GUIDE

CS for PGH Quickstart Guide

powered by remake learning

Small steps educators can take to expand opportunities for computer science learning in school

// ABOUT CSFORPGH

CSforPGH is a regional collaborative composed of individuals and organizations who seek to equitably equip and empower learners to use computer science, responsibly, as a tool to change their world. We prioritize problem-solving, logical thinking, discovery, and learning from failure as necessities to acquire this essential literacy.

Representing more than 50 organizations including schools, museums, libraries, after-school programs, ed-tech companies, universities, employers, and foundations, CSforPGH seeks to prioritize youth learning, build confidence in teaching, cultivate digital ethics, and build computer science capacity in the greater Pittsburgh region.

>> OUR VISION

Each and every child in our region has the opportunity to learn computer science

Educators in our region are confident and supported in their teaching of computer science

CSforPGH is a working group of Remake Learning, a network in the Pittsburgh region that ignites engaging, relevant, and equitable learning practices. Learn more at **csforpgh.org**

//GETTING STARTED

As computer science (CS) becomes a more mainstream subject in schools across the state of Pennsylvania, teachers are looking for ways to make this complex topic more approachable and empowering—both for their students and for themselves as educators.

Starting out in computer science can feel like a daunting task, especially for educators with little technical experience and for schools with limited access to computer hardware and software. But it doesn't have to be!

The CSforPGH Quickstart Guide is for educators looking for ways to integrate computer science and computational thinking into their teaching practice.

Drawing on examples of CS education practiced by K–12 teachers and out-of-school educators throughout the Pittsburgh region, this guide includes 25 "quick starts" as points of inspiration. Each quickstart offers a way for teachers to start small, learn valuable lessons together with their peers and students, and build on what works to take CS education to scale in their classroom, school building, or even across their whole district.

The quickstarts offered in this guide support teachers and educational leaders looking to take action steps identified in the CSforALL Blueprint for Action:

- Amplifying Communication and Public Awareness about CS
- Expanding Professional Development and Capacity Building
- Sharing Promising Practices, Evaluation, and Standards
- Advancing Program and Public Policies

Educators in the Pittsburgh region are on the leading edge of this national effort to bring CS to all students. **See more at csforall.org**

//THE QUICKSTARTS

The quickstarts in this guide offer a range of starting points, from one-and-done "hour of code" activities to year-long student projects where the teachers learn right alongside the students. Choose the starting point that works best for you and your students:

>> IF YOU'RE JUST STARTING OUT

>> IF YOU'RE READY TO LEVEL UP

>> IF YOU'RE READY FOR A REAL CHALLENGE

>> IF YOU'RE READY TO CHANGE YOUR SCHOOL

[SIDENOTE: LEVERS OF CHANGE]

The quickstarts in this guide are inspired by the "Levers for Change" concept used by School Retool. Levers for change are a set of methods educators can use to make change in their school. Learn more at **schoolretool.org**.

EVENTS
PROCESSES
FINANCE
SCHEDULES
ROLES
COMMUNICATIONS
RITUALS
INCENTIVES
SPACES

//SET UP

What is Computer Science?

Computer science means a lot of things to a lot of people. One definition of computer science is the study and application of mathematical processes to manipulate information.

Behind computer science is computational thinking, a way of solving problems that involves creating abstractions of the problem's component parts, using the principles of logic to understand the relationships between those components, and formulating step-by-step processes (algorithms) to solve the problem.

Writing code and programming software is how computer scientists make their work real, but the tools they're using to do it—problem-solving, logical thinking, discovery, and learning from failure—are all part of computational thinking. CSforPGH believes that this mindset is an essential literacy for today's students and it's our responsibility to equip educators with the resources and support they need to empower each and every student with the tools of CS.

>> KEY TERMS

from the code.org K–12 CS Framework

Abstraction: A new representation of a thing, a system, or a problem that helpfully re-frames a problem by hiding details irrelevant to the question at hand.

Algorithm: A step-by-step process to complete a task.

Code: Any set of instructions expressed in a programming language.

Computational Thinking: The human ability to formulate problems so that their solutions can be represented as computational steps or algorithms to be executed by a computer.

Computer: A machine or device that performs processes, calculations, and operations based on instructions provided by a software or hardware program.

Computer Science: The study of computers and algorithmic processes, including their principles, their hardware and software designs, their implementation, and their impact on society.

Data: Information that is collected and used for reference or analysis.

Digital: A characteristic of electronic technology that uses discrete values, generally 0 and 1, to generate, store, and process data.

Digital Citizenship: The norms of appropriate, responsible behavior when using technology.

Hardware: The physical components that make up a computing device.

Internet: A global network of computers all using shared protocols to communicate.

Network: A group of devices that exchange information through wired or wireless connections.

Program: A set of instructions that the computer executes to achieve a particular objective.

Software: Programs that run on a computing system, computer, or other computing device.

//OPERATING SYSTEM

>> WHY COMPUTER SCIENCE MATTERS

Computer science knowledge and computational thinking skills are quickly becoming essential competencies for modern life. Over the past two decades, we've come to rely on computers to connect with our families, to shop for shoes, to do our school work, to stay in touch with friends, to view our medical records, to express ourselves, to access the news, and to participate in democratic society.

Today's school students are growing up immersed in computer science and as they become adults, they'll enter a world where computing has become ubiquitous. So we need to make sure they are skilled computer users. And we need to make sure they understand how these powerful systems work, so that they can fully participate in building the world of tomorrow.

And of course, there is the economic reality to consider. The Bureau of Labor Statistics estimates employment in computer and information technology occupations to grow 13% through 2026. The growth rate is even higher in Pennsylvania at 26%. Plus, computer science jobs have average salaries more than twice the median annual wage in the U.S.

[STATS: CAREERS IN CS]

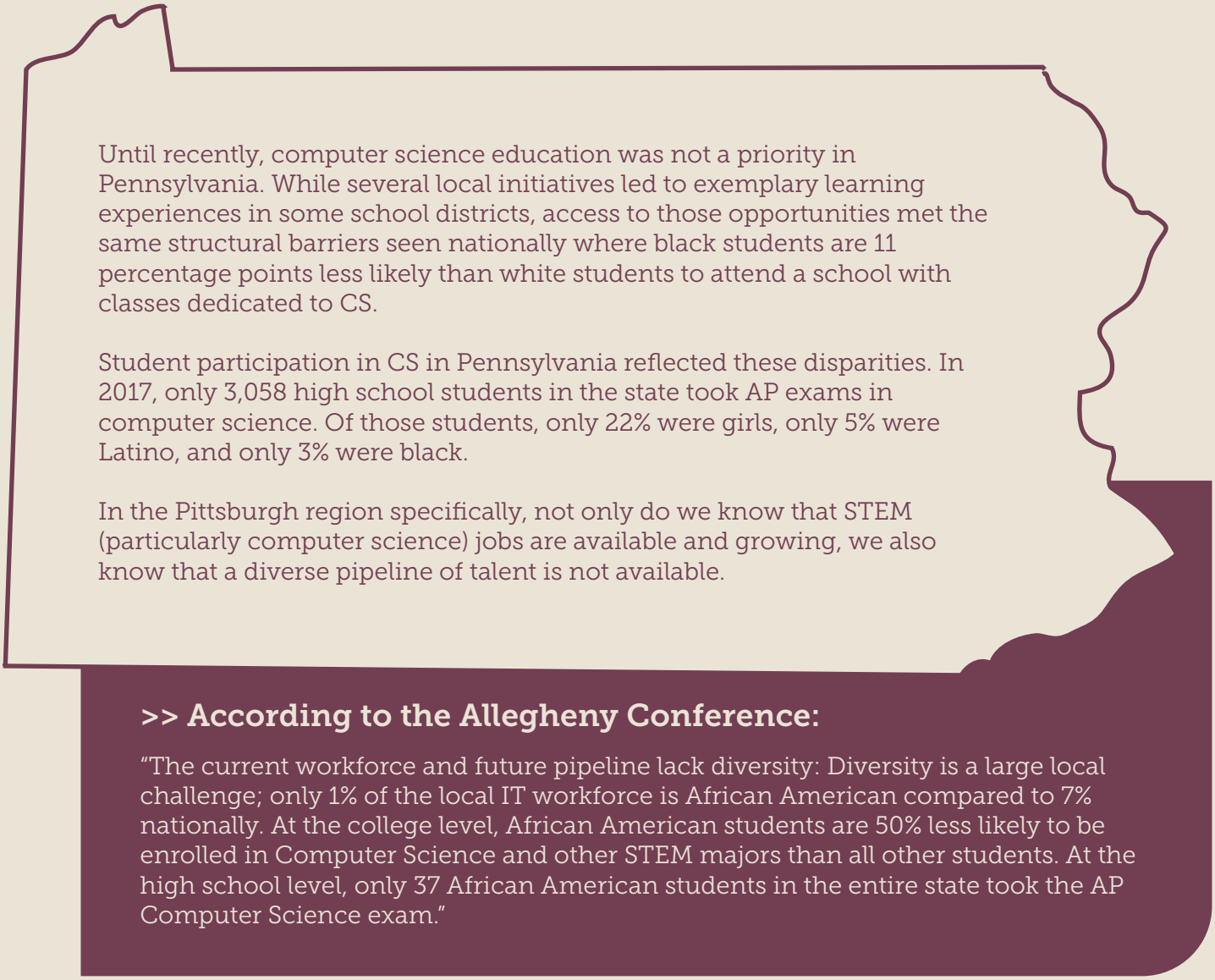
Number one source of all **new wages** in the U.S.

