

CS 684: Testing and Quality Assurance

Martin Kellogg

Welcome to CS 684!

Welcome to CS 684!

Today's agenda:

- What is 684 + course policies and expectations
- About the instructor (aka why you should listen to me)
 - and introducing our TA
- Survey of the course and brief introduction to topics
- In-class activity: background survey + start HW0

Welcome to CS 684!

Today's agenda:

- **What is 684 + course policies and expectations**
- About the instructor (aka why you should listen to me)
 - and introducing our TA
- Survey of the course and brief introduction to topics
- In-class activity: background survey + start HW0

Course policies

Course policies

- **Most important:** the first time each class you ask or answer a question, I throw candy at you (sorry for poor aim)

Course policies

- **Most important:** the first time each class you ask or answer a question, I throw candy at you (sorry for poor aim)
 - Let's try it now! **Suggested questions:**
 - Why would you do that?
 - Are you just bribing us to pay attention?
 - Does that actually work?
 - Do even silly questions count?

What is CS 684?

- This is a *software engineering* course

What is CS 684?

- This is a *software engineering* course
 - that is, our focus is on **how to build software correctly**

What is CS 684?

- This is a *software engineering* course
 - that is, our focus is on **how to build software correctly**
 - in contrast, some of your other courses may focus on how to build (particular kinds of) software **at all**
 - e.g., an Operating Systems class is about how to build an operating system

What is CS 684?

- This is a *software engineering* course
 - that is, our focus is on **how to build software correctly**
 - in contrast, some of your other courses may focus on how to build (particular kinds of) software **at all**
 - e.g., an Operating Systems class is about how to build an operating system
- This class focuses on two specific, related topics in SE:
 - **testing**, and
 - **quality assurance**

Testing and Quality Assurance

- the title of this course is misleading: “testing” is a really a sub-topic within the broader topic of “quality assurance”

Testing and Quality Assurance

- the title of this course is misleading: “testing” is really a sub-topic within the broader topic of “quality assurance”
 - in particular, it is a kind of *dynamic analysis* (more on this later)

Testing and Quality Assurance

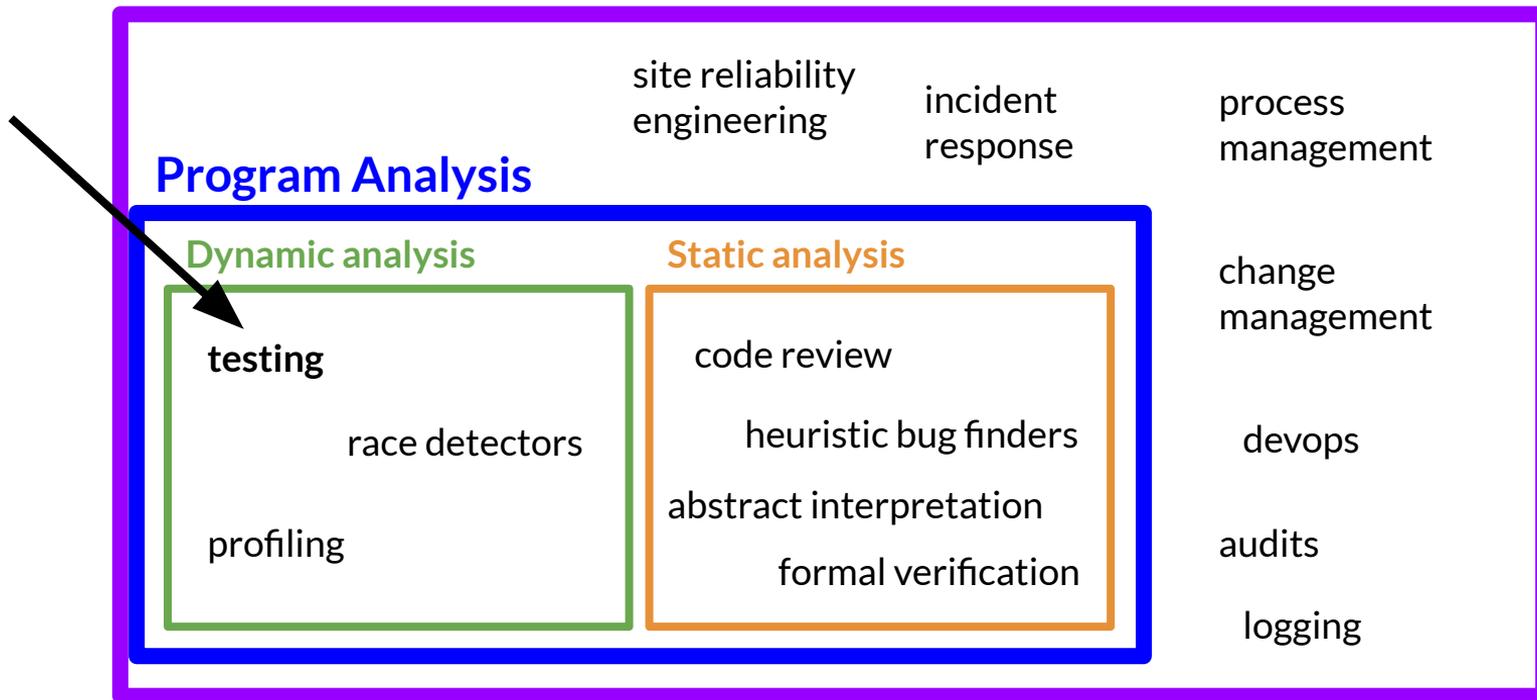
- the title of this course is misleading: “testing” is really a sub-topic within the broader topic of “quality assurance”
 - in particular, it is a kind of *dynamic analysis* (more on this later)
 - which is itself a kind of *program analysis*

