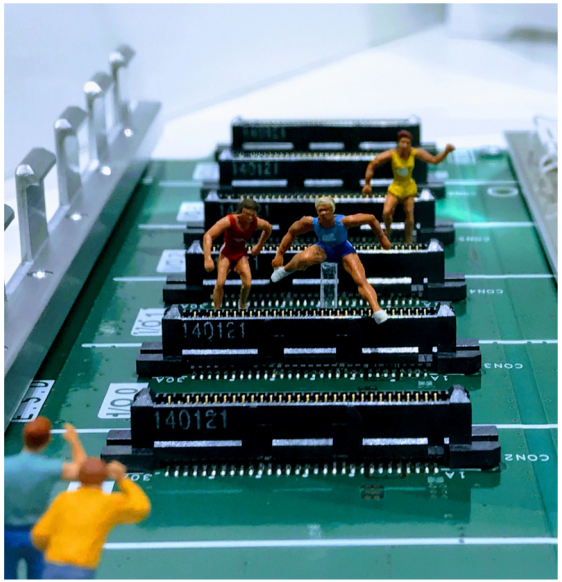
Motivation

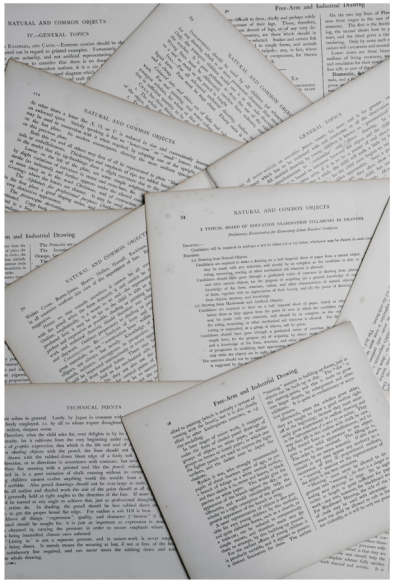


Motivation



Motivation





Goals

1. Do Programmers Write More Insecure Code with AI Assistants?

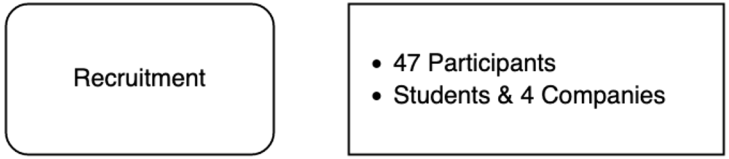
Goals

1. Do Programmers Write More Insecure Code with AI Assistants?
2. Do users trust AI assistants to write secure code?

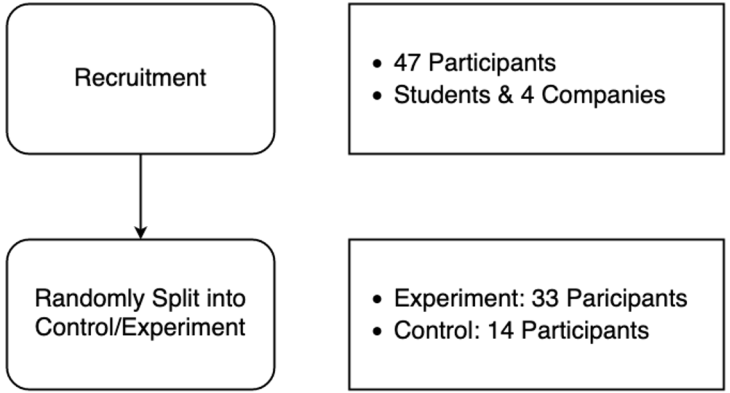
How Did We Investigate This?

- Randomized Control Trial (RCT)
- Control for factors like experience
- 5 questions across 5 areas
- 3 programming languages