20,000+ **open computer science jobs** in Pennsylvania

$85,000 average **annual salary** in Pennsylvania

//CONFIGURATION

The state of computer science education in Pennsylvania



Until recently, computer science education was not a priority in Pennsylvania. While several local initiatives led to exemplary learning experiences in some school districts, access to those opportunities met the same structural barriers seen nationally where black students are 11 percentage points less likely than white students to attend a school with classes dedicated to CS.

Student participation in CS in Pennsylvania reflected these disparities. In 2017, only 3,058 high school students in the state took AP exams in computer science. Of those students, only 22% were girls, only 5% were Latino, and only 3% were black.

In the Pittsburgh region specifically, not only do we know that STEM (particularly computer science) jobs are available and growing, we also know that a diverse pipeline of talent is not available.

>> According to the Allegheny Conference:

"The current workforce and future pipeline lack diversity: Diversity is a large local challenge; only 1% of the local IT workforce is African American compared to 7% nationally. At the college level, African American students are 50% less likely to be enrolled in Computer Science and other STEM majors than all other students. At the high school level, only 37 African American students in the entire state took the AP Computer Science exam."

[STATS: THE CS OPPORTUNITY GAP]

Only 40% of K–12 schools in PA **teach CS**

Only 10% of black students in PA show **STEM college/career readiness**

Only 8% of **rural schools** and 9% of **city schools** offer AP CS courses

//TROUBLESHOOTING

IT'S TIME FOR EACH AND EVERY CHILD TO HAVE CS EDUCATION OPPORTUNITIES

There is a clear and present need for widespread adoption of K–12 computer science education. The need is only more urgent for students from historically marginalized and under-served populations including learners of color, learners in poverty, learners in rural communities, girls in STEM, and learners with exceptionalities.

In response to these structural inequities, Pennsylvania's elected officials are taking steps to ensure that all students are prepared to participate fully in civic, economic, and political life shaped by computer science.

Act 86, signed into law in July 2016, allows high school students "to apply up to one credit towards a math or science high school graduation credit requirement upon successful completion of computer science or information technology coursework."

In January 2018, the State Board of Education endorsed the Computer Science Teacher Association K–12 Standards as a statewide framework for offering computer science education to all public school students in the Commonwealth.

As more CS education infrastructure is put into place, it's time for teachers in Pennsylvania to share ideas for effective K–12 CS learning.

>> Ready to bring CS to your school? Read on for ways to get started.

>>> QUICKSTART:

Building Capacity for CS

When educators at Fort Cherry School District wanted to expand CS K–12, they faced two very real and very common obstacles: zero funding and no specialist faculty. They decided to build slowly and intentionally, starting with establishing a team of teachers, students, administrators, and partners who could learn to teach CS.

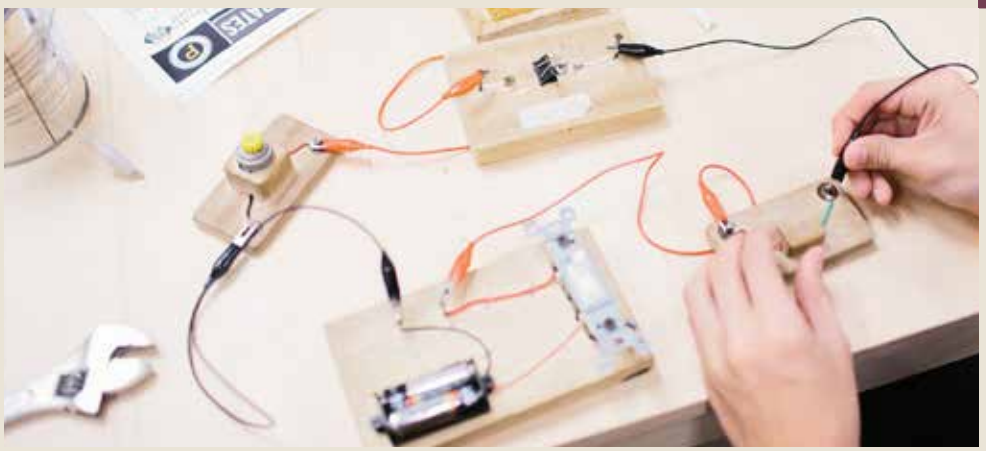
HACK DEVELOPED BY

Fort Cherry School District

A small, rural public school district in Washington County, PA about 20 miles southwest of the city of Pittsburgh. Fort Cherry serves a student population of about 1,100 across grades K–12. Learn more at **fortcherry.org**

CONTACT

Trisha Craig, High School Principal
tcraig@fortcherry.org



// Talking Points

If we're going to expect our students to learn computer science, we should be ready to learn too.

This is about meeting the needs of our students. We need to take responsibility for learning in the here and now so they can be ready to learn in the future.

We're not making computer science a priority because we expect every student to be a software engineer. We're doing it because we expect every student to be a fully literate citizen of tomorrow's digital world.

// Step-by-step

- [ ] Gather a small team that's ready to make change
- [ ] Set primary goals based on student needs
- [ ] Research similar programs in the region
- [ ] Identify the key stakeholders and potential partners
- [ ] Pick a first project to pilot
- [ ] Gather reference and training materials
- [ ] Implement the pilot project
- [ ] Document the success and learnings of the pilot
- [ ] Apply for funding to expand or try something new
- [ ] Repeat the process starting from the top

// Resources

MIT App Inventor: A visual programming environment that's great for beginners. See more at appinventor.mit.edu/explore

// Time

Organizing: 2–4 weeks to recruit partners and set vision
Planning: 6–10 weeks to develop teaching and learning plan for pilot
Training: 2–3 months of regular PD
Pilot period: 6–24 weeks depending on program

// Costs

Hardware: $1,200 / classroom
Software: $5,000

// People

1 principal (or similar level of leader) to develop and/or endorse the vision

1 curriculum specialist to develop a scope and sequence for the pilot project

3–5 teachers to pilot the project with their students

>>> QUICKSTART:

Games as CS Incentives

Computer science doesn't always come across as a fun and engaging subject to learn. Intermediate Unit 1 used the Code Monkey program to introduce 3rd graders to coding through an interactive experience that turned writing code into a game-like experience that incentivized students to learn by playing.

HACK DEVELOPED BY

Intermediate Unit 1

Intermediate Unit 1 is an education service agency that supports students, parents, educators, administrators, and communities across three counties in southwestern Pennsylvania.
Learn more at **iu1.org**

CONTACT

Alyssa Moore, STEM Curriculum Coordinator
alyssa.moore@iu1.org
Rob Baier, Curriculum Specialist
robert.baier@iu1.org

// Talking Points

Games that teach coding provide an engaging and approachable on-ramp for students who might not think that computer science is for them.

Games are a powerful way to learn problem solving, pattern recognition, abstraction, and algorithm design. All foundations of computational thinking.

Products like Code Monkey bridge the gap between drag and drop, visual coding and actual written coding for elementary school students.

// Step-by-Step

- [ ] Inquire at your local intermediate unit to find additional coaching and mentoring in CS and/or computational thinking
- [ ] Research coding programs to find one that will suit your students' needs and fit your team's capacity
- [ ] Find a webinar or professional development session focused on the program you chose
- [ ] Pick a group of students to pilot the program with and set aside a time to have fun playing games
- [ ] Pick a group of students to pilot the program with and set aside a time to have fun learning CS together in a low-stakes, high-energy setting

// Resources

Code Monkey: A game-like environment where students learn to code in a real programming language. See more at playcodemonkey.com

Code Monkey Implementation Guide: A framework for using Code Monkey with your students. See more at bit.ly/codemonkeyguide

Exploring Computational Thinking: A collection of lesson plans compiled by Google for Education. See more at https://edu.google.com/resources/programs

// Time

Training: 6 hour session, plus routine check-ins with coaches and mentors
Piloting: 8 weeks per unit
Full Course: 2 full semesters

// Costs

Hardware: $1,200 / classroom
Software: $5,000
Training: $800

// People

1 technology specialist to support research, procurement, and setup of selected program

1 teacher to enroll in training and pilot program (technology and math teachers especially)

>>> QUICKSTART:

First Doodles

Sometimes the best option is to start small. Manchester Academic Charter School helped its students start coding by creating their own Google Doodles using Google's free CS First program for schools. Students watched video tutorials and completed interactive modules to practice the basics of coding. Students then used Scratch to design and animate their own version of the famous Google logo (also known as a "Google Doodle").