Testing and Quality Assurance

- the title of this course is misleading: “testing” is really a sub-topic within the broader topic of “quality assurance”
 - in particular, it is a kind of *dynamic analysis* (more on this later)
 - which is itself a kind of *program analysis*
 - which, you’ll be relieved to learn, is a kind of QA

Testing and Quality Assurance: a bad visual

Quality Assurance



not to scale!

Testing and Quality Assurance

- the title of this course is misleading: “testing” is really a sub-topic within the broader topic of “quality assurance”
 - in particular, it is a kind of *dynamic analysis* (more on this later)
 - which is itself a kind of *program analysis*
 - which, you’ll be relieved to learn, is a kind of QA
- so why is the course titled “Testing **AND** Quality Assurance”?

Testing and Quality Assurance

- the title of this course is misleading: “testing” is really a sub-topic within the broader topic of “quality assurance”
 - in particular, it is a kind of *dynamic analysis* (more on this later)
 - which is itself a kind of *program analysis*
 - which, you’ll be relieved to learn, is a kind of QA
- so why is the course titled “Testing **AND** Quality Assurance”?
 - testing is the **single most important** quality assurance technique in practice

Testing and Quality Assurance

- the title of this course is misleading: “testing” is a really a sub-topic within the broader topic of “quality assurance”
 - in particular, it is a kind of *dynamic analysis* (more on this later)
 - which is itself a kind of *program analysis*
 - which, you’ll be relieved to learn, is a kind of QA
- so why is the course titled “Testing **AND** Quality Assurance”?
 - testing is the **single most important** quality assurance technique in practice
 - we’ll spend ~50% of the course exclusively on testing, the rest on other kinds of QA within the “program analysis” subarea

What's the point of quality assurance?

What's the point of quality assurance?

2023 FAA system outage

From Wikipedia, the free encyclopedia

(Redirected from [2023 FAA system outage in the United States](#))

On January 11, 2023, US flights were grounded or delayed as the [Federal Aviation Administration](#) (FAA) attempted to fix a system outage.^{[1][2]} FAA paused all flight departures until 9 a.m. ET.^[2] Flights already in the air were allowed to continue to their destinations.^[1] Around 8:30 a.m. ET, flights were beginning to resume departures.^[1] The outage was the first time since [September 11, 2001](#) that the FAA issued a nationwide [ground stop](#) in the United States.^[3]

A preliminary investigation of the incident demonstrated to FAA investigators that a "damaged database [file](#)" may have caused the outage of the FAA's Notice to Air Missions ([NOTAM](#)) system, responsible for notifying pilots of safety hazards.^[4] The FAA told [CNN](#) that there was "no evidence of a [cyberattack](#)" on its NOTAM system.^[4]

What's the point of quality assurance?

2023 FAA system outage

From Wikipedia, the free encyclopedia

(Redirected from [2023 FAA system outage in the United States](#))

On January 11, 2023, US flights were grounded or delayed by a major system outage of the [Federal Aviation Administration \(FAA\)](#) attempted system outage.^{[1][2]} FAA paused all flight departures at 10:00 a.m. ET.^[2] Flights already in the air were allowed to continue to their destinations.^[1] Around 8:30 a.m. ET, flights were beginning to resume departures.^[1] The outage was the longest since [September 11, 2001](#) that the FAA issued a nationwide [ground stop](#) in the United States.^[3]

A preliminary investigation of the incident demonstrated to FAA investigators that a "damaged database file" may have caused the outage of the FAA's Notice to Air Mission (NOTAM) system, responsible for notifying pilots of safety hazards.^[4] The FAA told [CNN](#) that there was "no evidence of a [cyberattack](#)" on its NOTAM system.^[4]

Carnegie Mellon

Toyota Case: Single Bit Flip That Killed

Junko Yoshida

10/25/2013 03:35 PM EDT

During the trial, embedded systems experts who reviewed Toyota's electronic throttle source code testified that they found Toyota's source code defective, and that it contains bugs -- including bugs that can cause unintended acceleration.