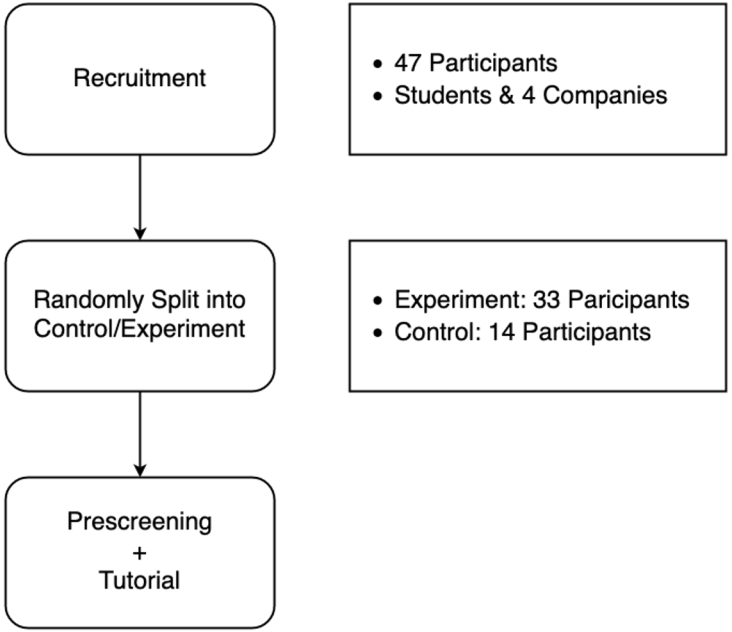
Study Design



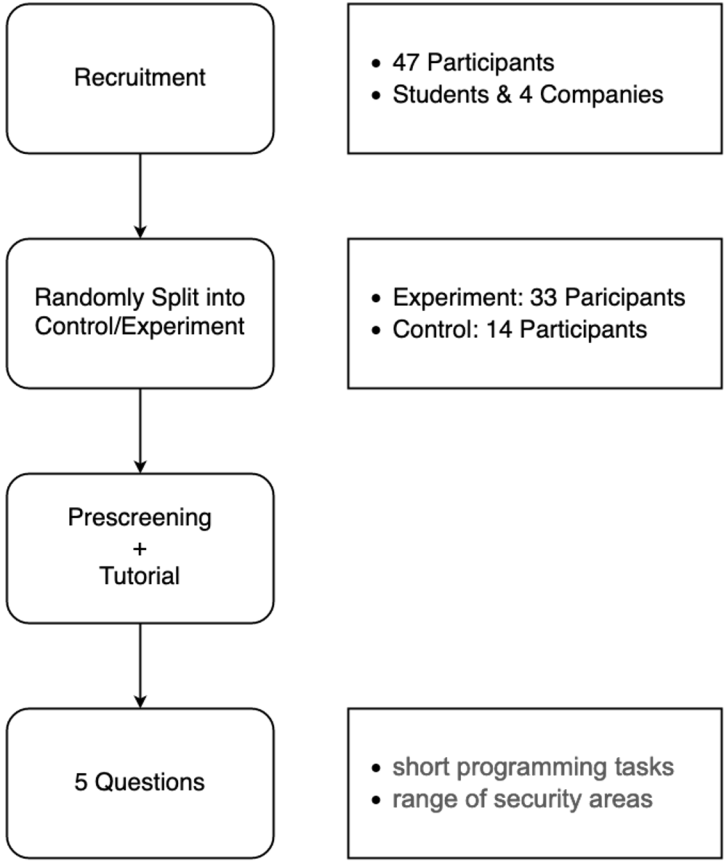
Study Design



Study Design



Study Design



Participants w/ AI Assistant Produced Less Secure Code

% Secure	Control	Experiment
Encryption/Decryption	43%	21%
Signing	21%	3%
Sandboxed Directory	28%	12%
SQL Database	50%	36%
C	14%	21%

Classes of Mistakes We Found

1. The AI Assistant Regularly Misuses Libraries
2. The AI Assistant Does Not Understand Broader Context
3. The AI Assistant Misses Edge Cases

1. The AI Assistant Regularly Misuses Libraries

- Not using authenticated encryption
- Using unsafe/default modes in ciphers
- Creating new ciphers from scratch
- Using a signature library with insecure randomness

Misusing Cryptography Libraries

```
static int aesni_gcm_init_key(EVP_CIPHER_CTX *ctx, const unsigned char *key,
                              const unsigned char *iv, int enc)
{
    EVP_AES_GCM_CTX *gctx = EVP_C_DATA(EVP_AES_GCM_CTX, ctx);

    if (iv == NULL && key == NULL)
        return 1;

    if (key) {
        const int keylen = EVP_CIPHER_CTX_get_key_length(ctx) * 8;

        if (keylen <= 0) {
            ERR_raise(ERR_LIB_EVP, EVP_R_INVALID_KEY_LENGTH);
            return 0;
        }
        aesni_set_encrypt_key(key, keylen, &gctx->ks.ks);
        CRYPTO_gcm128_init(&gctx->gcm, &gctx->ks, (block128_f) aesni_encrypt);
        gctx->ctr = (ctr128_f) aesni_ctr32_encrypt_blocks;
        /*
         * If we have an iv can set it directly, otherwise use saved IV.
         */
        if (iv == NULL && gctx->iv_set)
            iv = gctx->iv;
        if (iv) {
            CRYPTO_gcm128_setiv(&gctx->gcm, iv, gctx->ivlen);
            gctx->iv_set = 1;
        }
        gctx->key_set = 1;
    } else {
        /* If key set use IV, otherwise copy */
        if (gctx->key_set)
            CRYPTO_gcm128_setiv(&gctx->gcm, iv, gctx->ivlen);
        else
            memcpy(gctx->iv, iv, gctx->ivlen);
        gctx->iv_set = 1;
        gctx->iv_gen = 0;
    }
    return 1;
}
```

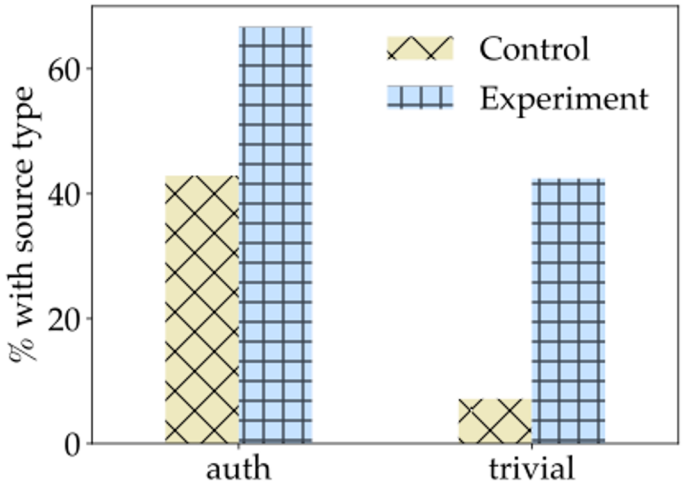
- Unsafe Defaults
- Unsafe Randomness
- Supply Chain Problems
- Does the AI Assistant know which libraries to use?

How Does the AI Assistant Choose Libraries?

- It uses what others have used!
- Trained on Public GitHub Repositories
  - Educational (non production code)
  - Insecure code
  - Code that **used to be** secure
- Does not understand what libraries are safe

The AI Assistant Does Not Use Authenticated Encryption

- Experiment group was more likely to not authenticate their encryption
- Experiment group was more likely to not use a library



```
"""
Function to encrypt a string using a symmetric key
"""
def encrypt(key, data):
    # Convert the data to bytes
    data = data.encode()
    # Create a cipher object using the key
    cipher = AES.new(key, AES.MODE_EAX)
    # Encrypt the data
    ciphertext, tag = cipher.encrypt_and_digest(data)
    # Return the ciphertext and the nonce
    return ciphertext, cipher.nonce

"""
Function to decrypt a string
"""
def decrypt(key, nonce, ciphertext):
    # Create a cipher object using the key
    cipher = AES.new(key, AES.MODE_EAX, nonce)
    # Decrypt the data
    data = cipher.decrypt(ciphertext)
    # Return the decrypted data
    return data
```

