Dr. Duncan Hull, University of Manchester

Contents

Hello!

Hello, my name is Duncan. I'm a lecturer in the Department of Computer Science at the University of Manchester where I lead the Industrial Experience (IE) program. This elective course has over 100 students every year working for 12 months in industry in the penultimate ("sandwich") year of their degree.

I teach undergraduate courses, supervise tutorials, final year projects and masters projects. I serve as second year tutor, employability tutor, while also serving on the mitigating circumstances committee and exam boards. I'm interested in methods that can deliver outstanding learning and student experience using innovative techniques like vertical tutoring, industrial mentoring, working with local schools, Wikipedia editing and live music.

If you are an employer who would like to recruit a summer intern, placement student or graduate please get in touch. During term time, we highlight opportunities for students in the Wednesday Waggle.

Full stack teaching

Regardless of the age or the stage, I enjoy the challenges of teaching and have taught english, maths, science and engineering to primary & secondary school children (K–12), undergraduates & postgraduates. In 2011, I completed a Postgraduate Certificate in Education (PGCE) at the University of Bath and trained at co-educational non-selective state-funded schools in Swindon[1], Shaftesbury[2] and Stockport[3] before returning to higher education. As well as working in educational institutions around the UK I've been fortunate enough to teach in India, Japan and America too.

Background

Born in Bath, Somerset and raised using a secret West Country recipe, my background is a mixture of Natural Sciences (Plant Sciences, BSc), Computer Science (MSc & PhD) and software engineering. I've worked as a consultant and software developer for various organisations including BBC Monitoring, the Ford Motor Company and the National Health Service (NHS). While working on Apache Taverna and myGrid I completed a PhD at the University of Manchester. This was followed by a postdoc at the Manchester Institute of Biotechnology (MIB) working on the Refine project and a stint as a software engineer on Chemical Entities of Biological Interest (ChEBI) in Cambridge, UK at the European Bioinformatics Institute (ebi.ac.uk).

Tools

This website[4] is published using bookdown and built with R markdown, the R language, gitbook, JabRef, JavaScript, knitr, LaTeX, Pandoc, RStudio,

[1] Greendown Community School, Swindon, Wiltshire https://en.wikipedia.org/wiki/Greendown_Community_School

[2] Shaftesbury School, Shaftesbury, Dorset https://en.wikipedia.org/wiki/Shaftesbury_School

[3] St Anne's RC High School, Stockport, Manchester https://en.wikipedia.org/wiki/St_Anne's_Roman_Catholic_High_School

[4] last updated 2019-11-15

Visual Studio Code and TLC. Thanks to Yihui Xie for the excellent tools and documentation. The source is available on github, but you'll be better off reading the *friendly* manual *Authoring Books and Technical Documents with R Markdown* first. If you're viewing this on an iPhone or iPad, there is known bug with the menu bar at the top of this page which means the menu might not display properly. I could have (should have?) used blogdown and Hugo, but opted for bookdown because it is much ~~less bloated~~ easier to use.

Chapter 1

Students

I teach, mentor, tutor, lecture on and supervise a variety of undergraduate and postgraduate courses. You can find me in the labs, my office and the lecture theatre.

\begin{figure}

{

}

\caption{Question everything, or *Nullius in verba* as the say at the Royal Society} \end{figure}

1.1 All years: debug your CV

- You can drop-in to my weekly one-to-one CV clinics for Computer Science students in LF25 during term-time, Thursdays from, 9.30am to 11am or catch me in LF31 on every Friday between 9am and 11am.
- If you haven't written a CV (two pages), résumé (one page) or LinkedIn profile before, you might find the *Debug your CV* guide useful at git.io/mycv. (Hull, 2019a)
- Get feedback on your CV from as many people as possible, because "given enough eyeballs, all bugs are shallow" (Raymond, 1999)
- Outside of term time, it's best to book an appointment

1.2 First year

- Teaching on First year team projects: COMP101 led by Ulrike Sattler (Sattler, 2019)
- Mentoring one group of six first year students
- Organising first year guest lectures, which mostly run in the second semester, February to May

1.3 Second year

- Teaching on Second year software engineering: COMP23311 led by Suzanne Embury (Embury, 2019)
- Organising the labs for the software engineering mentoring program
- Leading second year tutorials COMP2CARS which focus on wellbeing and working out your next steps.

1.4 Penultimate year

- Leading the course for "with industrial experience" (IE), an elective and intercalated year in industry.
- Visiting students on placement, either face to face or via telecon

1.5 Final year

- Supervising final year educational projects based in secondary schools in Greater Manchester, see coding their future. (Hull, 2019b)

1.6 Masters

- Supervising Master of Science projects in Computer Science and Data Science. (Wickham and Grolemund, 2017) This usually involves various combinations of Wikipedia, Wikdata, SPARQL (DuCharme, 2013) and chatbots. (Sharwood, 2019)

1.7 Extracurricular

\begin{figure}

{

}

\caption{Posing on the BBC Breakfast Sofa with the winning team of the BBC / Barclays University Technology Challenge (UTC) in MediaCityUK, Salford} \end{figure}

- Organising, facilitating and promoting extra-curricular activities such as hackathons (Briscoe and Mulligan, 2014; Warner and Guo, 2017) and edit-a-thons. (Hull, 2017, 2015; Mohammad-Qureshi and Hull, 2019) These usually occur off-timetable, for example Wednesday afternoons, evenings and weekends.