HACK DEVELOPED BY

Manchester Academic Charter School

An independent charter school serving students from kindergarten through eighth grade on Pittsburgh's Northside. Learn more at **macsk8.org**

CONTACT

Caranina Koloshinsky, Science Teacher
ckoloshinsky@macsk8.org

// Talking Points

When taking your first steps in coding, it's important to keep things simple and fun.

We all rely on Google every day. Wouldn't it be great to help our students learn how it's made?

Google doodles give students the opportunity to learn basic coding by making their own version of something they recognize.

// Step-by-Step

- [ ] Visit csfirst.withgoogle.com and explore the available curriculum
- [ ] Create school Gmail accounts for each student participating in CS First
- [ ] Set up Google CS First Classroom and select which lesson to start with
- [ ] Guide students through built-in tutorials, provide troubleshooting and assistance along the way
- [ ] Showcase student work and gauge their interest to go further

// Resources

CS First: A free set of CS learning resources and step-by-step tutorials. See more at csfirst.withgoogle.com

Lesson Plan: A step-by-step guide to creating your own Google logo. See more at bit.ly/firstdoodles

Scratch: A free block-based coding environment for students aged 8 to 16. See more at scratch.mit.edu

// Time

Account set up: 2–3 hours
Teacher course review: half day
Class time: 2–3 sessions of 45 minutes to 1 hour each

// Costs

Free

// People

1 teacher to set up accounts, select and practice activity, and guide students through tutorial

>>> QUICKSTART:

CS for the Littlest Learners

Carnegie Library of Pittsburgh (CLP) is a resource for digital literacy as much as it is for basic literacy. Alongside story time, they offer their youngest patrons programs like Little Learners Coding Concepts, where pre-schoolers practice early math and thinking skills like sorting and pattern making.

HACK DEVELOPED BY

Carnegie Library of Pittsburgh

A public library system with 19 branches and a system-wide digital literacy initiative. All CS programming is developed and supported by the Digital Learning Lead Librarian and the Library's Office of Programs and Partnerships. Learn more at **carnegielibrary.org**

CONTACT

Kristin Morgan, Digital Learning Lead Librarian
morgank@carnegielibrary.org

// Talking Points

A library's mission is literacy and learning. Literacy in the current era must include computer science.

Offering technology programs at the library provides young people with learning experiences that help prepare them for the future.

As public institutions, libraries have the opportunity to democratize computer science education by making coding skills more accessible to more people young and old.

// Step-by-Step

- ☐ Start a training program to support librarians who want to experiment with technology as a line of service.
- ☐ Develop a set of media mentorship and digital literacy tips to share with parents and educators during the program.
- ☐ Partner with CS experts in your community, just like you would with storytellers or authors, to share their knowledge with your patrons.
- ☐ Pair developmentally appropriate technology with books that feature early math concepts like patterns, repetition, counting, and sorting.

// Resources

Book lists: Librarian-selected STEM reading lists for families and young kids. See more at carnegielibrary.org/stem-booklists

Super Science Kits: Guides for running library programs in CS and STEM. See more at bit.ly/supersciencekits

// Time

Planning: 1 hour to rehearse; 1 hour to prepare hands-on activities

Librarian training: 1 hour research and policy review; 1 hour on technology; 1 hour on higher order thinking concepts

Program period: 4 programs occurring once a week throughout the month

// Costs

Books and resources: $25
Technology: $300

// People

1 children's librarian per location to implement activities

>>> QUICKSTART:

Learning to Teach CS

For students to be engaged in learning CS, their teachers need to be comfortable teaching CS. Carnegie Learning believes that strong, ongoing, mentor-based professional development can help all teachers prepare students for a future in CS, regardless of their content-area expertise. Carnegie Learning emphasizes game-design projects that engage teachers and students in a fun, yet rigorous approach to computer science.

HACK DEVELOPED BY

Carnegie Learning

A leading producer of programs for K–12 and post-secondary schools. Since 2016, Carnegie Learning has provided over 2,000 hours of teacher training for 62 school districts in PA, impacting more than 10,000 students. Learn more at **carnegielearning.com** and **emcp.com**

CONTACT

Nikki Navta,
VP K-12 Computer Science
nnavta@emcp.com

// Talking Points

If we want all of our students to learn CS, we first need to help all of our teachers teach CS.

Teachers are the "lead learners," especially in a subject like CS that's changing all the time.

Building and playing video games satisfies state and national curriculum standards, develops 21st-century skills, and promotes higher-order learning.

// Step-by-Step

- ☐ Recruit a cohort of teachers interested in building their capacity in CS instruction
- ☐ Explore self-directed courses to try out different methods of learning to teach CS
- ☐ Enroll in full-day game-design workshops to go deeper and become a "creative director" in your classroom
- ☐ Commit to earning a 30-hour certificate in CS instruction from CSTA
- ☐ Become a teacher of teachers and help other educators gain fluency with teaching CS

// Resources

Zulama Game Design Program: Carnegie Learning's premiere hands-on, project-based approach to learning CS. See more at emcp.com/applied-learning/zulama

Teaching Computing in Secondary Schools: A practical handbook for teachers. See more at bit.ly/teaching-cs-secondary

Career and Technology Education: A white paper on aligning CS education with college and career readiness.

// Time

Short courses: 16–20 hours
Long courses: 80–120 hours

// Costs

Training packages: $1,500 per year and up

// People

1 curriculum director to prioritize professional development time

1 teacher leader to be the first "lead learner" and coach other teachers

>>> QUICKSTART:

CS Jam Session

It's tough to get teenagers to actually admit they're interested in something. When the Carnegie Library of Pittsburgh asked teens what they would really like to geek out on, making video games was at the top of the list. CLP recruited a tech-savvy educator to design a five-day intensive based on the Global Game Jam model. The session challenged students to tap into their interests and stretch their CS learning to create video games.

HACK DEVELOPED BY

Carnegie Library of Pittsburgh

A public library system with 19 branches and a system-wide digital literacy initiative. The Labs are teen-only spaces at eight CLP locations where young people can learn and be creative with technology. Learn more at **carnegielibrary.org/kids-teens**

CONTACT
Heather Mallak,
Game Jam Coordinator
heather@public-studio.com

// Talking Points

Teens aren't checked out, they're tuned out. If we offered learning opportunities that connected to their passions, they'd tune back in.

Video games might seem frivolous at first, but they are complex systems that involve many layers of programming. Creating video games is a great starting point for teens learning CS.

When you make a learning experience like a game (by putting a time limit on it or making it competitive) you tap into students' innate interests and energy.

// Step-by-Step

- [ ] Ask the teens you serve what interests them, what questions or content they want to explore, what they wish they could do
- [ ] Based on the students' interest, research programs to model your activities on
- [ ] Reach out to similar programs, professionals in the industry, even other teens who have done interesting work in that area or platform
- [ ] Design a sequential learning experience that brings together student interests, outside voices, and an engaging learning challenge
- [ ] Run the intensive, make it special, celebrate what students learn and make

// Resources

Global Game Jam: The international model for hosting game jams to engage new and seasoned designers, coders, and CS learners. See more at globalgamejam.org

Sample Game Jam Deck: A visual agenda to inspire GGJ participants and guide them through the day's exercises. Get the deck at bit.ly/ggjnext

// Time

Program design: 3 months
Outreach and recruitment: 2 months
Intensive prep: 2 weeks
Intensive experience: 4 hours per day for 5 days

// Costs

Student gift cards: $100 each
Lunch & snacks: $300
Program lead & supplies: $2,000

// People

1 teen specialist librarian to manage program
1 program lead to design and lead instruction
1 paid student helper to assist
5 industry partners to coach activities
100s of student contributors across the world

>>> QUICKSTART:

Focus on Fundamentals

To lay the foundation for a K–12 CS curriculum, Cornell School District leveraged free lessons available from code.org, teacher training provided by the Allegheny Intermediate Unit, and partnership with the Computer Science Department at Carnegie Mellon University. They created a weekly CS Fundamentals course for K–5 students that starts with block-based programming as a precursor to writing code.

HACK DEVELOPED BY

Cornell School District

A small suburban public school district just outside of Pittsburgh that serves approximately 700 students in grades K–12. Learn more at **cornellsd.org**

CONTACT

Kristopher Hupp, Director of Technology & Instructional Innovation
khupp@cornellsd.org

// Talking Points

If we want our high school students to be ready for advanced CS, we need to start building CS fundamentals in elementary school.

We don't require students to take biology because we want them all to become biologists. We require biology because it's important that they understand the science of life. The same is true for computer science.