"We did a few things that NASA apparently did not have time to do," Barr said. For one thing, by looking within the real-time operating system, the experts identified "unprotected critical variables." They obtained and reviewed the source code for the "sub-CPU," and they "uncovered gaps and defects in the throttle fail safes."

The experts demonstrated that "the defects we found were linked to unintended acceleration through vehicle testing," Barr said. "We also obtained and reviewed the source code for the black box and found that it can record false information about the driver's actions in the final seconds before a crash."

Stack overflow and software bugs led to memory corruption, he said. And it turns out that the crux of the issue was these memory corruptions, which acted "like ricocheting bullets."

Barr also said more than half the dozens of tasks' deaths studied by the experts in their experiments "were not detected by any fail safe."

© Copyright 2014, Philip Koopman. CC Attribution 4.0 International license.

Bookout Trial Reporting

http://www.eetimes.com/document.asp?doc_id=1319903&page_number=1
(excerpts)

"Task X death
in combination
with other task
deaths"

What's the point of quality assurance?

2023 FAA system outage

From Wikipedia, the free encyclopedia

(Redirected from [2023 FAA system outage in the United States](#))

On January 11, 2023, US flights were grounded or delayed by a major system outage. The [Federal Aviation Administration](#) (FAA) attempted to resume flights at 10 a.m. ET.^[2] Flights already in the air were allowed to continue to their destinations.^[1] Around 8:30 a.m. ET, flights were beginning to resume departures.^[1] The outage was the longest since [September 11, 2001](#) that the FAA issued a nationwide [ground stop](#) in the United States.^[3]

A preliminary investigation of the incident demonstrated to FAA investigators that a "damaged database file" may have caused the outage of the FAA's Notice to Air Missions (NOTAM) system, responsible for notifying pilots of safety hazards.^[4] The FAA told [CNN](#) that there was "no evidence of a [cyberattack](#)" on its NOTAM system.^[4]

Toyota Case: Single Bit Flip That Killed

Junko Yoshida

10/25/2013 03:35 PM EDT

During the trial, embedded systems experts who reviewed Toyota's electronic throttle source code testified that they found Toyota's source code defective, and that it contains bugs -- including bugs that can cause unintended acceleration.

"We did a few things," Barr said. For example, the experts obtained and reviewed "uncovered gaps."

The experts determined that an unintended acceleration also occurred. They also found that it can occur in the final seconds of a race.

Stack overflow said. And it turns out that there are corruptions, which is what Barr also said.

Barr also said the experts in the field are "safe."

Carnegie Mellon

Bookout Trial Reporting

<http://www.eetimes.com/do>

HealthCare.gov

Learn

Get Insurance

Log in

Español

Individuals & Families

Small Businesses

All Topics

Search

SEARCH

The System is down at the moment.

We're working to resolve the issue as soon as possible. Please try again later.

Please include the reference ID below if you wish to contact us at 1-800-318-2596 for support.

Error from: https%3A//www.healthcare.gov/marketplace/global/en_US/registration%23signUpStepOne

Reference ID: 0.cdd74f17.1380634949.2f9c301c

Health Insurance Marketplace

181 DAYS LEFT TO ENROLL

OCT 1 Open Enrollment Begins

JAN 1 Coverage Can Begin

MAR 31 Open Enrollment Closes

Live Chat

What's the point of quality assurance?

2023 FAA system outage

From Wikipedia, the free encyclopedia

Ariane flight V88^[1] was the failed maiden flight of the **Arianespace Ariane 5** rocket, vehicle no. 501, on 4 June 1996. It carried the **Cluster** spacecraft, a constellation of four **European Space Agency** research satellites.

The launch ended in failure due to multiple errors in the software design: **dead code**, intended only for **Ariane 4**, with inadequate protection against **integer overflow** led to an **exception handled** inappropriately, halting the whole otherwise unaffected **inertial navigation system**. This caused the rocket to veer off its flight path 37 seconds after launch, beginning to disintegrate under high aerodynamic forces, and finally self-destructing via its automated **flight termination system**. The failure has become known as one of the most infamous and expensive **software bugs** in history.^[2] The failure resulted in a loss of more than US\$370 million.^[3]

Toyota Case: Single Bit Flip That Killed

Junko Yoshida

10/25/2013 03:35 PM EDT

During the trial, embedded systems experts who reviewed Toyota's

Carnegie Mellon

Bookout Trial Reporting

<http://www.eetimes.com/do>

Get Insurance

Log in

Español

Search

SEARCH

n at the moment.

soon as possible. Please try again later.

to contact us at 1-800-318-2596 for support.

face/global/en_US/registration%23signUpStepOne
1380634949.2f9c301c

LEFT TO
LL

OCT
1 Open
Enrollment
Began

JAN
1 Coverage
Can Begin

MAR
31 Open
Enrollment
Closes

Live Chat

What's the point of quality assurance?