- Served as a judge for studenthack.com and greatunihack.com since 2014. These two hackathons are organised by UniCS, a student-led tech society formed by the merger of HackSoc (computer geekery) and CSSoc (socialising). Many other hackathons exist, they are usually aimed at beginners looking to learn new skills, rather than experts looking to compete. (Briscoe and Mulligan, 2014; Fogarty, 2015)

Chapter 2

Employers

We work with a wide range of employers from the smallest bedroom startup to the worlds largest multi-national corporations, and are always looking for more organisations that can offer our students a stimulating working environment. According to highfliers.co.uk, the University of Manchester is the most targeted University in the UK by the Times Top 100 Graduate Employers. (Birchall, 2019a,b,c) We can still do better, for example by engaging with a more diverse group of employers, especially those in Manchester and the Northern Powerhouse, see git.io/manc. (Hull, 2019c; Davis, 2014b,a; Ovenden, 2019; Wainwright, 2019)

2.1 Recruiting students

If you are recruiting computer scientists and software engineers as a summer interns, placement students or as graduates please get in touch with me or Mabel Yau (careers and placements officer). We typically have around 250 undergraduate students graduating annually, alongside a smaller number of Masters and PhD students. The entry tariff of our students (A* AA including A* in mathematics) is comparable to other leading Computer Science (CS) departments in Russell Group universities as shown in the table below.

Figure 2.1: Any employer recruiting our students is welcome to join our industry club, see details below

Institute	UCAS entry tariff	CS Students per year
University of Cambridge	A* A* A	~100
University College London	A* A* A	~150
University of Manchester	A* A A	~250
Imperial College London	A* A A	~200
University of Oxford	A* A A	~50

A demographic breakdown of our students in Computer Science is show in Figure~2.2. If you are looking to recruit science and engineering students from other disciplines like Physics, Maths, Chemistry, Mechnical, Aerospace & Civil Engineering (MACE), Materials Science and Electrical & Electronic Engineering (EEE) you should talk to: * staff in those departments and/or * the central careers service at careers.manchester.ac.uk

2.2 Careers fairs

Our annual Computer Science careers fair is held in the Kilburn building in autumn, we typically have around 30 employers exhibiting over two days. As space is limited, we are always over-subscribed and are not able to accommodate every employer that our students will be interested in. We give priority to employers that offer internships, placements and graduate roles and have contributed to our community through the activities described on this page. The central careers service also organises:

- the big careers fair in Manchester Central every autumn, see the Big Careers Fair
- a smaller careers fair in Fallowfield Armitage centre in May

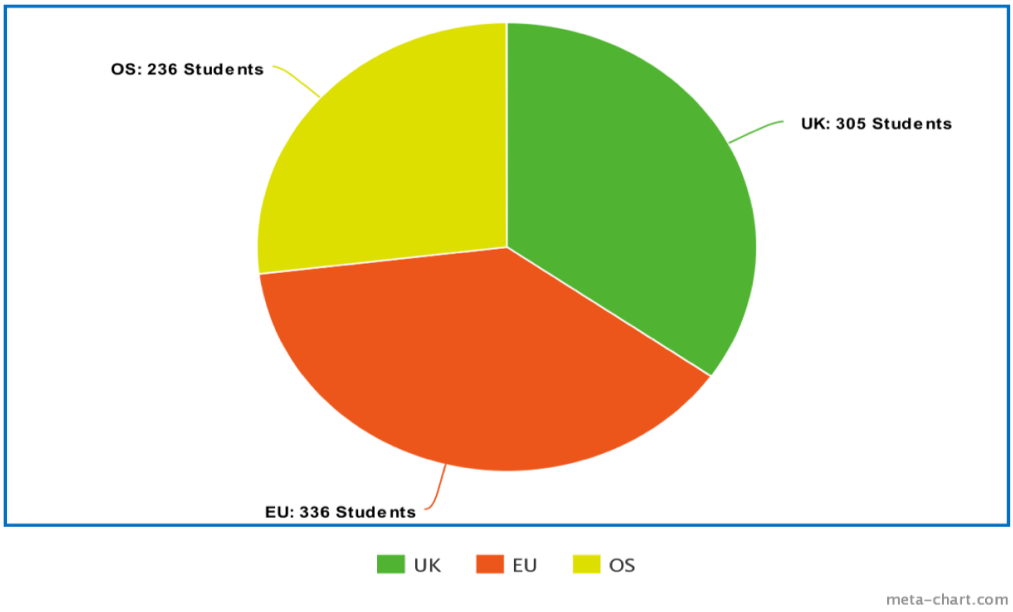

Figure 2.2: Where our undergraduate students come from: As of October 2019 we have 877 undergraduate Computer Science students divided between UK/EU (shown in green), EU not UK (shown in orange) and non-EU overseas (shown in yellow)

foyer to talk to computer science students informally on their way to and from lectures. These usually happen during lunch in term time. If you're interested in exhibiting at either of these events, please contact the careers and placements officer Mabel Yau.