// Step-by-Step

- [ ] Gather teachers and curricular staff to determine a vision and define a set of student impacts
- [ ] Visit neighboring schools with successful CS programs and study their approach
- [ ] Enroll in professional development sessions for teaching early CS concepts and fundamentals
- [ ] Explore free online curriculum sources like code.org and csfirst.withgoogle.com
- [ ] Design a scope and sequence that meets student needs and works within district constraints
- [ ] Pilot instruction through weekly lessons over at least one school year

// Resources

Code.org: A leading resource for CS lesson plans and resources, including a full CS fundamentals curriculum for elementary schools. See more at code.org/student/elementary

micro:bit: A hand-held computer used to engage students with creative coding challenges. See more at microbit.org

Conductivity Lesson: A sample lesson plan for using micro:bits to combine science standards with CS learning. See the lesson at bit.ly/cornell-conductivity

// Time

Design & development: 1 year
Teacher training: Two-week summer intensive
Instructional delivery: 5 hours per week per class

// Cost

Micro:bit devices: $15 each
Code.org lessons: free

// People

1 technology and/or innovation lead to develop vision and support implementation
1 teacher to specialize in elementary CS instruction
1 teacher to partner with CS instructor to integrate classroom lessons

>>> QUICKSTART:

Coding Bootcamp

The Alice Bootcamp is a one-day event where students learn the fundamentals of computer science by creating 3D games and animations using Alice block-based software. More than 50 students can participate in morning workshops followed by self-guided creative time in the afternoon.

HACK DEVELOPED BY

The Alice Project

A project of Carnegie Mellon University's Entertainment Technology Center that supports CS education through software development, research, curriculum materials, and teacher training. Learn more at **alice.org**

CONTACT
Eric Brown, Director
ewbrown@andrew.cmu.edu

// Talking Points

A coding bootcamp is a great way to take students from zero CS experience to creating their own prototype program in a single day without disrupting normal class time.

Bootcamps are fun and high-energy events, they get kids excited about a topic they might not otherwise show interest in.

Bootcamps like this allow kids to work in groups so they can learn from one another and develop communication and collaboration skills.

// Step-by-Step

- ☐ Set up a "challenge" that will motivate students to solve a problem or produce a prototype.
- ☐ Balance instructional time that helps get students going with exploration time that allows them to discover more on their own.
- ☐ Encourage students to bring their own laptop, but be prepared to provide backups with all the required software installed.
- ☐ Set up "help desks" that run in parallel to the agenda so students can get one-on-one help.

// Resources

Alice Challenge Program Guide: Dive into a comprehensive step-by-step guide to running a bootcamp. See more at bit.ly/alice-plan-master

Sample Agenda: Make the best use of available time. See a sample agenda at bit.ly/bootcamp-agenda

Floor plan: Bootcamps aren't for sit-and-get classrooms. See a model floor plan at bit.ly/bootcamp-floorplan

// Time

Planning: 25 hours
Coordination: 25 hours
Outreach: 15 hours
Set up: 4 hours
Execution: 6 hours

// Costs

Venue: donated
Advertising: $350
Printing: $400
Swag: $250
Food: $500

// People

1 leader to design and direct the event
2 helpers to produce and coordinate the event
3–4 instructors to teach at the event
4–6 volunteers to help out at the event

>>> QUICKSTART:

Rethinking the Textbook

Computer science is interactive, so shouldn't the textbook be too? Carnegie Mellon University's Computer Science Academy is a free online textbook and curriculum for high school computer science classes. CS Academy also supports teachers with ongoing professional development as their students learn Python.

HACK DEVELOPED BY

CS Academy

A project in Carnegie Mellon University's School of Computer Science focused on developing a novel, world-class, online, interactive high school computer science curriculum that is entirely free. Learn more at **academy.cs.cmu.edu**

CONTACT
Erin Cawley, Program Manager
ecawley@andrew.cmu.edu

// Talking Points

CS Academy supports teachers who want to learn along with their students.

Students learn best when they have a teacher there helping them through the tough spots. CS Academy sets the frame, but the teacher fills in the picture.

CS Academy is designed and developed by Carnegie Mellon University's School of Computer Science, the #1 ranked CS school in the U.S.

// Step-by-Step

- [ ] Sign up for the CS Academy program
- [ ] Complete required in-person and online teacher training
- [ ] Implement CS Academy scope and sequence
- [ ] Participate in ongoing professional learning community

// Resources

CS Academy Overview: An in-depth look at what CS Academy is and how it works. See more at bit.ly/CSAcademyDeck

CS1 Scope & Sequence: A 16-unit course plan used in CS Academy's Python-focused program for 9th graders. See more at bit.ly/cs-academy-scope

// Time	// Costs	// People
Training: 3–5 day professional development session, plus independent study **Full Course:** 2 full semesters of daily 45-minute lessons	**Free**	**1 principal or curriculum director** to endorse teacher participation **1 teacher** to enroll in training and pilot program (technology and math teachers especially welcome) **1 university champion** to commit to supporting CS K-12 education

>>> QUICKSTART:

Drop-in After-school CS

Beta Builders teaches students in grades 10 to 12 how to code and write software through free, drop-in after-school sessions at four Pittsburgh Public Schools. Software engineers and teachers partner to teach basic computational skills students need to get started with computer science. Small stipends and snacks make this low-commitment first step hard to resist for students.

HACK DEVELOPED BY

Academy Pittsburgh

A 12-week web development bootcamp for adults focused on strengthening the regional entrepreneurial ecosystem while partnering with local non-profits to expand impact. Learn more at **academypgh.com**

CONTACT

Max Dennison, Executive Director
mhbetabuilder@gmail.com

// Talking Points

Giving students access to programs that adults participate in helps students feel like they are being taken seriously.

Learning in teams, together with peers, makes an intimidating subject like computer science more approachable.

Building real websites and software applications makes lessons more real and more exciting.

Students who attend workshops like these have shown increased interest in computer science and higher GPAs.

// Step-by-Step

- ☐ Cultivate a partnership between a coding bootcamp for adults and local schools' existing after-school programs.
- ☐ Figure out your tech: determine what you need and who can provide it—don't expect students to bring their own.
- ☐ Connect content experts with experienced educators to build a program that is both rigorous and engaging for students.
- ☐ Take the time to get to know the students and relate to their interests and life experiences.

// Resources

CSforPGH: Connect to others in the Pittsburgh region and find CS education resources at csforpgh.org

// Time

Recruiting: 1 month
Curriculum Planning: 3 months
Execution: 2 hours per week

// People

1 facilitator to build relationships and funding
1 lead instructor to prepare curriculum
2–3 instructors to help during sessions

>>> QUICKSTART:

Tech Lending Library

Computer science happens on computers. Hardware and software are a necessity, but what if you can't afford to buy equipment for every classroom? The STEAM Lending Library at the Allegheny Intermediate Unit offers sharing model as an alternative. Through the support of the Pittsburgh Penguins Foundation, the library houses more than 2,000 pieces of equipment. AIU offers any teacher in Allegheny County the chance to build their capacity to use teaching technologies in their classroom.

HACK DEVELOPED BY

Allegheny Intermediate Unit

An education service agency that serves 42 school districts surrounding Pittsburgh. AIU provides essential supports and shared services, such as transportation, special education, and professional development.
Learn more at **aiu3.net**

CONTACT

Tyler Samstag,
Director of Instructional Innovation
tyler.samstag@aiu3.net

// Talking Points

Building a lending library will help increase access to tech tools, without breaking the bank.

With the library model, we can stock up on different kinds of tools so that as kids "level up," they can try more advanced hardware and software.

Using a lending library model gives educators more variety, and also incentivizes them to learn new skills and approaches to teaching CS.

// Step-by-Step

- [ ] Survey your likely patrons (both educators and students) to find out what tools they need
- [ ] Conduct research to find the most versatile, reliable, and cost-effective tools to meet patron needs
- [ ] Purchase individual items and assemble thematic kits, containing technology for a classroom set
- [ ] Set up an inventory management system and establish lending protocols
- [ ] Offer hands-on demonstrations and tutorials as professional development for educators interested in borrowing items from the library

// Resources

STEAM Lending Library Inventory: A list of the major items and kits available through the AIU's lending library. See the list at bit.ly/steam-lending-library-inventory

Common Sense Media: A trusted source for reviews of hardware and software for students. Browse their reviews at commonsensemedia.org

Techsoup: A nonprofit technology marketplace for community organizations, schools, and more. Browse their inventory at techsoup.org

// Time

Program start up: 1–3 months
Library orientations: 30 minutes, monthly
Teacher training: 1 day, twice a year
Inventory management: 2 hours monthly
Program development: 1 hour monthly

// Costs

Hardware: $5,000 minimum
Repairs and maintenance: $1,000 annually
Storage and maintenance: Varies, depending on existing capacity

// People

1 project director to design the program and develop the collection

1 inventory assistant to assist director in overseeing inventory and conducting basic maintenance

>>> QUICKSTART:

Volunteers for CS

Closing the opportunity gap is a tall order for any school district. When a large urban system like Pittsburgh Public Schools wants to do it, they look to community partners. Through their participation in Microsoft's TEALS program, PPS recruits volunteers from local industry and CS graduate programs to collaborate with teachers and support high school students learning CS. Over the course of two years of twice-weekly lessons, teachers develop their own capacity to teach CS on their own.