2023 FAA syst

From Wikipedia, the free encyclo

Ariane flight V88^[1]

vehicle no. 501, on 4
four [European Space](#)

The launch ended in
intended only for [Aria](#)
an [exception handled](#)
[navigation system](#). The

launch, beginning to disintegrate under high aerodynamic forces, and finally self-destructing via its automated [flight termination system](#). The failure has become known as one of the most infamous and expensive [software bugs](#) in history.^[2] The failure resulted in a loss of more than US\$370 million.^[3]

The **Therac-25** was a computer-controlled [radiation therapy](#) machine produced by [Atomic Energy of Canada Limited](#) (AECL) in 1982 after the Therac-6 and Therac-20 units (the earlier units had been produced in partnership with *Compagnie Générale de Radiologie* (CGR) of France).

It was involved in at least six accidents between 1985 and 1987, in which patients were given massive [overdoses of radiation](#).^{[1]:425} Because of [concurrent programming errors](#) (also known as race conditions), it sometimes gave its patients radiation doses that were hundreds of times greater than normal, resulting in death or serious injury.^[2] These accidents highlighted the dangers of software [control](#) of safety-critical systems, and they have become a standard case study in [health informatics](#), [software engineering](#), and [computer ethics](#). Additionally, the overconfidence of the engineers^{[1]:428} and lack of proper [due diligence](#) to resolve reported [software bugs](#) are highlighted as an extreme case where the engineers' overconfidence in their initial work and failure to believe the end users' claims caused drastic repercussions.

face/global/en_US/registration%23signUpStepOne
1380634949.2f9c301c

LEFT TO
LL

OCT 1	Open Enrollment Began	JAN 1	Coverage Can Begin	MAR 31	Open Enrollment Closes
----------	-----------------------------	----------	-----------------------	-----------	------------------------------

Live Chat

What's the point of quality assurance?

2023 FAA syst

From Wikipedia

Arian

vehic

four E

The la

intend

an ex

navig

launch

destructing via its automated [flight termination system](#). The failure has become

known as one of the most infamous and expensive [software bugs](#) in history.^[2] The

failure resulted in a loss of more than US\$370 million.^[3]

& [cyberattack](#) on its [NOTAM](#) system.

The **Therac-25** was a computer-controlled [radiation therapy](#) machine produced by [Atomic Energy of Canada Limited](#) (AECL) in 1982 after the Therac-6 and Therac-20 units (the earlier units had been produced in partnership with *Compagnie Générale de Radiologie* (CGR) of France).

Code has consequences
in the real world!

massive
e conditions),
l, resulting in
-critical
ering, and
[due diligence](#)
erconfidence

gnUpStepOne

LEFT TO
LL

OCT 1	Open Enrollment Began	JAN 1	Coverage Can Begin	MAR 31	Open Enrollment Closes
----------	-----------------------------	----------	-----------------------	-----------	------------------------------

Live Chat

Will all of you work on life-or-death systems?

Will all of you work on life-or-death systems?

- **Of course not!**

Will all of you work on life-or-death systems?

- **Of course not!**
- But, the techniques that were developed there are useful for **getting your systems correct**, too.

Will all of you work on life-or-death systems?

- **Of course not!**
- But, the techniques that were developed there are useful for **getting your systems correct**, too.
 - and, it's good to **be aware** of higher-assurance techniques, for the rare times that you do need to use them

My expectations

- You know how to *program*

My expectations

- You know how to *program*
 - you can write code
 - you can program against an English specification
 - you can read code and figure out what it does
 - you can teach yourself a new programming language
 - you can debug code that's not behaving like you expect
 - you can install software yourself + do basic troubleshooting
 - when you get stuck, you know how to Google for answers

My expectations

- You know how to *program*

- you can write code
- you can program against an English specification
- you can read code and figure out what it does
- you can teach yourself a new programming language
- you can debug code that's not behaving like you expect
- you can install software yourself + do basic troubleshooting
- when you get stuck, you know how to **Google** for answers

The ability to solve problems yourself with just a search engine is a **critical** skill for a software engineer!