2.4 Industry Club

All employers are welcome to join our industry club mailing list by sending an email to listserv@listserv.manchester.ac.uk with the the text **subscribe cs-industryclub yourfirstname yoursecondname** in the body of the email message. The industry club is part of our wider business engagement activities.

The mailing list is low-traffic, typically two to three updates per year and an invitation to our annual industry club meeting. We promise not to spam you or sell your email details on to third parties.

2.5 Industrial mentoring

The Industrial mentoring scheme for software engineers allows employers meet students during code review sessions.

2.6 The Wednesday Waggle

During term time, we highlight events and vacancies for Computer Science students from a wide range of sources in a weekly newsletter called the Wednesday Waggle . This goes out to around ~1000 Bachelors and Masters students in Manchester each week. If you have vacancies or events you would like our students to know about, get in touch with us or contact the careers service.

2.7 Join the community

There is a thriving community of engineers and entrepreneurs in Manchester and across the North of England. One of the best ways to recruit engineers and scientists is to join and *contribute* to the community. Get involved in events, sponsor a hackathon, deliver a guest lecture, host your own event or become a software engineering mentor. Employers who engage **early and often** are much more likely to get something back. As an employer, you may also be interested in events run by:

- The Institute of Student Employers (ISE)
- The Association of Graduate Careers Advisory Services (AGCAS)
- The Work Based and Placement Learning Association (ASET)

If you're a startup new to employment, you may find the guide at gov.uk/employ-someone useful.

2.8 Buzzing!

At peak times, we can get **very busy** with many concurrent employer events on campus. (Birchall, 2019b) Please be patient and persistent if we do not reply immediately. Unfortunately, we are not always able to respond to everyone because our students, staff and space are all finite resources. We give priority to employers that have already given their time and expertise to our community.

Table 5.8 Universities Targeted by the Largest Number of Top Employers in 2018-2019

		*Ranking in 'Good University Guide' **			*Ranking in 'Good University Guide' **
1.	**Manchester**	*20*	14.	**Exeter**	*12*
2.	**Bristol**	*19*	15.	**Edinburgh**	*28*
3.	**Birmingham**	*14*	16.	**York**	*22*
4.	**Warwick**	*10*	17.	**London** School of Economics	*9*
5.	**London** University College	*8*	18.	**Newcastle**	*21*
6.	**Cambridge**	*1*	19.	**Southampton**	*18*
7.	**Nottingham**	*16*	20.	**London** King's College	*35*
8.	**Durham**	*7*	21.	**Loughborough**	*5*
9.	**Leeds**	*11*	22.	**Glasgow**	*17*
10.	**Oxford**	*2*	23.	**Cardiff**	*32*
11.	**London** Imperial College	*4*	24.	**Liverpool**	*31*
12.	**Sheffield**	*25*	25.	**Leicester**	*38*
13.	**Bath**	*13*			

Source - The Graduate Market in 2019

** The Times & Sunday Times Good University Guide 2019*

Figure 2.3: According to [highfliers.co.uk](https://www.highfliers.co.uk), the University of Manchester is the most targeted University in the UK by the [Times Top 100 Graduate Employers](https://www.top100graduateemployers.com) [@highfliers2019]

2.9 Employability

We are working hard to improve the employability of students because while having a Computer Science is necessary for some jobs, it is not sufficient. (Aaron, 2013; Shadbolt, 2016; Fincher and Finlay, 2016; Fincher et al., 2017) Over the last decade we have been successful in *more than doubling* the number of our students going on year long placements in industry. This is a win-win-win situtation for:

1. **Students**: benefit from a broader education, and develop social and non-cognitive skills that can be challenging to teach and learn in a purely academic environment. This is known as the winning personality (de Vries and Rentfrow, 2016)
2. **Employers**: placements are a cost-effective way for employers to recruit (and retain) graduate talent
3. **Universities**: produce better graduates(Mandilaras, 2004) with broader and deeper skills, who earn more and get better jobs (de Vries and Rentfrow, 2016)

year placement started	No. of Computer Science undergraduate students on placement
2011	42
2012	45
2013	70
2014	54
2015	70
2016	65
2017	100
2018	97
2019	110

In 2019 our students have secured year long placements at Accenture, Agilent Technologies, Amazon (2), AND Digital, Apadmi (5), Arggo, ARM (5), Autodesk, AVL Powertrain, BAML, BBC (2), Biorelate, BJSS,

Bloomberg (2), BMW Mini, Bsquare Controls (2), BT, Cantarus (3), Celtra, CERN (3), Codethink, d3t, Elysian Systems, Feral Interactive (2), Fidelity, FiveAI, HMRC, IBM (2), Imagination Technologies, Intel (4), ISA Software (2), JP Morgan (4), Keysight Technologies, KPMG (1), Matillion (4), McAfee (2), Mentor Graphics (4), Monoprix, Morgan Stanley (2), NCC Group, Nokia, Nomura, Novacoast (2), Ocado (3), PA Consulting, PwC, Schlumberger, ServiceNow, Siemens (3), Soda Software, SteamaCo, The Hut Group (10) The Start Up Factory (2), Uber, Visa and Vodafone.

There's still more we can do to improve the employability of our graduates. If you'd like to help our graduates become more employable, get in touch.