HACK DEVELOPED BY

Pittsburgh Public Schools

A large urban school district serving students and families in the city of Pittsburgh. PPS enrolls approximately 25,000 students Pre-K through 12. The district supports advanced studies through its gifted and talented program. Learn more at **pghschools.org**

CONTACT

Kashif Henderson,
Gifted and Talented Coordinator
khenderson2@pghschools.org

// Talking Points

We might not have the expertise right now, but this is a long-term process. If we're going to commit to it, we need to build our own capacity to teach CS.

The help we need is right here in our backyard. Technology professionals and grad students can help our teachers get ready to teach CS.

Closing the opportunity gap in CS is more about human capital than technology purchases. What if we cultivated our teachers' own talent and enthusiasm to help students learn?

// Step-by-Step

- [ ] Research the requirements of the TEALS program to ensure you and your school qualify
- [ ] Apply to the TEALS program through Microsoft Philanthropies
- [ ] Once accepted, work with TEALS representative to recruit a pool of volunteers
- [ ] Audition potential volunteers through mock lesson activity
- [ ] Volunteers complete summer training program
- [ ] Selected volunteers and teachers co-plan lessons and class projects
- [ ] Volunteers and teachers co-teach 2–4 times per week for two years

// Resources

TEALS Website: A comprehensive introduction to the program. See more at tealsk12.org

Implementation Guide: Details about school requirements, planned learning outcomes, and more. See the guide at bit.ly/teals-guide

School Partnerships: A map of high schools participating in TEALS across the U.S. See more at tealsk12.org/about/#map

// Time

TEALS application process: 2 months
Volunteer recruitment: 3 months
Volunteer training: 3 months
Program implementation: 280 hours across the school year

// Costs

Wireless Internet: Varies, depending on existing infrastructure
Computers and software: Varies, depending on existing infrastructure
Volunteer reimbursement: $500

// People

1 program coordinator to oversee application planning, and implementation
1 IT staffer to support technology set up
1 teacher for every 15 students participating

>>> QUICKSTART:

Leveling Up in CS

Learning to program takes hours and hours of practice, so pacing is important. Avonworth School District took a leveling-up approach. In grades K–2, students participate in digital media sessions once every six days. In grades 3–6, that grows to twice every six days for half the year. By grades 7–9, students are ready for six-week rotations learning Python. This prepares students for college-level study in high school that prepares them to take the AP Computer Science exam.

HACK DEVELOPED BY

Avonworth School District

A small suburban school district about 9 miles north of Pittsburgh. Avonworth educates about 1,800 students in grades K–12. In recent years, the district has embarked on a comprehensive effort to integrate interdisciplinary, project-based learning. Learn more at **avonworth.k12.pa.us**

CONTACT
Dr. Jeff Hadley,
Assistant Superintendent
jhadley@avonworth.k12.pa.us

// Talking Points

You have to crawl before you can walk, and walk before you can run. The same is true for CS. That's why we should start early and go slow to help young learners become familiar with digital media.

To learn complex skills, it takes focused attention. Students in band practice their instruments every day. That's what our middle school students can be doing in CS.

Each time a student levels up, that's an opportunity to reignite their excitement for CS. It's like they're unlocking a new skill or accessing a new world.

// Step-by-Step

- ☐ Research sources for free, age-appropriate CS activities and lessons for each level, elementary through high school
- ☐ Assemble a team of teachers to enroll in professional development in CS at different grade levels
- ☐ Review your schedule to see where regular exposure to digital media and coding can fit in to existing lesson cycles
- ☐ Review the prerequisites for AP Computer Science and other advanced courses to identify key learning goals for middle school grade levels

// Resources

Understanding by Design Template: A model for planning lessons, including in CS. See the template at bit.ly/ubd-lesson

Code Breaker: Approachable courses for engaging and empowering students with CS. Learn to teach coding at codebreaker.teachable.com

CoSpaces Edu: A platform for teaching students CS by building 3D words. Visit cospaces.io/edu for more

// Time

Design and development: 1 to 2 years
Teacher training: Two-week summer intensives, plus on-going trainings and workshops every month
Framework development: 1 semester
Ongoing collaboration and prep: 1 period per day

// People

1 senior administrator to advocate for a district-wide commitment to CS
1 digital media specialist for grades K–2
1 digital media specialist for grades 3–6
1 computer science teacher for middle school
1.5 computer science teachers for high school

>>> QUICKSTART:

Transdisciplinary CS

After nearly two decades of offering CS courses in high school, Canon-McMillan School District expanded computational thinking across its entire K–12 curriculum. Embedding CS within interdisciplinary STEM, STEAM, and Maker activities has been key to making CS work for all students, especially in elementary and middle school grades.

HACK DEVELOPED BY

Canon-McMillan School District

A public school district in Washington County, PA. Canon- McMillan serves approximately 5,355 students in grades K–12. Learn more at **cmsd.k12.pa.us**

CONTACT
Grace Lani, Director of Curriculum & Instruction
lanig@cmsd.k12.pa.us

// Talking Points

To realize the true promise of "computer science for all," we need a district-wide commitment.

Effective CS programs are built on a strong foundation of teacher professional development. Teachers need ongoing support and coaching, as well as adequate time and space to practice teaching CS.

Learning CS helps students unlock knowledge, skills, and dispositions they'll use in every class: collaboration, problem-solving, deep inquiry, and creativity

// Step-by-Step

- ☐ Work with instructional leaders to develop a district-wide continuum of computational thinking.
- ☐ Pilot teaching modules at different grade levels and in different subjects to experiment and learn what works.
- ☐ Partner with a professional development provider to support teacher adoption of new instructional methods.
- ☐ Provide continuous PD and use a scaffolded approach for teachers to learn how to code.
- ☐ Look for ways to use CS and computational thinking as a teaching tool rather than as a separate course.

// Resources

K–12 Continuum: A district-wide scope and sequence for CS. See a template at bit.ly/can-mac-continuum

Code to the Future: A professional development partner. Learn more at codetothefuture.com

Code.org: An online resource full of self-paced lessons and projects for teachers and students. Learn more at code.org

// Time

Curriculum writing: 10–20 hours / course
Teacher training: Ongoing
Lesson planning: 30 mins / day
Classroom projects: 2–8 weeks

// Costs

Curriculum & coaching : $66,000
Hardware: $1,200 / classroom
Software: $5,000

// People

1 district administrator to plan and direct
1–2 CS coaches to provide outside expertise
1 technology director to coordinate integration
1 teacher leader to co-teach with other teachers
3–5 teachers to pilot implementation

>>> QUICKSTART:

Friends in Higher Ed

Schools looking for help developing creative CS experiences for their students can often find partners in higher education. The Entertainment Technology Center (ETC) at Carnegie Mellon University provides professional development for teachers and collaborates in planning and delivering project-based CS lessons in the curriculum. Projects can span a single unit, a full year, or multiple years and range from 2D computer animation to developing 3D video game engines.

HACK DEVELOPED BY

Entertainment Technology Center

A department of Carnegie Mellon University offering a Masters of Entertainment Technology degree combining fine arts and computer science. ETC students work with community partners to create video games, museum exhibits, mobile apps, and more. Learn more at **etc.cmu.edu**

CONTACT

John Balash,
Director of Educational Engagement
jbalash@andrew.cmu.edu

// Talking Points

Instead of trying to keep up with specific tech tools, ETC focuses on helping students learn processes that allow them to fail forward and problem solve.

Creativity entertainment (both using it and making it) can provoke curiosity in students. They are more likely to dig in when something captures their imagination.

The best way to learn CS is through interdisciplinary activities that combine tech skills with other subjects and creative capacities.

// Step-by-Step

- ☐ Gather a group of teachers across multiple disciplines like art, humanities, and technology
- ☐ Identify a teacher or administrator who understands student interest and can act as the lead partner
- ☐ Work together to develop a novel solution that fits your school and your students' needs
- ☐ Develop ideas for creative technology projects based on student interests and school strengths
- ☐ Seek a meeting with outreach staff at a higher education partner to explore opportunities
- ☐ Pilot the project

// Resources

CSforPGH: Connect to others in the Pittsburgh region and find CS education resources at csforpgh.org

// Time	// Costs	// People
Program design: 3 weeks to 3 months **Teacher training:** Varies, at least 1 day **Class time:** Varies, at least 2 hours per week for a semester	**Partnership fee:** $25,000–$50,000 (typically grant supported)	**1 technology and/or innovation lead** to develop vision and build relationships with partners **3–5 teachers** to participate in training and integrate program into class time

>>> QUICKSTART:

Cultivating CS Talent

Simcoach Games designs games for young people, so who better to help them develop and iterate games than local youth? Working with high-school-aged interns throughout the year, Simcoach helps local students develop their CS and STEM skills while gathering valuable youth perspectives that directly inform game design. And all the while, they're cultivating the next generation of CS/STEM talent in the region.