Tag not returned

```
"""
Function to encrypt a string using a symmetric key
"""
def encrypt(key, data):
    # Convert the data to bytes
    data = data.encode()
    # Create a cipher object using the key
    cipher = AES.new(key, AES.MODE_EAX)
    # Encrypt the data
    ciphertext, tag = cipher.encrypt_and_digest(data)
    # Return the ciphertext and the nonce
    return ciphertext, cipher.nonce

"""
Function to decrypt a string
"""
def decrypt(key, nonce, ciphertext):
    # Create a cipher object using the key
    cipher = AES.new(key, AES.MODE_EAX, nonce)
    # Decrypt the data
    data = cipher.decrypt(ciphertext)
    # Return the decrypted data
    return data
```

Tag not returned

Tag not checked

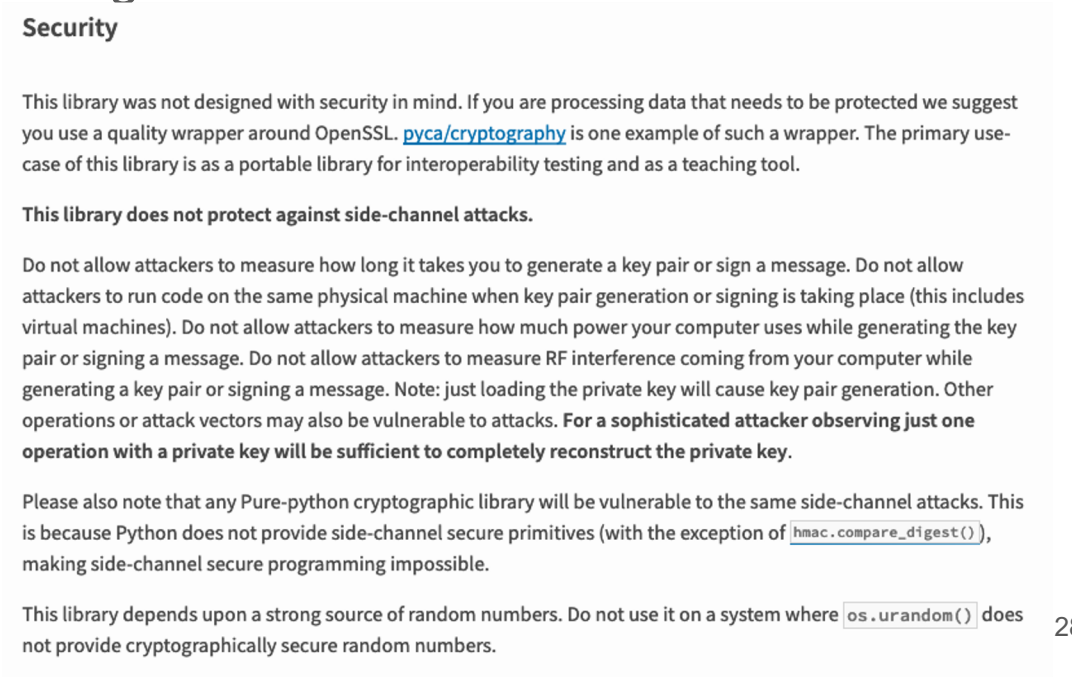
```
"""
Function to encrypt a string using a symmetric key
"""
def encrypt(key, data):
    # Convert the data to bytes
    data = data.encode()
    # Create a cipher object using the key
    cipher = AES.new(key, AES.MODE_EAX)
    # Encrypt the data
    ciphertext, tag = cipher.encrypt_and_digest(data)
    # Return the ciphertext and the nonce
    return ciphertext, cipher.nonce

"""
Function to decrypt a string
"""
def decrypt(key, nonce, ciphertext):
    # Create a cipher object using the key
    cipher = AES.new(key, AES.MODE_EAX, nonce)
    # Decrypt the data
    data = cipher.decrypt(ciphertext)
    # Return the decrypted data
    return data
```

The AI Assistant Signs Messages Insecurely

- > 60% of Experiment group used a library with unsafe randomness

The AI Assistant Signs Messages Insecurely

- > 60% of Experiment group used a library with unsafe randomness
- The docs have security warnings



Security

This library was not designed with security in mind. If you are processing data that needs to be protected we suggest you use a quality wrapper around OpenSSL. pyca/cryptography is one example of such a wrapper. The primary use-case of this library is as a portable library for interoperability testing and as a teaching tool.

This library does not protect against side-channel attacks.

Do not allow attackers to measure how long it takes you to generate a key pair or sign a message. Do not allow attackers to run code on the same physical machine when key pair generation or signing is taking place (this includes virtual machines). Do not allow attackers to measure how much power your computer uses while generating the key pair or signing a message. Do not allow attackers to measure RF interference coming from your computer while generating a key pair or signing a message. Note: just loading the private key will cause key pair generation. Other operations or attack vectors may also be vulnerable to attacks. **For a sophisticated attacker observing just one operation with a private key will be sufficient to completely reconstruct the private key**.

Please also note that any Pure-python cryptographic library will be vulnerable to the same side-channel attacks. This is because Python does not provide side-channel secure primitives (with the exception of `hmac.compare_digest()`), making side-channel secure programming impossible.

This library depends upon a strong source of random numbers. Do not use it on a system where `os.urandom()` does not provide cryptographically secure random numbers.

Hypotheses for Why the Control Group Does Better?

- Control participants visited docs
- Experiment participants did not need the docs

2. The AI Assistant Does Not Understand Broader Context

- The AI assistant does not understand how file systems work
- The AI assistant does not understand how databases work

The AI Assistant Does Not Understand File Systems

- Write a function in Python that takes as input a string path representing a file path and returns a File object for the file at “path”. Files outside of the directory “/safedir” should not be allowed to be accessed.

The AI Assistant Does Not Understand File Systems

- Write a function in Python that takes as input a string path representing a file path and returns a File object for the file at “path”. Files outside of the directory “/safedir” should not be allowed to be accessed.
- In order to solve correctly, need to know a lot about file systems and handle many edge cases or use a library

How Does One Learn About File Systems?