Chapter 3

Research

My research interests are in Computer Science Education (aka Computing Education and pedagogy. (Biggs and Tang, 2011; Fry et al., 2014) I'm interested in methods that can deliver outstanding learning and student experience using innovative techniques like vertical tutoring, industrial mentoring, live music, working with schools, editing Wikipedia and more.

\begin{figure}

{

}

\caption{Too many educational practices are not backed up by good evidence that they actually work. More evidence is needed to support many of the claims made about effective pedagogy. *Wikipedian Protester* cartoon by Randall Munroe at xkcd.com/285 Creative Commons Attribution-NonCommercial 2.5 License} \end{figure}

3.1 SIGCSE

Computer Science has only been taught to undergraduates in the UK for 50 short years (Brackenbury, 2005), so there's lots of open questions about how to teach both the practical and theoretical aspects of the subject. To that end:

- I'm an active member of the Association for Computing Machinery (ACM) Special Interest Group (SIG) in Computer Science Education (SIGCSE.org).
- As part of that I founded and chair a journal club for educators in Manchester, if you'd like to join us, subscribe to the mailing list by emailing listserv@listserv.manchester.ac.uk with the text **subscribe sigcse-journal-club yourfirstname yoursecondname** in the body of your message
- I'm serving on the program committee for Computing Education & Practice (CEP) conference at Durham University in 2020.

3.2 Industrial mentoring

Since we started the Industrial mentoring scheme for software engineers in 2015, more than 1000 students have been through the mentoring scheme with 250 students taking the course every year. We are very grateful for continued support from our industrial partners in making this happen.

Mentors meet with a group of six second year students for two one hour meetings and do some gentle code review of their gitlab repository, as they start to fix bugs and add features to a large open source software project. You don't *need* to be an expert in the tools students are using (Java, Eclipse, Jenkins, Git, JUnit and Ant) it is more about the general process (and politics) of building and testing high quality software in large and distributed teams, than the specifics of the codebase (https://stendhalgame.org/) we happen to be using. Mentors are typically software engineers, both junior and senior.

3.3 Vertical tutoring

We are currently piloting a vertical tutoring (VT) scheme, see vertical tutoring for details. (Barnard, 2010; Drury, 2013)

3.4 Code Club

I lead an after school CodeClub as part of a global network of free coding clubs for 9–13 year olds. (Smith et al., 2014) The aim is to have fun using Scratch, (Resnick et al., 2009) python and other interesting technology we can get our hands on including Raspberry Pi, (Halfacree, 2019) Micro:bits, (Sentance et al., 2017) LEGO® MINDSTORMS®, (Papert, 1980; Klassner and Anderson, 2003) Oculus Rift, Sonic Pi (Aaron et al., 2016) and CodeBug etc.

3.5 Wikipedia and literacy

Wikipedia and wikidata.org (Vrandečić and Krötzsch, 2014; Turki et al., 2019) are powerful tools for improving both digital skills and communication skills, regardless of your age or level of computer literacy, (Proffitt, 2018; Reagle Jr., 2010) particularly in the following areas:

- Literacy generally, the ability to read and write in any natural language. The literacy skills of some engineers and scientists leaves plenty of room for improvement, but literacy has many overlapping dimensions including:
  - Data literacy the ability to read and write (data)
  - Digital literacy the ability to read and write (digitally)
  - Computer literacy the ability to read and write (using a computer)
  - Information literacy the ability to read and write (information)
  - Scientific literacy the ability to read and write (science). How many people do you know who *unashamedly* proclaim their scientific or mathematical illiteracy? (Stacey, 2009; Gowers, 2016; Garner, 2012)

As an experienced and long serving editor of Wikipedia since 2004, I organise and participate in edit-a-thons which recruit and train new Wikipedia editors. Some recent examples include:

1. 2019-10-19 Learn to edit Wikipedia with Ada Lovelace, Sackville Street Building, University of Manchester (Mohammad-Qureshi and Hull, 2019)

2. 2019-10-12 Wikipedia Edit-a-Thon with Zebra Hub HQ, Pankhurst Centre, Manchester
3. 2017-10-13 Physiology Friday, Hodgkin Huxley House, Farringdon, London (Hull, 2017)
4. 2015-09-02 First Wikipedia Science Conference #wikisci, Wellcome Collection, London, NW1 (Hull, 2015; Hodson, 2015)

More information on past and future events like this can be found at:

- wiki-loves-scientists.org.uk
- en.wikipedia.org/wiki/User:Duncan.Hull

3.6 Tuning complete

Tuning complete are a boy band from Manchester, Lancashire . Our current lineup includes:

- Jez Lloyd: Bachelor of Music, DJ and backing vocals
- Steve Furber: six string guitar and bass
- Justin Timberlake: lead vocals, lead dancer
- Duncan Hull: MC, synth / drum machine and embarassing dad dancing (Facepalm, 2015)

We are called *Tuning Complete* because technically speaking we are Turing Complete. (Turing, 1937; Brailsford, 2005) Artistically, this means that what we lack in youth, good looks, fame, fortune, fashion sense, fanbase and back catalogue we compensate for with:

Musical geekery (Fauvel et al., 2006)
Mathematical geekery (Rosenthal, 2005)
Computer geekery (Aaron et al., 2016)

\begin{figure}

{

}

\caption{Tuning Complete consists of Jez Lloyd, Steve Furber, Justin Timberlake and me. We have an agile and constantly evolving membership model which means we are still waiting for Justin to turn up like he *promised* he would in rehearsals} \end{figure}

We played our debut gigs to packed theatres of over 200 second year & first year undergraduate computer science students in the autumn of 2019 and are currently planning future live events while writing a lucrative hit single, working title: #LivingTheDream. If you would like to book our services for your next event, hackathon, wedding, bar mitzvah etc, please contact our agent Mrs. Kilburn shown in Figure~3.1.