HACK DEVELOPED BY

Simcoach Games

A game design company that creates video games to inspire youth to careers and connect them to training and job opportunities. Based in Pittsburgh's Strip District, Simcoach spun out of Carnegie Mellon University. Learn more at **simcoachgames.com**

CONTACT
Jessica Trybus,
Chief Games Officer
jtrybus@simcoachgames.com

// Talking Points

Young people bring a unique perspective to the business setting that can enrich your offerings and improve your products.

Technology companies have a responsibility to help cultivate and train a diverse future workforce.

Internships at technology companies are a powerful way to expose young people to the possibilities in their futures, and to the knowledge, skills, and dispositions they'll need to succeed.

// Step-by-Step

- [ ] Develop a project that would give student interns real responsibility, like developing a game.
- [ ] Connect with local workforce development organization or summer youth employment program to find summer interns
- [ ] Challenge youth to run their own playtesting sessions to check their ideas and garner feedback from users
- [ ] Create opportunities for interns to present their work along the way. This helps them practice critical soft skills
- [ ] Make clear connections to career pathways that show the diversity of options related to CS
- [ ] Connect interns to potential next steps for pursuing certification or degree programs, or other industry opportunities
- [ ] At the end of each cycle, evaluate internship structure based on feedback from interns and your team and plan improvements

// Resources

Simcoach Games: A growing collection of games that develop career readiness skills and awareness. Browse games at simcoachgames.com/games

Skill Arcade: A platform for learning career skills using gaming technology. Get started at simcoachgames.com/skill-arcade

Learn & Earn: A summer youth employment program that places teens in workplaces. Learn more at partner4work.org/programs/learn-earn

// People

1 point person to manage internship program

Interdisciplinary team to support and mentor youth interns as they work on projects

>>> QUICKSTART:

Seeing a Future in CS

You can't be what you can't see. For many black students, that includes being a computer scientist, as African Americans make up only 8% of the computing workforce. M-PowerHouse seeks to change that by connecting young black students with people of color working in STEM fields. Through a combination of inspirational speaking and hands-on coaching in coding, the program gives young people a role model to identify with and the early skills to get a head start on their CS pathway.

HACK DEVELOPED BY

M-PowerHouse

A nonprofit organization dedicated to connecting under-served youth to career and educational opportunities through life skill exploration, mentoring and hands-on experience in health, engineering, and STEAM learning. Learn more at **m-power-house.org**

CONTACT

Terry Smith, President
info@m-power-house.org

// Talking Points

The underrepresentation of black people in tech should be a national priority. We can take steps now to start shifting the narrative of what's possible.

We keep talking about all these jobs awaiting today's students. We also have to help them see themselves in those jobs. We have to make it real.

Students are more likely to respond to someone who looks like they do, who shares the same cultural identity as they do, who shares the same social experience as they do.

// Step-by-Step

- [ ] Conduct outreach to identify and recruit CS professionals of color in your community
- [ ] Work with these partners to develop a learning experience that is culturally relevant to your students
- [ ] Use a "look, listen, and learn" approach that combines an introductory presentation to familiarize students with the field and terminology, a guest speaker from the industry, and a hands-on coding activity
- [ ] Offer awards and recognitions to incentivize student participation
- [ ] Create a resource packet to guide students looking for follow-on CS learning opportunities in the community

// Cultural Competency Resources

Beyond Plight: A framework for promoting optimal development in black men and boys. See the framework at bit.ly/beyond-plight

Inequities affecting Black Girls in Pittsburgh and Allegheny County: A report about the barriers faced by black girls. See the report at bit.ly/inequities-black-girls

// Time

Outreach and recruitment : 6–12 hours
Program design and coordinating: 8 hours per week
Program implementation: 90 minutes, twice a week

// Costs

Computers and software: Varies, depending on existing infrastructure
Speaker stipend: $500

// People

1 program coordinator to oversee design, development, outreach, and implementation
2–3 guest speakers to share stories and inspire students
1 coding coach to engage students in hands-on learning

>>> QUICKSTART:

Connecting Girls to CS

Women are underrepresented in CS-related fields, so it's important to provide girls with on-ramps to CS and connect them with supportive role models who can help breakdown gender-based barriers. The Technology Leadership Initiative at Pitt launched Tech Divaz to create a girls-only space for CS learning. Over a week-long intensive, an all-female staff teach girls in grades 6 to 9 everything from hardware and operating systems to web design and programming.

HACK DEVELOPED BY

Technology Leadership Initiative

An outreach program of the University of Pittsburgh's School of Computing and Information. TLI works to create a pipeline into computer and information science programs for students from groups traditionally underrepresented in the field. Learn more at **tli.cs.pitt.edu**

CONTACT

Mackenzie Ball,
Director of Outreach
mackenzie@cs.pitt.edu

// Talking Points

Girls have tremendous potential in CS. To let gender discrimination and outdated cultural norms persist is unacceptable.

When students see people they identify with thriving in a field, they're more likely to envision themselves in that field, too. That's why it's important for girls to meet and learn from women in CS.

Community is important. Creating a space for girls to learn CS together is a powerful way to make them feel like this is a subject for them.

// Step-by-Step

- [ ] Assemble a team of women educators interested in teaching girls CS
- [ ] Design a 4–5 day lesson plan in CS principles, including some time spent coding
- [ ] Conduct outreach to girls in your school or program who might be interested in forming a CS club
- [ ] Run a pilot "camp" or intensive that brings the girls together to learn over a condensed period of time.
- [ ] Host a parent symposium to showcase what the girls learned and created
- [ ] Survey girls at the end of the camp to see what they want to learn next

// Resources

TLI Program Overview: An outline of the theory and design of the TLI program. See the overview at bit.ly/tli-program

// Time

Program design: about 5 hours per week for 8 weeks

Recruitment: 5 hours per month

Program implementation: Monday – Friday, 9:00 AM to 3:00 PM

// Costs

3 Instructors: $1,500 each per week

Food for students: $700

Food for parent symposium: $1,000

Class materials: $300

// People

1 program director to design the program and oversee implementation

3 instructors to conduct lessons and mentor participating girls

>>> QUICKSTART:

Team Teaching CS

Computing is a powerful tool to solve problems across the full spectrum of human endeavor. To reveal this to students (and teachers), West Greene School District developed an interdisciplinary program led by a team of three teachers. Students move seamlessly between a mechatronics engineering course, an aquaponics agricultural sciences course, and a computer science course, using what they learn in each field to conduct their own inquiries, experiments, and problem solving.

HACK DEVELOPED BY

West Greene School District

A small, rural public school district in Greene County, southwest of Pittsburgh. West Greene educates about 800 students in grades K—12. The district is one of several in southwestern PA combining CS, STEM, and agricultural sciences. Learn more at **wgsd.org**

CONTACT

Jed Hamberger,
K-12 Academic Director
hambergerj@wgsd.org

// Talking Points

Teachers have ideas. If we can give them the space and time to develop those ideas with their peers, untold learning is possible.

CS can be so abstract that it's hard for students to see how it is relevant. What if we connected CS with courses where they experience more concrete, hands-on learning?

Our lives are interdisciplinary. Shouldn't our learning be, too?

// Step-by-Step

- ☐ Build a team of teachers interested in developing a cross-disciplinary project-based learning program
- ☐ Develop a pilot project that leverages the strengths of the school and the interests of the students
- ☐ Adjust expectations for how students use class time to allow them to work fluidly across courses involved in the project
- ☐ Seek out professional development opportunities for teachers to practice project-based learning
- ☐ Design a class schedule that allows time for teachers to plan and prep together

// Resources

CSforPGH: Connect to others in the Pittsburgh region and find CS education resources at csforpgh.org

// Time

Program design : 6—10 weeks
Teacher training: 2 sessions per month
Lesson planning and coordination: 10 hours
Coursework: 1 semester, 2 hours per day

// People

1 curriculum director to prioritize professional development time and support teacher team

1 principal to design and/or endorse flexible class schedule

3 teachers to design and implement interdisciplinary project-based learning program

>>> QUICKSTART:

CS Internships

For schools that have successfully integrated CS into their curriculum, the next step is building connections to CS careers. Nazareth Prep partners with more than 130 community and corporate internship providers to create unique internship experiences for each student. One such partnership with the game-design startup EDGE teaches students how to design, develop, and market products made with computer science.