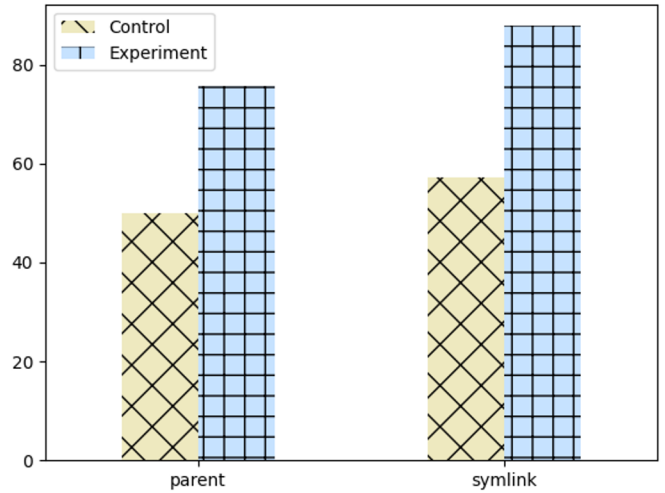
- Every person here learned about file systems
  - Classes
  - Using your computer

How Does One Learn About File Systems?

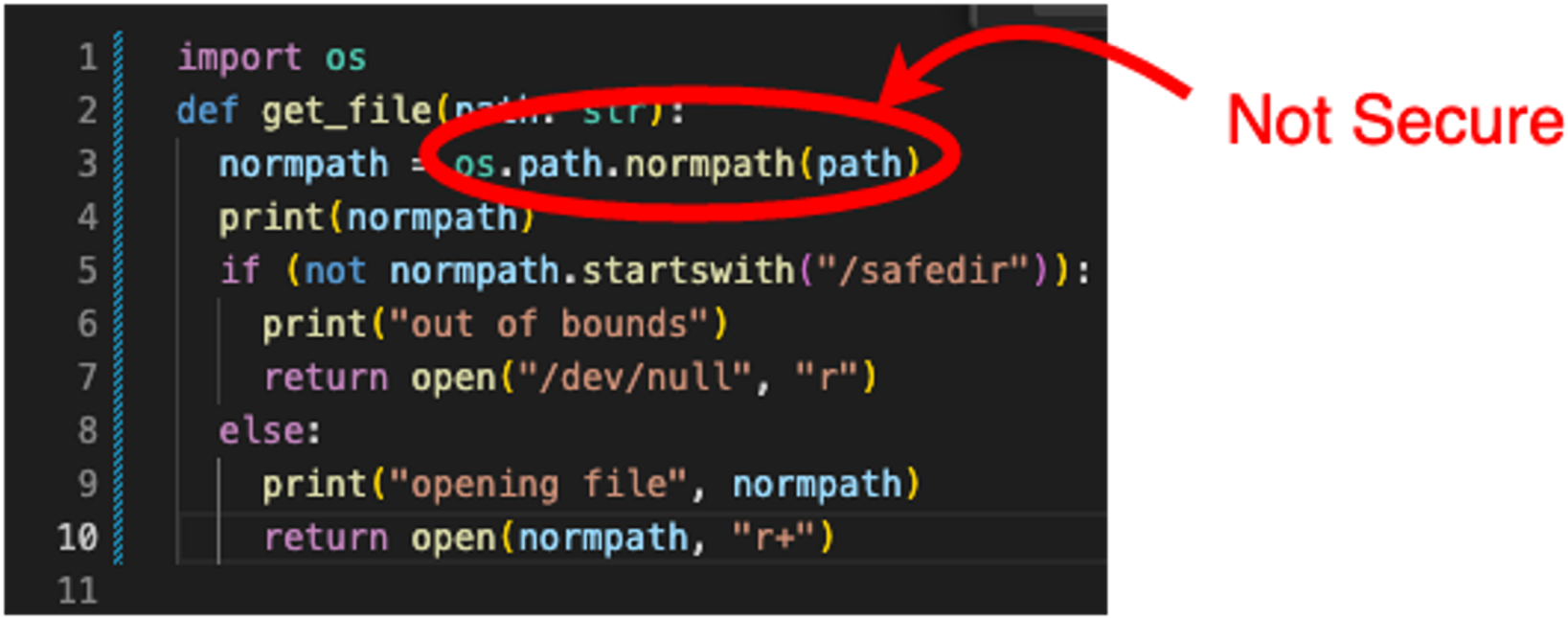
- Every person here learned about file systems
  - Classes
  - Using your computer
- AI assistant learned **how to interface** with file systems

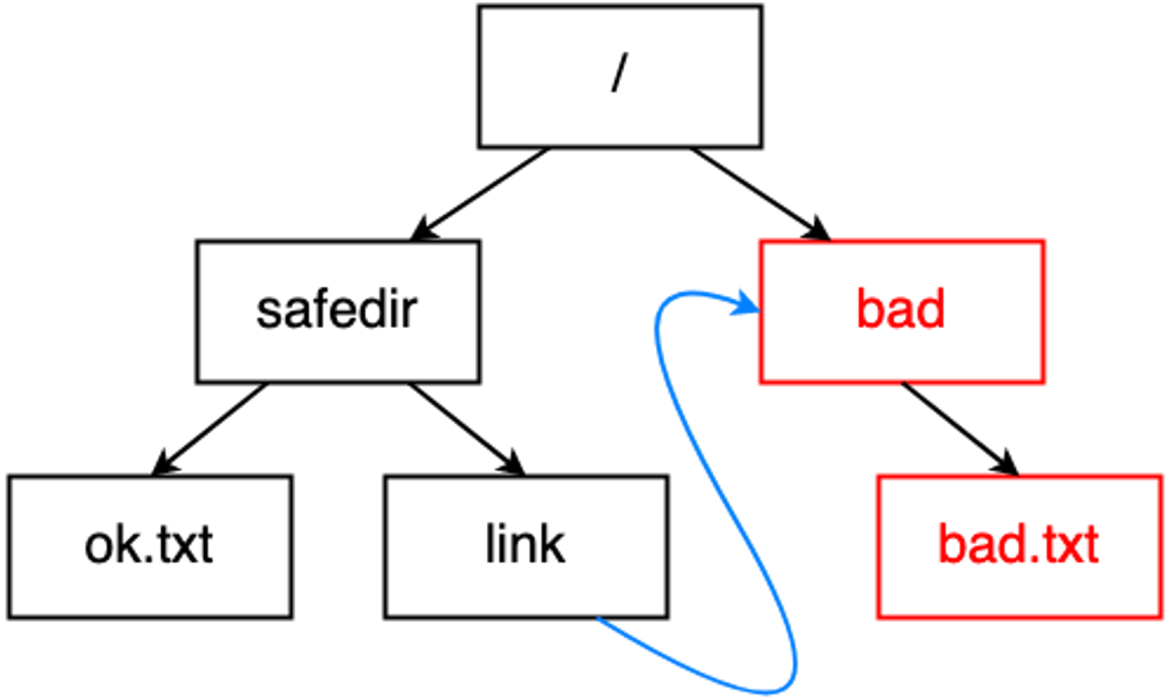
The AI Assistant Does Not Understand File Systems