3.7 Publications

Informal publications can be found on my sporadically updated blog

- duncan.hull.name/lablog

Figure 3.1: Mrs. Kilburn is our manager, booking agent and promoter. She is the power behind our boy band throne, so all bookings must be approved and scheduled by her office. Please do not approach band members directly with gig requests or offers of marriage, we are all answered for!

Formal peer-reviewed publications can be found on DBLP, ORCID, Google Scholar, the ACM Digital Library, Wikidata etc:

- dblp.org/pid/h/DuncanHull
- wikidata.org/wiki/Q47012855
- orcid.org/0000-0003-2387-503X
- dl.acm.org/author_page.cfm?id=81350580198
- scholar.google.com/citations?user=iDJ-t7IAAAAJ

According to Google scholar, my most cited papers are on:

1. Apache Taverna, published in *Nucleic Acids Research* (Hull et al., 2006)
2. Another Taverna paper, published in *Concurrency and Computation* (Oinn et al., 2006)
3. A paper on modelling human metabolism, published in *Nature Biotechnology* (Thiele et al., 2013)
4. A review of tools for managing large bibliographies, published in *PLOS Computational Biology* (Hull et al., 2008)

Chapter 4

Vertical tutoring

We are currently piloting a vertical tutoring (VT) system for undergraduate students. VT is already widespread in secondary education, (Barnard, 2010; Drury, 2013) but as far as we know has not been widely used in higher education.

Extending the idea of Peer Assisted Study Sessions (PASS) pass.manchester.ac.uk, vertical tutoring creates tutorial groups with a representative from *one of each* year of undergraduate study combined with alumni.

4.1 Full stack mentoring

A vertical tutor group will typically contain five members as shown in Figure~4.1. The group meets physically where possible, or virtually via a slack channel which consists of:

1. One first year student
2. One second year student
3. One penultimate year student (out on industrial placement)
4. One final year student (returned from placement or summer internship)
5. One member of our alumni, recent graduate or via network.manchester.ac.uk

Vertical tutor groups meet twice per semester. It is very unlikely that a free timetable slot for all years and alumni can be found during normal office

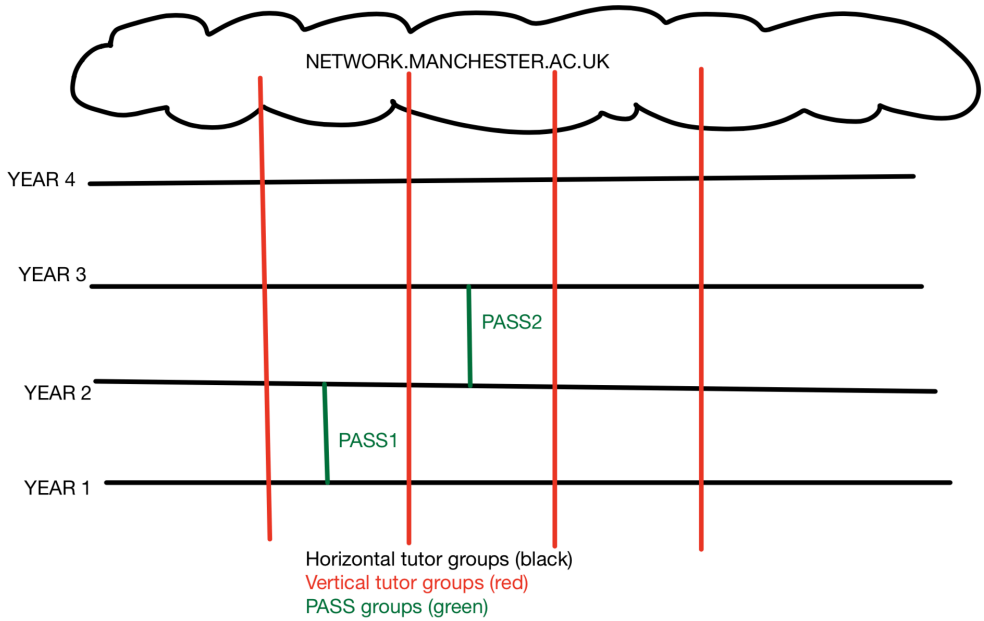

Figure 4.1: Conventional horizontal tutor groups (shown in black) bring together a group of students in the same year. For example, year 1 students meet as a small group once per week during term time with their tutor. Vertical tutor groups (shown in red) are made of of one student from each year and an alumni. Vertical tutor groups extend the idea of PASS, to full stack mentoring, crossing all levels

hours, because of the complexities of timetabling. So evenings will be likely to work best. Where possible, tutor groups will meet face to face, with remote members (e.g. placement students and alumni) typically joining virtually by slack or similar.