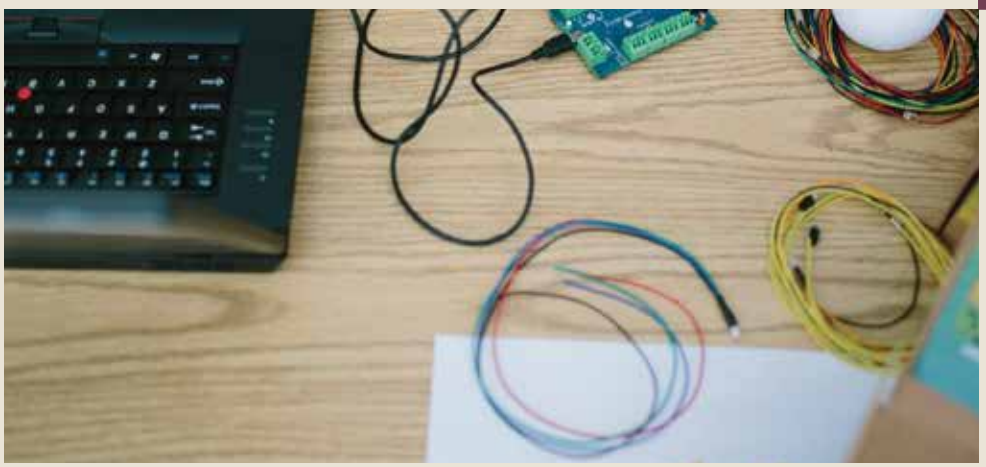
HACK DEVELOPED BY

Nazareth Prep

An independent Catholic school focused on career and college preparation. Every student at Nazareth Prep experiences a rigorous yet personalized program and unique internship experience. Learn more at **nazarethprep.org**

CONTACT

Lisa Abel-Palmieri, Head of School
abel-palmieri.lisa@nazarethprep.org



// Talking Points

Internship experiences help make computer science learning feel more real and relevant to students thinking about their future.

In the next ten years, companies will have huge gaps to fill in computer science and related jobs. Our students can get a foot in the door now through CS internship experiences.

Coursework is essential, but there is nothing like learning on-the-job. That's where theory meets reality and our students can learn by doing.

// Step-by-Step

- ☐ Learn about the gaps in representation in the CS field, especially among students of color and girls
- ☐ Invite companies to offer career awareness workshops, tutoring, job shadowing and internships
- ☐ Create flexible time in the high school schedule to allow for internships or dual enrollment
- ☐ Research companies in your community who may be open to hosting students as interns
- ☐ Research local community colleges that provide Pell grants for students while still in high school
- ☐ Provide students with guidance on finding and selecting internship or dual enrollment in CS

// Resources

CSforPGH: Connect to others in the Pittsburgh region and find CS education resources at csforpgh.org

// Time

Relationship building: 6 months to a year
Lesson planning: 12–15 hours per week
Internship period: 1 day per week, September thru May

// People

1 technology and/or innovation lead to develop vision and build relationships with partners

1 advisor to guide student search and selection of internship experiences

3 teachers to support CS learning and connect intern experiences to coursework

>>> QUICKSTART:

CS for Graduation

Once students are through their first steps in coding, CS can become much more rigorous. In fact it has to, in order to prepare students for advanced studies and eventually careers in CS. To introduce this level of rigor, Northgate School District replaced its traditional Business, Computing, and Information Technology graduation requirements with a course in Computer Science Principles endorsed by the College Board. Students can complete the course as freshmen or sophomores to earn credit toward graduation and also earn community college credits.

HACK DEVELOPED BY

Northgate School District

A small, suburban public school district adjacent to Pittsburgh. Northgate educates about 1,200 students in grades K–12. Learn more at **northgate.k12.pa.us**

CONTACT

Shaun Tomaszewski,
Director of Curriculum and Assessment
stomaszewski@northgate.k12.pa.us

// Talking Points

Traditional computing skills like typing and using word processors aren't sufficient. Our students are capable of so much more than basic computer usage.

Most businesses today involve a good deal of computer work, including advanced systems knowledge and coding. We have to be sure our BCIT courses are enhancing every student's potential for success in the modern job market.

Learning to code is way more interesting than learning to type. If we want to engage the hard-to-reach students, shouldn't we be offering them chances to really learn something new?

// Step-by-Step

- [ ] Research local and state guidelines for courses that can count toward BCIT graduation requirements
- [ ] Advocate to your school board in favor of offering a CS course in addition to or in place of outdated computer skills courses
- [ ] Enroll BCIT-certified teachers in professional development to build their capacity to teach CS
- [ ] Research model high school CS programs and select one that fits your context and capacity
- [ ] Add introductory CS course to course catalog as an option for students to fulfill BCIT graduation requirements

// Resources

AP Computer Science Principles: A model high school course for introducing students to CS essentials See more at apcentral.collegeboard.org/courses/ap-computer-science-principles

Dynamic Learning Project: A coaching program offered by Google to support educators bringing CS into their school. See more at edu.google.com/giving/dynamic-learning-project

// Time

Research and planning: 1 semester
Teacher training: 5 days, minimum
Coursework: 2 hours per day, 3 days per week full academic year

// Costs

Wireless Internet: Varies, depending on existing infrastructure
Computers: Varies, depending on existing infrastructure
Instructional technology coach: $80,000

// People

1 curriculum director to develop plan and advocate to school board

1 instructional technology coach to support teachers

1–2 BCIT-certified teachers to lead instruction

>>> QUICKSTART:

Integrating CS

CS doesn't exist in isolation. To help bring CS into every classroom, Propel Schools pairs CS integration educators with teachers to co-plan and co-teach two projects per school year. Grades K–4 integrate CS and art to spark engagement and develop transferable skills. Grades 5–8 integrate CS with design and engineering to challenge students to think critically and creatively solve real-world problems. In grades 9–12, students can opt to replace one science or math credit with an approved CS course.

HACK DEVELOPED BY

Propel Schools

A network of public charter schools that transforms the lives of children in under-served communities through innovative, student-centered learning. Propel serves 4,000 students at 13 locations in and around Pittsburgh. Learn more at **propelschools.org**

CONTACT
Kristen Golomb,
Director of Innovation and Partnerships
kgolomb@propelschools.org

// Talking Points

It's critical to take a holistic approach when developing and implementing an integration. The goal is to build CS competencies across curriculum, and not create a separate CS classroom.

Integration allows for a fluid model. Teachers can customize learning based on student needs and timing.

Pairing CS with other subjects helps teach design thinking processes and focus on Four Cs of 21st-century learning: Collaboration, Communication, and Critical Thinking and Creativity.

// Step-by-Step

- [ ] Start by examining student needs and evaluating teacher capacities to identify the best subjects to start integrating with CS
- [ ] Hire an integration specialist with a strong background in planning, professional development, and technology to coordinate the effort
- [ ] Pair interested teacher(s) with CS integration educators to create a plan for a pilot project
- [ ] Implement pilot project over multi-week unit, evaluate and adjust before developing wider projects
- [ ] Launch semester-long project

// Resources

Integration Framework: A sample framework for designing subject-specific projects that integrate CS with established curriculum. See the framework at bit.ly/integration-framework

Integration Specialist Job Description: A sample job description for hiring a CS integration specialist. See the job description at bit.ly/integration-job

Integration Lending Library: An inventory of shared technology tools for integrating CS across a school. See the list at bit.ly/integration-lending-library

// Time

Design and development: 1 to 2 years
Teacher training: Full-day trainings twice a month, plus follow up staff sessions
Framework development: 1 semester
Pilot project: 1 semester

// Costs

Technology purchases: $20,000
Integration Specialist: $50,000+

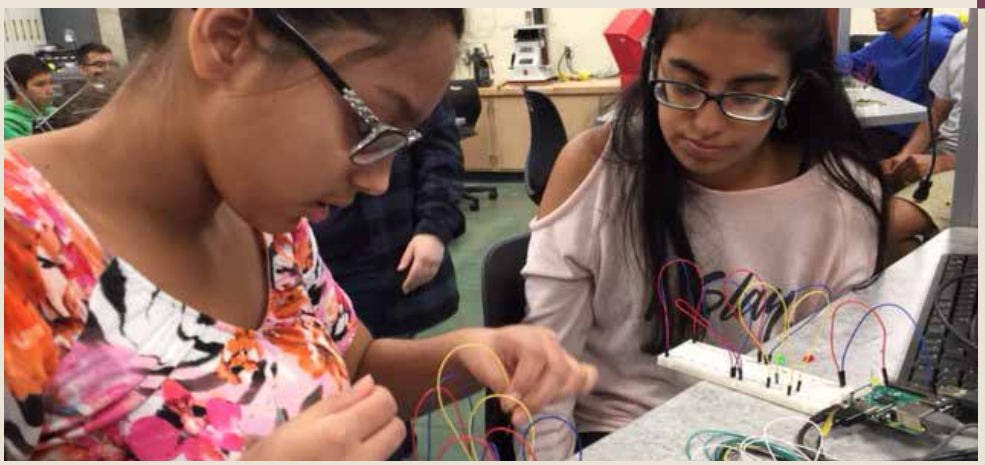
// People

5 teachers to co-plan and co-teach with CS integration educators
5 CS integration educators to co-plan and co-teach with teachers
1 integration specialist to support integration pairing

>>> QUICKSTART:

Scaffolding CS K–12

Computational thinking is an essential literacy. That's why South Fayette Township School District worked to scaffold computational thinking across the K–12 experience. Students learn block-based coding in kindergarten, start programming motors, sensors, and LEDs in elementary grades, learn to create mobile apps and design in CAD in middle school, and make the leap to text-based coding in 8th grade. By the time they reach high school, students are ready for advanced courses in Python, Java, and data science.