My expectations

- You know how to *program*
- Professionalism

My expectations

- You know how to *program*
- Professionalism
- Participation

Your expectations

Your expectations / my priorities

- In this class, I'll cover cutting-edge quality assurance techniques that are used in the **best** software engineering firms in the world
- You will need to both read and write programs
 - surprisingly, this class involves reading more than writing programs
- Fair and fast grading
- Reasonably prompt responses to your questions (“within 1 business day” unless special circumstances, e.g., I have the flu)

Grading

- My classes are usually considered difficult
 - e.g., a common complaint on course evaluations

Grading

- My classes are usually considered difficult
 - e.g., a common complaint on course evaluations
 - hint: I take this as a compliment

Grading

- My classes are usually considered difficult
 - e.g., a common complaint on course evaluations
 - hint: I take this as a compliment
 - I expect that most of you will feel similarly about this course

Grading

- My classes are usually considered difficult
 - e.g., a common complaint on course evaluations
 - hint: I take this as a compliment
 - I expect that most of you will feel similarly about this course
- My exams will probably still be difficult for most of you
 - e.g., last semester's CS 490 final had an average of 87/150

Grading

- My classes are usually considered difficult
 - e.g., a common complaint on course evaluations
 - hint: I take this as a compliment
 - I expect that most of you will feel similarly about this course
- My exams will probably still be difficult for most of you
 - e.g., last semester's CS 490 final had an average of 87/150
- The class will be aggressively curved
 - e.g., in last semester's CS 490, everyone with a raw score over 80 got an A

Welcome to CS 684!

Today's agenda:

- What is 684 + course policies and expectations
- **About the instructor (aka why you should listen to me)**
 - and introducing our TA
- Survey of the course and brief introduction to topics
- In-class activity: background survey + start HW0

Who am I?

- I started at NJIT as an assistant professor in Fall 2022
- Previously:
 - PhD at University of Washington (Seattle) until June 2022
 - BS at University of Virginia (Charlottesville) in 2016



Who am I?

- I started at NJIT as an assistant professor in Fall 2022
- Previously:
 - PhD at University of Washington (Seattle) until June 2022
 - BS at University of Virginia (Charlottesville) in 2016

I'm an academic, not a professional software engineer



So what do I know about quality assurance?

So what do I know about quality assurance?

- My research area is in software engineering, related to QA
 - more specifically, static analysis design (“compilers”)

So what do I know about quality assurance?

- My research area is in software engineering, related to QA
 - more specifically, static analysis design (“compilers”)
- ~25% of my PhD spent embedded at AWS
 - two co-authored publications
 - my analysis tools deployed on > 70M lines of AWS code

So what do I know about quality assurance?

- My research area is in software engineering, related to QA
 - more specifically, static analysis design (“compilers”)
- ~25% of my PhD spent embedded at AWS
 - two co-authored publications
 - my analysis tools deployed on > 70M lines of AWS code
- My lab is one of the few in the world to take SE seriously when writing research code
 - Inherited from my PhD advisor, who employed 3 SDEs concurrently while I was a student!

Welcome to CS 684!

Today's agenda:

- What is 684 + course policies and expectations
- About the instructor (aka why you should listen to me)
 - **and introducing our TA**
- Survey of the course and brief introduction to topics
- In-class activity: background survey + start HW0

Our TA: Kazi Siddiqui

- Kazi is one of my PhD students
- He has deep expertise in machine learning and in program analysis
- He's a good *systems programmer*, which is helpful for many of the assignments in this class
 - go to him for help!
- Office hours Mondays 3:30-5, on Webex



Welcome to CS 684!

Today's agenda:

- What is 684 + course policies and expectations
- About the instructor (aka why you should listen to me)
 - and introducing our TA
- **Survey of the course and brief introduction to topics**
- In-class activity: background survey + start HW0

Course structure

- Class time will generally be structured as a lecture followed by an **in-class activity**

Course structure

- Class time will generally be structured as a lecture followed by an **in-class activity**
 - in-class activities are usually the homework assignments
 - exceptions: background survey, midterm, course eval

Course structure

- Class time will generally be structured as a lecture followed by an **in-class activity**
 - in-class activities are usually the homework assignments
 - exceptions: background survey, midterm, course eval
- Each class will start with a **reading quiz**
 - check the calendar on the course website for each week's readings
 - you get $\frac{1}{2}$ credit on the reading quiz for writing you name + UCID, $\frac{1}{2}$ credit for getting the questions correct

Course structure

Reading quizzes can also cover any part of the **course syllabus** at any point during the semester

- Class time will generally be spent on **in-class activity**
 - in-class activities are usually **10-15 minutes**
 - exceptions: background survey, midterm, course eval
- Each class will start with a **reading quiz**
 - check the calendar on the course website for each week's readings
 - you get $\frac{1}{2}$ credit on the reading quiz for writing your name + UCID, $\frac{1}{2}$ credit for getting the questions correct