4.2 What is it good for?

Vertical tutoring is an attractive idea but does it actually work? If so, how? What is it useful for? We would like to find out:

1. If there is any appetite for vertical tutoring amongst students and alumni
2. How it could work e.g. with slack.com or discord etc?
3. How many times can/should vertical tutor groups meet? Twice per semester? More frequently? Less frequently?
4. What are suitable topics for discussion in a vertical tutorial? Careers, wellbeing, networking etc
5. What kind of specialist groups could be useful e.g. all female groups, research focussed tutorial groups (with MSc & PhD students), ordinary "vanilla" groups etc
6. How much can we breakdown entrenched year groups that persist throughout education (Robinson, 2006)
7. What might the benefits be? (Robinson, 2010)

As this is an experiment, students have been selected on a voluntary basis. If you're a student or former student and would like to get involved, let me know.

4.3 How long will all this take?

We ask that tutees commit to:

- two one hour sessions per semester
- some setup and administration, slack channels, scheduling suitable times and dates with your group

- Two hours of time for feedback and review after each semester, by email survey

Chapter 5

Coding their future

Coding their future is a collaboration & partnership between secondary schools and the Department of Computer Science at the University of Manchester. Our aims are to:

- improve and support Computer Science education at key stages 3, 4 and 5. (Furber, 2012, 2017; Sentance et al., 2018; Swan et al., 2013)
- widen participation in higher education, especially in under-represented groups. (Agnew, 2018; Friedman and Laurison, 2020; Green and Kynaston, 2019; Rajan, 2019)
- enable our undergraduate students to develop their leadership and communication skills

The University provides schools with a final year student who can teach Computer Science in your school or college as a teaching assistant (TA). In return, the school provides our undergraduate students with a safe and supportive environment in which to teach which extends and augments your current curriculum. This can either be an after school, extension / lunchtime club or during scheduled lesson time, typically between year 7 and 13. This is similar to the Undergraduate Ambassador Scheme (UAS), (Singh, 2005; Cooper and D'Inverno, 2005) and school placements (Moller and Powell, 2019) except students work is assessed using our final year project framework. (Morris, 2019a,b) Since these projects were started in 2012, our students have worked with a range of schools in the private and

Figure 5.1: Undergraduate students in computer science regularly work with schools as part of our wider [social responsibility activities](https://www.cs.manchester.ac.uk/connect/social-responsibility/) [@m2020; @m20202] and [schools, colleges and public activities](https://www.cs.manchester.ac.uk/connect/schools-colleges-public/)

public sector, both selective and non-selective, co-educational and single-sex including:

- Fairfield High School for Girls, Droylsden
- Trinity CofE High School, Central Manchester
- University Technical College (UTC) @MediaCityUK, Salford
- Manchester Communication Academy, Harpurhey
- The Barlow RC High School, Didsbury
- Cheadle Hulme High School, (CHHS) Stockport
- Laurus Cheadle Hulme, Stockport
- Knutsford Academy, Knutsford
- Altrincham Grammar School for Girls, Trafford
- Altrincham Grammar School for Boys, Trafford
- Manchester Grammar School (MGS), Fallowfield

The projects were setup by Duncan Hull and David Rydeheard (who retired in 2019), and are now run and supervised by Duncan. We hope to transfer ideas between private and public sector, as there are lots of open questions about how Computer Science should be taught. (Sentance et al., 2018; Sentance, 2018; Stephenson, 2018; Fincher and Petre, 2004) To find out more, see the guidance for teachers and guidance for students below.

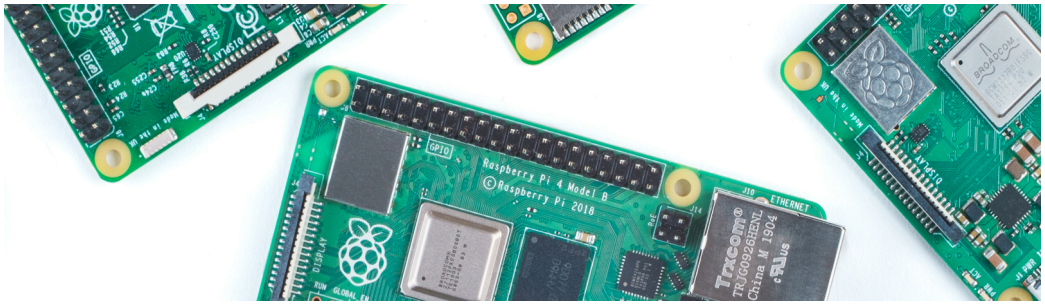

Figure 5.2: An abundance of free software and relatively cheap new hardware like the [Raspberry Pi](https://www.raspberrypi.org) [@raspberrypi], [Microbit](https://microbit.org), [@Sentance2017] [Makey Makey](https://makeymakey.com) [@nevertooold; @makeymakey;] [Crumble Controller](https://redfernelectronics.co.uk/crumble/) and [Arduino](https://www.arduino.cc) [@arduino] has opened up lots of new possibilities for teaching Computer Science. Picture via Alex Bate. [@SnazzyRPi]

5.1 Guidance for teachers

Our aim is to support the teaching and learning of Computer Science in your school and to help engage schoolchildren in the subject. This page describes what we can provide you with and what we expect to get in return.