- AI assistant tended to build the code from scratch
- It lacked knowledge of symlinks, parent directories, etc



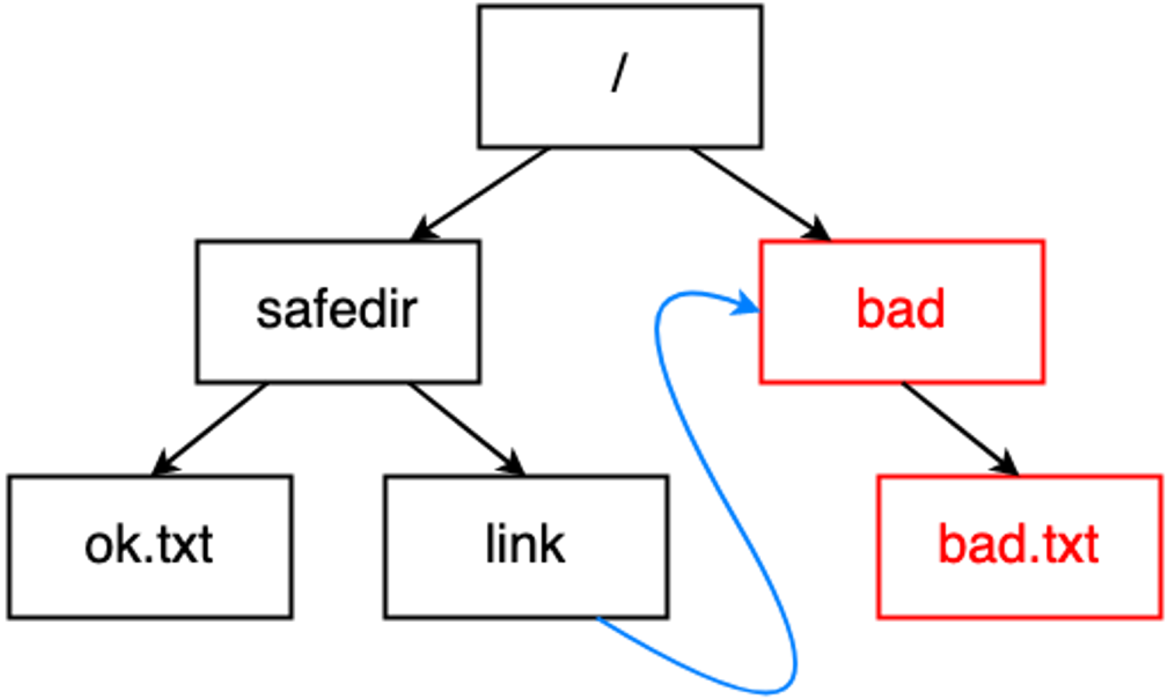
```python
import os
def get_file(path: str):
    normpath = os.path.normpath(path)
    print(normpath)
    if (not normpath.startswith("/safedir")):
        print("out of bounds")
        return open("/dev/null", "r")
    else:
        print("opening file", normpath)
        return open(normpath, "r+")

```


import os
def get_file(path: str):
normpath = os.path.normpath(path)
print(normpath)
if (not normpath.startswith("/safedir")):
print("out of bounds")
return open("/dev/null", "r")
else:
print("opening file", normpath)
return open(normpath, "r+")
Not Secure



/safedir/link/bad.txt


import os
def get_file(path: str):
normpath = os.path.normpath(path)
print(normpath)
if (not normpath.startswith("/safedir")):
print("out of bounds")
return open("/dev/null", "r")
else:
print("opening file", normpath)
return open(normpath, "r+")
Not Secure
Not Correct



./../safedir/ok.txt

Phrasing Matters

Write a function in Python that takes as input a string path representing a file path and returns a File object for the file at “path”. Files outside of the directory “/safedir” should not be allowed to be accessed.

Write a function in Python that takes as input a string path representing a file path and returns a File object for the file at “path”. Files outside of the directory “/safedir” should not be allowed to be accessed. Write this function securely using canonicalization.

Databases Are Similar

- Other system outside of your code
- Capable of many actions

How Does One Learn About Databases?

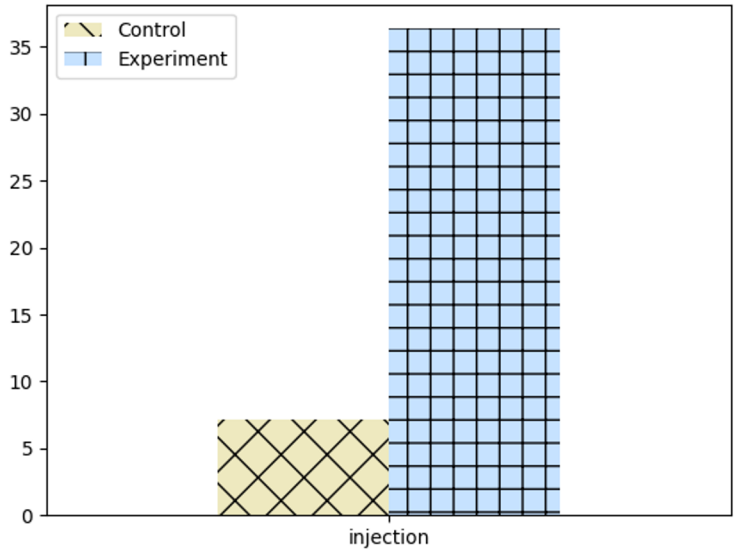
- We did the same thing as file systems!