In-class activities/homeworks

- Homeworks are generally due the following Wednesday, “**Anywhere on Earth**” (aka “AoE”)
 - this means that as long as it’s still Wednesday anywhere on the planet, you can submit with no penalty
 - in EST, this is 7am on Thursday morning

In-class activities/homeworks

- Homeworks are generally due the following Wednesday, “**Anywhere on Earth**” (aka “AoE”)
 - this means that as long as it’s still Wednesday anywhere on the planet, you can submit with no penalty
 - in EST, this is 7am on Thursday morning
- You may submit a homework assignment up to 24 hours late
 - called “using a late day”

In-class activities/homeworks

- Homeworks are generally due the following Wednesday, “**Anywhere on Earth**” (aka “AoE”)
 - this means that as long as it’s still Wednesday anywhere on the planet, you can submit with no penalty
 - in EST, this is 7am on Thursday morning
- You may submit a homework assignment up to 24 hours late
 - called “using a late day”
 - you can use **up to 2** late days throughout the semester

In-class activities/homeworks

- Homeworks are generally due the following Wednesday, “**Anywhere on Earth**” (aka “AoE”)
 - this means that as long as it’s still Wednesday anywhere on the planet, you can submit with no penalty
 - in EST, this is 7am on Thursday morning
- You may submit a homework assignment up to 24 hours late
 - called “using a late day”
 - you can use **up to 2** late days throughout the semester
 - submissions later than that **will not be accepted**

In-class activities/homeworks

- Homeworks are generally due the following Wednesday, “**Anywhere on Earth**” (aka “AoE”)
 - this means that as long as it’s still Wednesday anywhere on the planet, you can submit with no penalty
 - in EST, this is 7am on Thursday morning
- You may submit a homework assignment
 - called “using a late day”
 - you can use **up to 2** late days throughout the semester
 - submissions later than that **will not be accepted**

Homeworks are the **bulk of your grade** in this class. Take them seriously!

Working with Others

- For all homework assignments, you are permitted to work in **pairs** (you may choose to work alone)

Working with Others

- For all homework assignments, you are permitted to work in **pairs** (you may choose to work alone)
 - exception: HW0 (today), because everyone must have a working setup. However, you are allowed to help each other with HW0 in any way except copying `output.txt`

Working with Others

- For all homework assignments, you are permitted to work in **pairs** (you may choose to work alone)
 - exception: HW0 (today), because everyone must have a working setup. However, you are allowed to help each other with HW0 in any way except copying `output.txt`
 - larger groups are not permitted

Working with Others

- For all homework assignments, you are permitted to work in **pairs** (you may choose to work alone)
 - exception: HW0 (today), because everyone must have a working setup. However, you are allowed to help each other with HW0 in any way except copying `output.txt`
 - larger groups are not permitted
- In general, you are welcome to use **any resource** that you'd like, **as long as you cite your source**

Working with Others

- For all homework assignments, you are permitted to work in **pairs** (you may choose to work alone)
 - exception: HW0 (today), because everyone must have a working setup. However, you are allowed to help each other with HW0 in any way except copy
 - larger groups are not permitted
- In general, you are welcome to use **AI** **as long as you cite your source**

All homeworks after HW0 will have a **written report** that you need to submit. Cite your sources there.

Working with Others

- For all homework assignments, you are permitted to work in **pairs** (you may choose to work alone)
 - exception: HW0 (today), because everyone must have a working setup. However, you are allowed to help each other with HW0 in any way except copying `output.txt`
 - larger groups are not permitted
- In general, you are welcome to use **any resource** that you'd like, **as long as you cite your source**
 - e.g., ChatGPT is okay (as long as you cite it!)

Working with Others

- For all homework assignments, you are permitted to work in **pairs** (you may choose to work alone)
 - exception: HW0 (today), because everyone must have a working setup. However, you are allowed to help each other with HW0 in any way except copying `output.txt`
 - larger groups are not permitted
- In general, you are welcome to use **any resource** that you'd like, **as long as you cite your source**
 - e.g., ChatGPT is okay (as long as you cite it!)
 - exception: you don't need to cite sources for HW0

Exams

- This course has two exams:

Exams

- This course has two exams:
 - a midterm (worth 10% of your total course grade) right after spring break

Exams

- This course has two exams:
 - a midterm (worth 10% of your total course grade) right after spring break
 - a cumulative final (worth 15% of your total course grade) during the final exam slot
 - while the final is cumulative, except it to emphasize topics that we covered after the midterm