5.1.1 What the University is offering your school

The University of Manchester will provide your school or college with at least one student ambassador with some relevant training who has completed two years of study in Computer Science and has:

- A good knowledge of, and enthusiasm for Computer Science
- Completed Disclosure and Barring Service (DBS) clearance
- An interest in teaching and working with young people
- Achieved a minimum of a 2:1 or 1st class degree in their second year

5.1.2 What the University expects from your school

In return, we expect that the school provides the undergraduate student with:

- Opportunities to engage with a classroom or after school club of children as a Teaching Assistant (TA). This is typically for around one or two hours during term time. Initially, this could be through classroom observation and teacher assistance, culminating in the student delivering at least one lesson (and potentially a series of lessons) with your support and guidance
- Advice, suggestions, feedback, assessment and encouragement from you to suggest the kinds of resources that would be useful, appropriate or engaging for the Computer Science curriculum you are teaching
- Classroom and behaviour management: the students are not trained teachers and will be relying on your expertise in classroom and behaviour management.

5.1.3 Resources developed by students

Undergraduates typically develop a range of resources. The project will involve development of a computer-based system together with supporting activities, lessons and resources. The resource could be a variety of things including, a game, robotics, animations, hardware (Raspberry Pi, Arduino etc) or software, intended to enthuse school students at one of the Key stages 3 or 4 about fundamental concepts in computing preferably linked to one of the new Computer Science curricula.

5.1.4 Project timing

The projects run for 6 months from September to March, divided into three phases.

1. **September to October** Observation in the classroom teaching by the student around once per week. Development of ideas for an educational tool that the student will make, with the advice of the classroom teacher

Figure 5.3: Lecture theatre 1.1 (LT 1.1) in Kilburn full of first year students

2. **November to January** From November to January, our students develop and tests prototype tool (or tools) with the supporting material, this can happen sooner for students who make a quick start to the project.
3. **February to April** From February to April, our students are expected to liaise closely with teachers to develop an educational tool that will be of use in the classroom using teachers' suggestions as to what is appropriate to build. Students will spend some time in a classroom working closely with teachers and students developing and delivering a new resource for teaching. More details on final year projects can be found in COMP300, the undergraduates already know what is required from their project

5.1.5 Assessment and monitoring

Formal supervision and mentoring is undertaken by the university (Duncan and David), but we will ask you to fill in a one page form on your assessment of their progress during their time at your school, we very much value your input and hope that these projects can beneficial for both your school and the University. We don't want to burden you with unnecessary bureaucracy that all teachers battle with!

5.2 Guidance for students

So why would you, an undergraduate student, want to work on an education project in secondary school? The UK government would like Computer Science should be taught in all secondary schools in the UK. (Furber, 2017) However, in many UK schools there is a shortage of teachers who are trained in Computer Science, consequently, many teachers find themselves being asked to teach a subject they may know little about. (Furber, 2012)

Undergraduate students can make a significant difference here, by supporting teachers in the classroom to create and deliver new classroom resources in Computer Science. (Hull, 2019b) In addition, undergraduate students will have the chance to:

- develop leadership skills in the classroom
- gain valuable experience of working on "real world" problems in a stimulating environment
- improve your communication skills, especially spoken communication work as part of a team (in the school) and join a small group of like-minded undergraduate students (in the University) working on related projects
- test your knowledge & technical ability in a challenging and dynamic environment working with young people
- last, but not least, there is a good chance you will have lots of fun and have a rewarding experience of teaching make yourself more employable by doing all of the above

5.2.1 Who is involved?

Initially, the number of undergraduate students involved in these projects will be less than ten. We also require that you will have a minimum of a 2:1 or 1st in your second year exams. Projects are co-supervised by Duncan with additional supervision from an experienced member of teaching staff at a participating school.

We have carefully selected schools in Manchester that are relatively easy for you to get to, are already teaching Computer Science and have supportive staff and teachers in place to help you. You will be expected to work

directly with school children with the support of the teaching staff in your school. Schools we have worked with are all the Manchester area.

5.2.2 What will the educational projects be expected to deliver?

You will be expected to work closely with the teacher to develop resources that

- engage students with one or more aspects of the new Computer Science curriculum at an appropriate key stage. This is usually key stage 3, key stage 4 or key stage 5 ages 11-18.
- complement **and extend** the schools current provision for computer science in the school

During the project you will be spending a significant amount of time in the classroom, visiting your school every week during school term time throughout the duration of your project to develop resources. These must include a computer-based teaching tool which may use, for example, Raspberry Pi's, visual aids, demonstrations, videos, online questionnaires, formative feedback, games, drones, robotics, music, (Aaron et al., 2016) algorithms (Kubica, 2012) or even just the command line (Smedley, 2019) etc.[1] In addition, guidance on classroom use, such as a lesson or series of lessons to support the tool. Remember that you don't actually need a computer, see Computer Science Unplugged: Computer Science without a Computer. (Bell and Vahrenhold, 2018)

All deliverables for standard final year projects will be expected of these projects including:

- first semester presentation
- demonstration of the resource being used in the classroom
- final written report

[1]Conquer the command line is part of the The MagPi essentials series, there are lots of others like it you may find useful on using the camera module, gaming in python, simple electronics and more at https://store.rpipress.cc

Assessments for these projects will be as for standard projects, (Morris, 2019a,b) but part of the evaluation of the project will be a classroom demonstration, a description and evaluation of which should be included in your final report.