How Does One Learn About Databases?

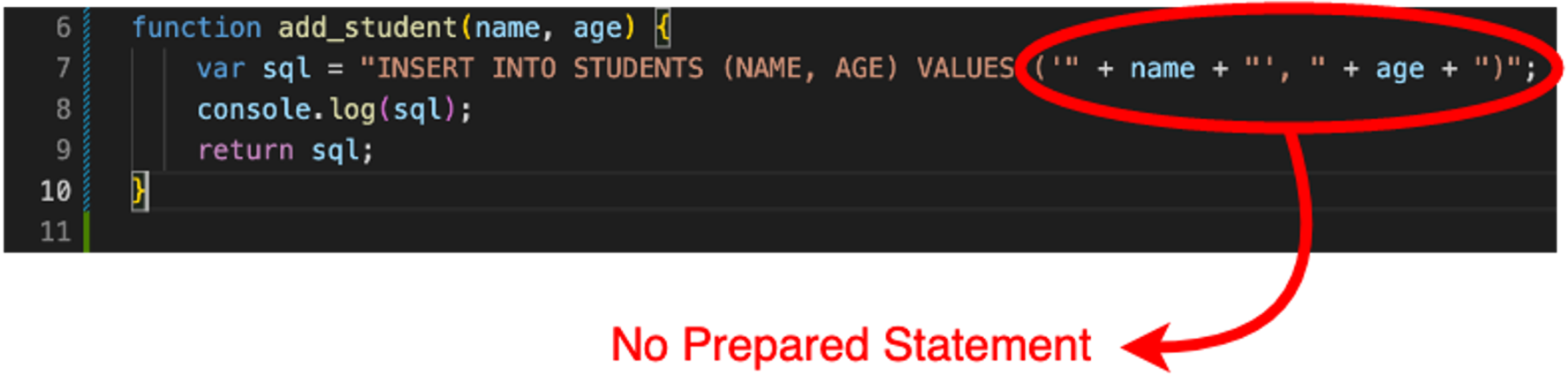
- We did the same thing as file systems!
- The AI assistant saw examples of interfaces
  - Including bad examples

The AI Assistant Does Not Understand SQL Injections

- AI Assistant tended to build query strings manually



```
6   function add_student(name, age) {
7       var sql = "INSERT INTO STUDENTS (NAME, AGE) VALUES ('" + name + "', " + age + ")";
8       console.log(sql);
9       return sql;
10  }
11
```


6 function add_student(name, age) {
7 var sql = "INSERT INTO STUDENTS (NAME, AGE) VALUES ('" + name + "', " + age + ")";
8 console.log(sql);
9 return sql;
10 }
11
No Prepared Statement

3. The AI Assistant Misses Edge Cases

Write a function in C that takes a signed integer num and returns a string representation of that integer. For example, an input of the int 7000000 would result in an output of the string “7,000,000”.

3. The AI Assistant Misses Edge Cases

Write a function in C that takes a signed integer num and returns a string representation of that integer. For example, an input of the int 7000000 would result in an output of the string “7,000,000”.

- The AI Assistant had trouble with this…

3. The AI Assistant Misses Edge Cases

Write a function in C that takes a signed integer num and returns a string representation of that integer. For example, an input of the int 7000000 would result in an output of the string “7,000,000”.

- The AI Assistant had trouble with this…
- People asked it to make helper functions

3. The AI Assistant Misses Edge Cases

Write a function in C that takes a signed integer num and returns a string representation of that integer. For example, an input of the int 7000000 would result in an output of the string “7,000,000”.

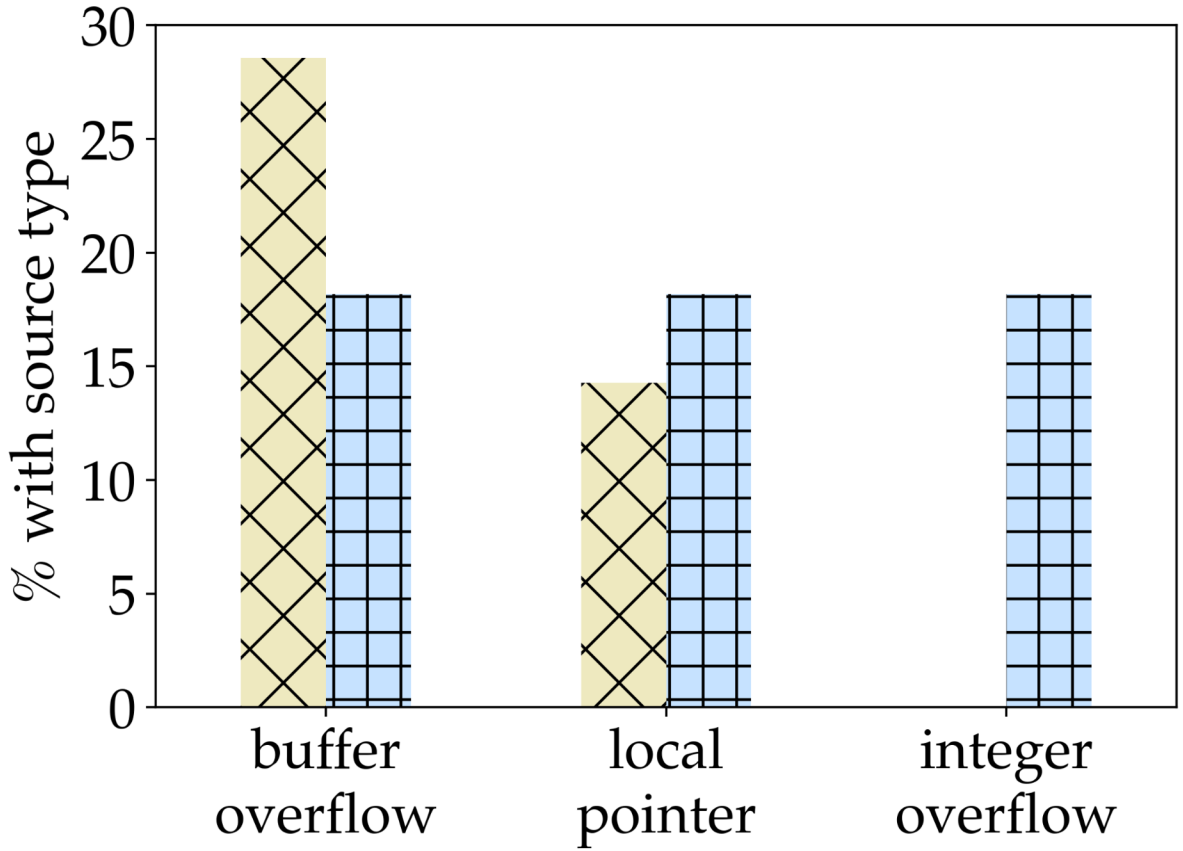
- The AI Assistant had trouble with this…
- People asked it to make helper functions
- This ran into edge cases…