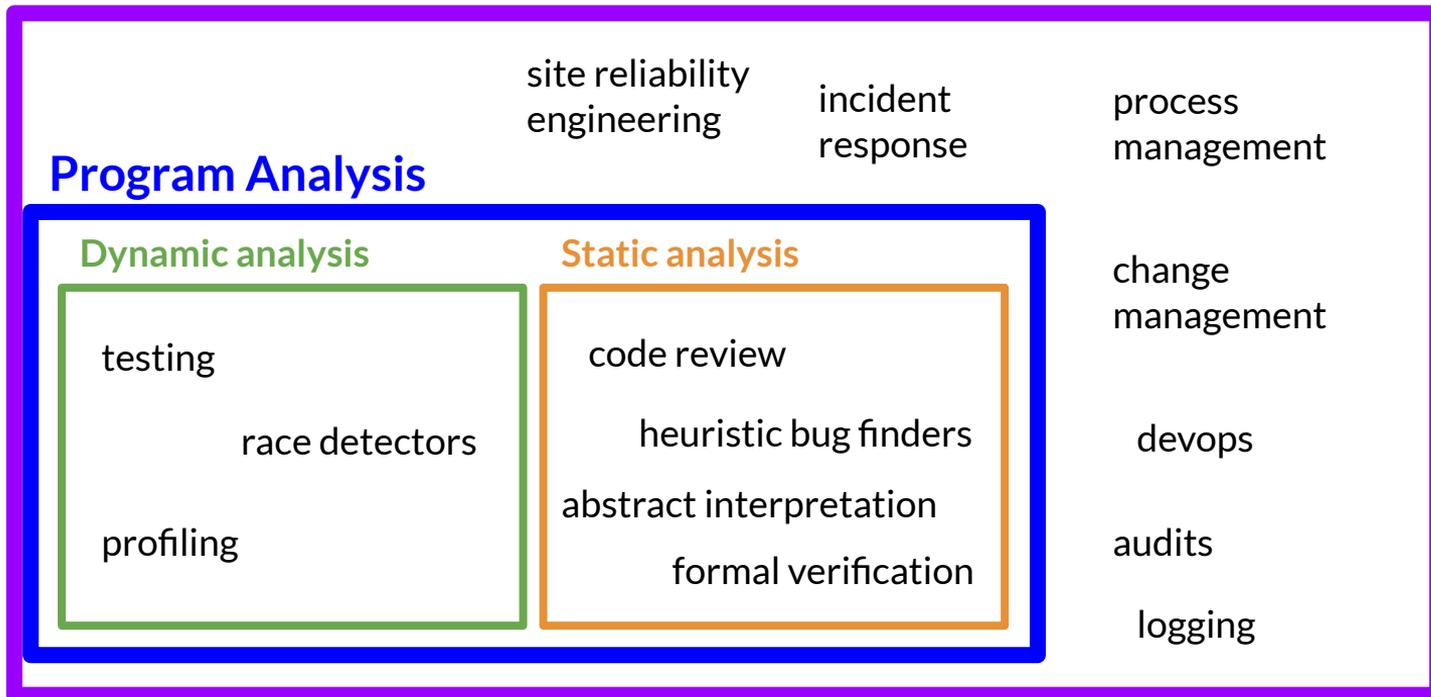
Exams

- This course has two exams:
 - a midterm (worth 10% of your total course grade) right after spring break
 - a cumulative final (worth 15% of your total course grade) during the final exam slot
 - while the final is cumulative, expect it to emphasize topics that we covered after the midterm
- I may add short (10 minute) “quizzes”/mini-exams to reading quizzes 2-3 times during the semester
 - if you miss up to one of these, there is no penalty

Course Content

Course Content

Quality Assurance



not to scale!