HACK DEVELOPED BY

South Fayette Township School District

A public school district about 16 miles southwest of Pittsburgh that serves more than 3,375 students and has been recognized for its comprehensive K–12 computational thinking curriculum. Learn more at **southfayette.org**

CONTACT

Aileen Owens,
Director of Technology and Innovation
amowens@southfayette.org

// Talking Points

Computer science isn't a single subject: it's a whole body of knowledge. That's why we need it to be a standard subject in every grade, just like reading, writing, and math.

We don't learn a language all at once. Before writing an essay, students have to master the alphabet. The same is true for CS.

It's about more than coding. We need to help students develop the essential capacities—the new literacies—they'll need to participate fully in today's world.

// Step-by-Step

- ☐ Cultivate an innovation mindset within your district so that teachers, administrators, and support staff are comfortable with iterative processes
- ☐ Seek input from faculty, staff, board, universities, businesses, and community members to develop a vision for computational thinking in your district
- ☐ Look outside of the district to identify partners who can help, especially other school districts with successful models and professional development providers to support teacher learning
- ☐ Engage students through focus groups and interviews to learn more about their interests in CS
- ☐ Build and nurture a team of teachers and support staff that can lead, develop, and sustain the effort

// Resources

Rethinking Traditional Education: A report by South Fayette on lessons learned from K–12 CS. See the report at bit.ly/rethinking-traditional-education

From Incubator to Launch: A timeline showing how South Fayette's K–12 continuum evolved. See the model timeline at bit.ly/incubator-to-launch

// Time

Design and development: 1 to 2 years

Teacher training: Two-week summer intensives, plus embedded professional development in STEAM classes grades 1-5, and ongoing trainings and workshops throughout the year

// People

1 STEAM teacher focusing on K–2

2 STEAM teachers and 1 technology literacy coach for grades 3-5

2 computer science teachers, 2 innovation studio teachers, 1 technology education/engineering teacher, and **supporting teachers in core curriculum areas** grades 9-12

>>> QUICKSTART:

Making STEAM with CS

While learning to use technology is essential, learning to create technology is far more empowering. That's why Pittsburgh Public Schools weaves computer science throughout the curriculum at its STEAM magnet schools. The district's STEAM Coordinator worked with building principals to design a schedule that ensured each student would have at least one section of CS during every six days. On these days, classes are jam-packed with students excited to practice programming computers and robots.

HACK DEVELOPED BY

Pittsburgh Public Schools

A large urban school district serving students and families in the city of Pittsburgh. PPS enrolls approximately 25,000 students Pre-K thru 12. The district includes 3 STEAM magnet schools. Learn more at **pghschools.org**

CONTACT

Vincent Scotto,
STEAM Coordinator
vscotto1@pghschools.org

// Talking Points

Our students might encounter computers every day, but if we don't create a space for them to learn computer science, they might never realize they have the power to shape the world around them.

Every field, every job, every career is changing. Learning CS is going to be valuable for our students throughout their lives, no matter what they decide to do.

If we really want to get our students engaged and in love with learning, let's get them programming so they can learn to build the future.

// Step-by-Step

- ☐ Develop a unified vision for CS as part of your STEM or STEAM curriculum. Top-down and bottom-up commitment is key.
- ☐ Visit another school that has implemented a CS, STEM, or STEAM program in the past 2–3 years to learn from their recent experience.
- ☐ Enroll teachers in professional development workshops on CS integration with STEM or STEAM.
- ☐ Ensure teachers have the space and time to develop a peer-to-peer professional learning community.

// Resources

Example Lesson: A sample of how one teacher at PPS integrated CS into their coursework. See the report at bit.ly/steam-school-lesson

Buck Institute for Education: A professional development provider that helps teachers and school leaders design project-based learning models. See more at bie.org

Center for Creativity: A program at the AIU offering free training for teachers in STEM, STEAM, and CS. See more at centerforcreativity.net

// Time

Planning: 6 months
Pilot/developmental period: 6 months

// Costs

Wireless Internet: Varies, depending on existing infrastructure
Computers: $750 each
LEGO Robotics Kits: $500 each
Annual maintenance: $5,000

// People

1 STEAM coordinator to oversee development of vision and implementation plan

1 STEAM lead teacher to lead lesson planning and development

2–3 teachers to deliver CS-integrated lessons

>>> WHAT'S YOUR QUICKSTART?

HACK DEVELOPED BY

CONTACT

// Talking Points

// Step-by-Step

- [ ]
- [ ]
- [ ]
- [ ]

// Resources

// Time	// Costs	// People

//CREDITS

A Root + All project
Outreach by LaTrenda Leonard Sherrill
Research by Heather Hopson, Motor Mouth Multimedia
Cover photo by Christopher Sprowls
Additional photography provided by quickstart subjects and
- Ben Filio
- Nico Segall Tobon
- Brian Cohen
- Joey Kennedy

This publication was supported by

BNY MELLON

//SPECIAL THANKS TO

Lisa Abel-Palmieri, Nazareth Prep
John Balash, Entertainment Technology Center
Mackenzie Ball, University of Pittsburgh
Eric Brown, The Alice Project
Erin Cawley, Carnegie Mellon University
Sunanna Chand, Remake Learning
Trisha Craig, Fort Cherry School District
Maximilian Dennison, Academy Pittsburgh
Kristen Golomb, Propel Schools
Jeff Hadley, Avonworth School District
Jed Hamberger, West Greene School District
Kashif Henderson, Pittsburgh Public Schools
Kristopher Hupp, Cornell School District
Caranina Koloshinksy, Manchester Academic Charter School
Grace Lani, Canon-McMillan School District
Heather Mallak, Pittsburgh IGDA
Ani Martinez, Remake Learning
Alyssa Moore, Intermediate Unit 1
Kristin Morgan, Carnegie Library of Pittsburgh
Nikki Navta, EMC School
Aileen Owens, South Fayette Township School District
Tyler Samstag, Allegheny Intermediate Unit
Vince Scotto, Pittsburgh Public Schools
Terry Smith, M-PowerHouse Inc.
Shaun Tomaszewski, Northgate School District
Jessica Trybus, Simcoach Games

//WORKS REFERENCED

Andes, S., Horowitz, M., Helwig, R., Katz, B. (2017) *Capturing the next economy: Pittsburgh's rise as a global innovation city*. Brookings Institution.

Bureau of Labor Statistics. *Occupational Outlook Handbook*. Retrieved from https://www.bls.gov/ooh/computer-and-information-technology/home.htm.

Chung, An-Me. (2018) *The Future: Mo'Time for CS in Out-of-School Time*. CS for ALL. Retrieved from https://medium.com/csforall-stories/the-future-motime-for-cs-in-ost-f29c7716245.

Code.org. *K-12 Computer Science Framework*. Retrieved from https://k12cs.org/.

Code.org. *Support K-12 Computer Science Education in Pennsylvania*. Retrieved from https://code.org/promote/pa.

Google. *Computer Science Learning: Closing the Gap, Rural and Small-Town School Districts*. Retrieved from https://services.google.com/fh/files/misc/computer-science-learning-closing-the-gap-rural-small-town-brief.pdf.

Google. *Diversity Gaps in Computer Science: Exploring the Underrepresentation of Girls, Blacks and Hispanics*. Retrieved from http://services.google.com/fh/files/misc/diversity-gaps-in-computer-science-report.pdf.

Pennsylvania Department of Education. *Fulfilling High School Graduation Requirements with Computer Science Coursework*. Retrieved from https://www.education.pa.gov/Documents/K-12/Computer%20Science%20Guidance.pdf.

Pennsylvania Department of Education. *Computer Science Resolution*. Retrieved from https://www.stateboard.education.pa.gov/Documents/About%20the%20Board/Board%20Actions/2018/Computer%20Science%20Resolution.pdf.

Stanford d. school. *School Retool*. Retrieved from http://schoolretool.org/.

The **CSforPGH Quickstart Guide** is a project of Remake Learning's CSforPGH Working Group.

CSforPGH is a regional collaborative composed of individuals and organizations who seek to equitably equip and empower learners to use computer science, responsibly, as a tool to change their world. We prioritize problem-solving, logical thinking, discovery, and learning from failure as necessities to acquire this essential literacy.

See more at csforpgh.org

a resource of:

remake learning

support for this guide provided by:

BNY MELLON