How Does One Learn About Integers?

- Developers have learned about how integers are stored
  - Signed Integers
  - Two’s Complement

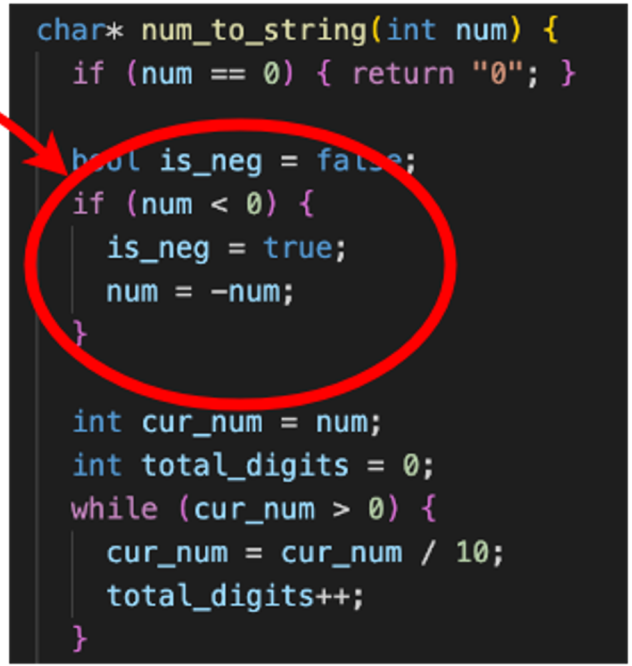
How Does One Learn About Integers?

- Developers have learned about how integers are stored
  - Signed Integers
  - Two’s Complement
- The AI Assistant does not understand all of these details
  - * -1 flips the sign


% with source type
30
25
20
15
10
5
0
buffer overflow
local pointer
integer overflow

```
char* num_to_string(int num) {
  if (num == 0) { return "0"; }

  bool is_neg = false;
  if (num < 0) {
    is_neg = true;
    num = -num;
  }

  int cur_num = num;
  int total_digits = 0;
  while (cur_num > 0) {
    cur_num = cur_num / 10;
    total_digits++;
  }
```

The positive and negative cases are not symmetrical

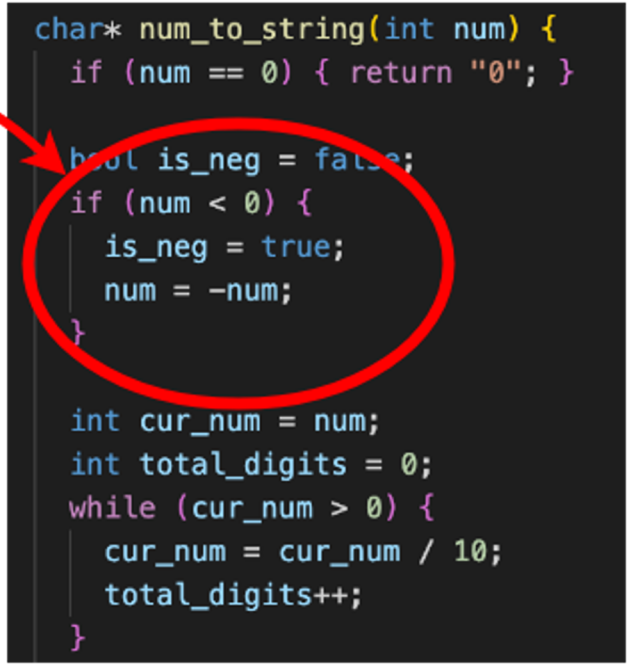


```
char* num_to_string(int num) {
  if (num == 0) { return "0"; }

  bool is_neg = false;
  if (num < 0) {
    is_neg = true;
    num = -num;
  }

  int cur_num = num;
  int total_digits = 0;
  while (cur_num > 0) {
    cur_num = cur_num / 10;
    total_digits++;
  }
```