Course Content

Quality Assurance

Before
Spring
Break

Program Analysis

Dynamic analysis

testing

race detectors

profiling

Static analysis

code review

heuristic bug finders

abstract interpretation

formal verification

site reliability
engineering

incident
response

process
management

change
management

devops

audits

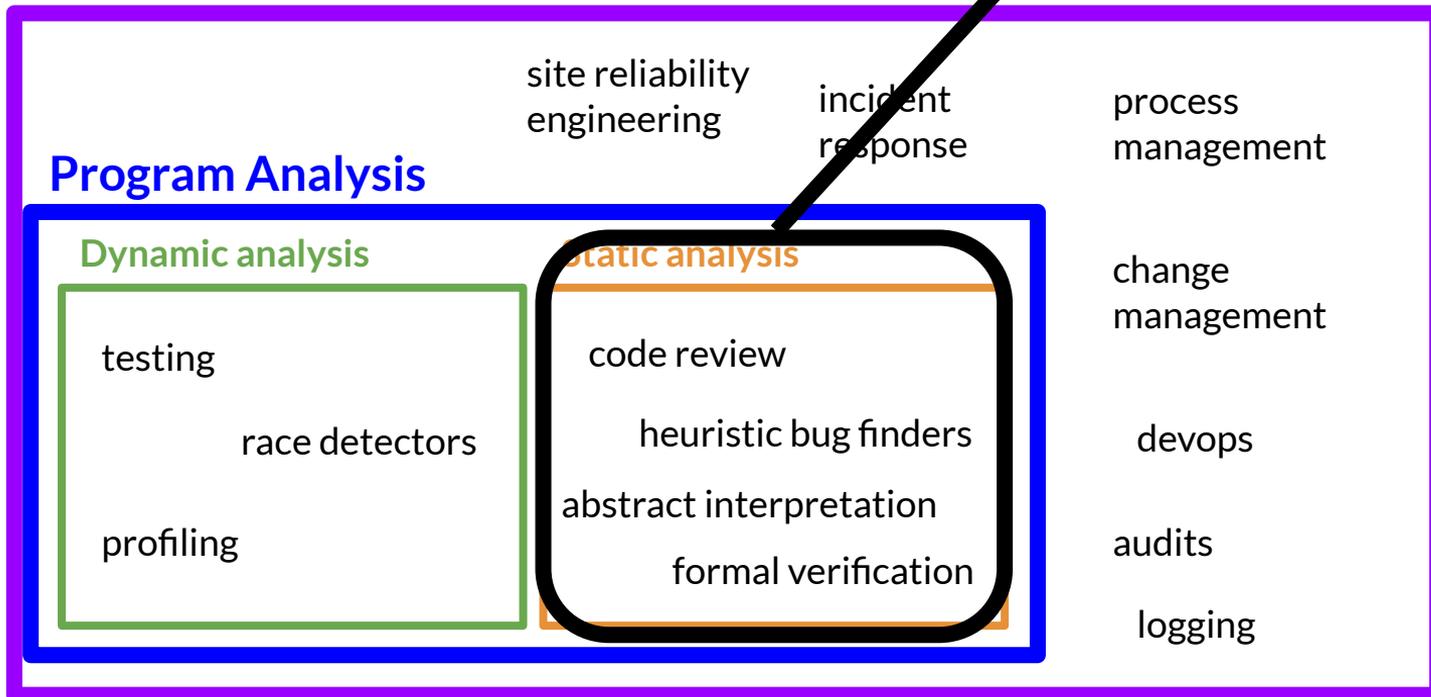
logging

not to scale!

Course Content

After
Spring
Break

Quality Assurance



not to scale!

Course Content

- Full list of topics is on the calendar on the course website
 - all readings are also already up

Course Content

- Full list of topics is on the calendar on the course website
 - all readings are also already up
- I know what the assignments will be (see HW titles on the webpage), but I haven't yet converted them into a standard form
 - they come from a mix of courses at other universities
 - I will try to get them up ASAP (meant to do it this weekend, but I had the flu)

Course Content

- Full list of topics is on the calendar on the course website
 - all readings are also already up
- I know what the assignments will be (see HW titles on the webpage), but I haven't yet converted them into a standard form
 - they come from a mix of courses at other universities
 - I will try to get them up ASAP (meant to do it this weekend, but I had the flu)
- My slides will go up the day after lecture
 - useful for studying for exams!

Acknowledgements

- This is a good time for me to **cite my sources**:

Acknowledgements

- This is a good time for me to **cite my sources**:
 - Wes Weimer's EECS 481 at the University of Michigan



Acknowledgements

- This is a good time for me to **cite my sources**:
 - Wes Weimer's EECS 481 at the University of Michigan
 - René Just's CSE 504P at the University of Washington



Acknowledgements

- This is a good time for me to **cite my sources**:
 - Wes Weimer's EECS 481 at the University of Michigan
 - René Just's CSE 504P at the University of Washington
- You might find their lecture slides on these topics useful, too, if you need a different perspective



Welcome to CS 684!

Today's agenda:

- What is 684 + course policies and expectations
- About the instructor (aka why you should listen to me)
 - and introducing our TA
- Survey of the course and brief introduction to topics
- **In-class activity: background survey + start HW0**

Today's in-class activities

Two things to do in-class today:

- a **background survey** to help me understand what you're hoping to get out of the class and your existing knowledge base
 - also used to verify your presence today
- get started on **HWO** (dev setup), which is due next Wednesday. Kazi and I are here to help (as we will be each week).

You can do these in any order. However, the background survey closes on **Saturday morning at 7am**, so you must complete it by then.

Today's in-class activities

Links to both are also available on the course webpage at:

<https://web.njit.edu/~mjk76/teaching/cs684-sp24/>

Background survey



HW0