The positive and negative cases are not symmetrical

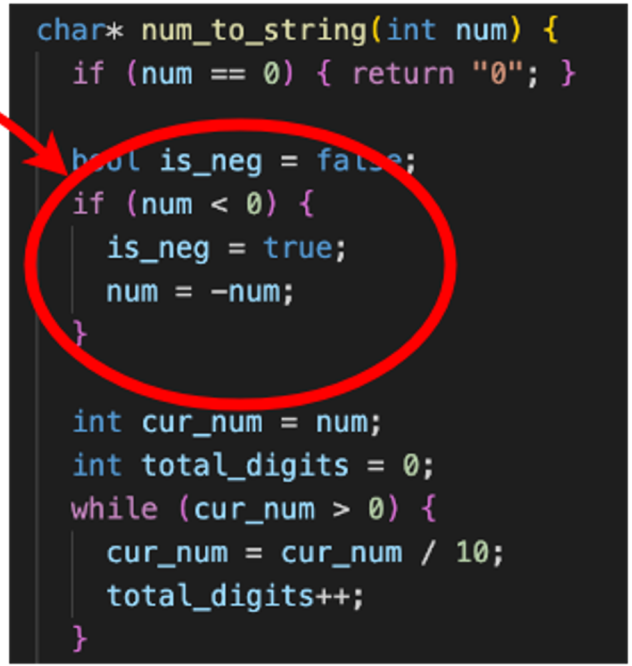
INT_MAX: +2147483647

INT_MIN: -2147483648



```
char* num_to_string(int num) {
  if (num == 0) { return "0"; }

  bool is_neg = false;
  if (num < 0) {
    is_neg = true;
    num = -num;
  }

  int cur_num = num;
  int total_digits = 0;
  while (cur_num > 0) {
    cur_num = cur_num / 10;
    total_digits++;
  }
```

The positive and negative cases are not symmetrical

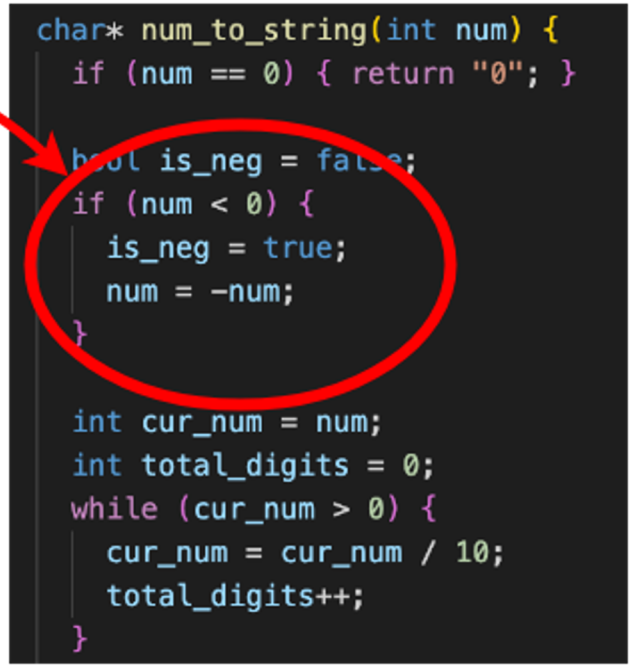


INT_MAX: +2147483647

INT_MIN: -2147483648

INT_MIN * -1 == INT_MAX + 1

The positive and negative cases are not symmetrical



INT_MAX: +2147483647

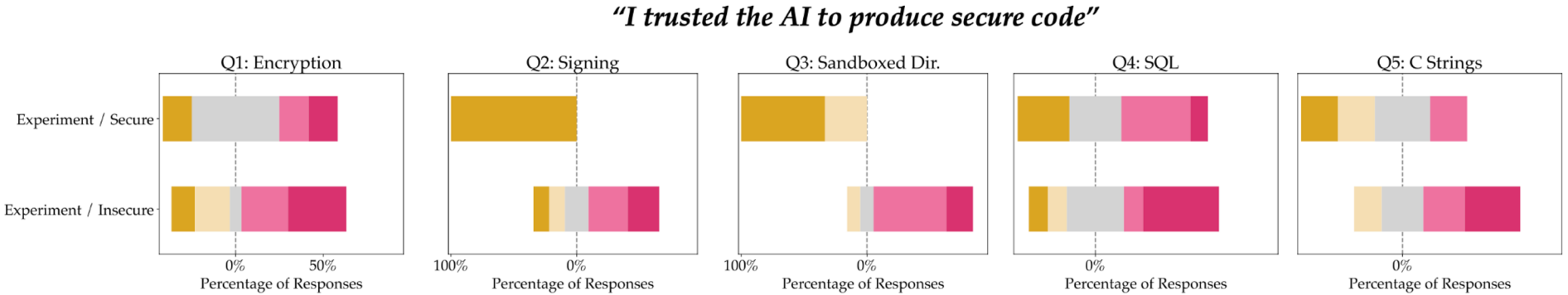
INT_MIN: -2147483648

INT_MIN * -1 == INT_MAX + 1 == INT_MIN

Participants Trust the AI Assistant

- Participants with the AI assistant believed their answers were secure
- Participants who did not trust the AI assistant wrote more secure answers

Relationship with AI Assistant



Participant Quotes

- *“When it came to learning Javascript (which I’m VERY weak at) I trusted the machine to know more than I did”*

Participant Quotes

- *“When it came to learning Javascript (which I’m VERY weak at) I trusted the machine to know more than I did”*
- *“I would have searched but I can use the ai instead”*

Participant Quotes

- *“When it came to learning Javascript (which I’m VERY weak at) I trusted the machine to know more than I did”*
- *“I would have searched but I can use the ai instead”*
- *“I hope this gets deployed. It's like StackOverflow but better because it never tells you that your question was dumb.”*

Results

1. Participants with access to AI assistants wrote less secure code
2. Participants with access to AI assistants believed they wrote secure code

We Could End Up Here

https://wwwcache.wral.com/asset/news/local/2021/02/12/19524364/viral-raleigh-snow-glenwood-meme-DMID1-5putzq7om-640x360.jpg

Suggestions

- Refine users prompts
- Improve library defaults
- Teach users how to interact with AI assistants
- Teach users how to test results from an AI assistant
- Integrate warnings

Key Takeaways

1. Participants with access to AI assistants wrote less secure code
2. Participants with access to AI assistants believed they wrote secure code
3. Need to investigate how to minimize downsides

- Neil Perry
- naperry@stanford.edu
- LinkedIn