5.2.3 When do the projects start and finish?

Projects start annually in September and are handed at Easter time, see final year project guidelines. For more information contact Duncan Hull.

Chapter 6

Contact

You can contact us using the details below, which include directions and parking information.

\begin{figure}

{

}

\caption{Paying homage to Alan Turing at a mural on the Princess Parkway by tankpetrol.com. Turing is, as Jonathan Swinton puts it, the "patron saint of Manchester" (Swinton, 2019). As a Manchester icon, he is

commemorated locally by the Alan Turing building, the Alan Turing Memorial and Alan Turing Way (Cooksey, 2013)} \end{figure}

6.1 Office

Our offices are in the Kilburn building, close to the Byte cafe, past the Student Support Office (SSO), through the double doors, down the ramp.

Dr. Duncan Hull, Lecturer

- Room LF25, Kilburn Building
- email: duncan.hull ATE manchester.ac.uk
- telephone: +44 161 275 6186
- linkedin.com/in/duncanhull

Mabel Yau, Careers and placements officer

- Room LF26, Kilburn Building
- email: mabel.yau ATE manchester.ac.uk
- telephone: +44 161 275 6140
- linkedin.com/in/mabel-yau

Student Support Office

- Room LF21, Kilburn Building
- email compsci-sso@manchester.ac.uk
- telephone: +44 161 306 8155

6.2 Elsewhere

You can get in touch via t'internet at:

- Slack: search for "Duncan Hull" or my work email
- Skype: search for "duncanhull"
- Blog: duncan.hull.name
- Github: github.com/dullhunk
- Twitter: twitter.com/dullhunk

6.3 Postal address

Send post by snail mail to :

Dr. Duncan Hull
Lecturer
Department of Computer Science
Kilburn Building
The University of Manchester
Oxford Road
Manchester
M13 9PL
Lancashire

6.4 Kilburn building directions

From the train stations, it takes about 20 minutes to walk from Manchester Piccadilly (MAN) and ten minutes from Manchester Oxford Road (MCO). Our official postcode (M13 9PL) takes you to University Place next door, so you're better of using the what3words locations (Leatherdale, 2019) below which are more accurate:

- Google map of the Kilburn building bit.ly/directions-to-kilburn-building
- There are two ground floor entrances to the Kilburn building, North and South
  - North entrance: what3words.com/port.museum.rips
  - South entrance: what3words.com/common.wiping.email
- There is no formal reception so the best place to meet is bit.ly/ByteCafe on the first floor
- See also cs.manchester.ac.uk/about/maps-and-travel/

6.5 Parking

If you are driving, the nearest car parks are:

- **University Car Park B** Manchester Aquatics Centre Car Park, NCP M13 9SS
- **University Car Park D** Booth Street West Car Park, M15 6AR, access via Higher Cambridge Street
- See estates.manchester.ac.uk/services/operationalservices/carparking

Last updated 2019-11-15

Bibliography

Aaron, A. (2013). Why your computer science degree won't get you a job.

Aaron, S., Blackwell, A. F., and Burnard, P. (2016). The development of sonic pi and its use in educational partnerships: Co-creating pedagogies for learning computer programming. *Journal of Music, Technology and Education*, 9(1):75–94.

Agnew, C. (2018). Widening participation: Annual report.

Barnard, P. (2010). *Vertical Tutoring: Notes on school management, learning relationships and school improvement*. Grosvenor House Publishing Limited.

Bell, T. and Vahrenhold, J. (2018). *CS Unplugged—How Is It Used, and Does It Work?*, pages 497–521. Springer International Publishing.

Biggs, J. and Tang, C. (2011). *Teaching for Quality Learning at University*. McGraw-Hill Education Ltd.

Birchall, M. (2019a). Civil service named the UK's number one graduate employer. *The Times Top 100 Graduate employers*.

Birchall, M. (2019b). *The Graduate Market in 2019: Annual review of graduate vacancies & starting salaries at the UK's leading employers*. High Fliers Research Limited.

Birchall, M. (2019c). *The Times Top 100 Graduate Employers 2019-2020*. High Fliers Publications Ltd.

Brackenbury, L. (2005). Digital 60: An interview with linda brackenbury. *BBC Manchester*.

Brailsford, D. (2005). Turing complete: computerphile.

Briscoe, G. and Mulligan, C. (2014). Digital innovation: The hackathon phenomenon.

Cooksey, K. (2013). Alan turing: Manchester celebrates pardoned genius.

Cooper, P. and D'Inverno, R. (2005). The future of the discipline? mathematics and the undergraduate ambassadors scheme. *Journal of Mathematics Teacher Education*, 8(4):329–342.

Davis, E. (2014a). The case for making hebden bridge the UK's second city. *BBC NEWS*.

Davis, E. (2014b). Mind the gap: London v the rest. *BBC TWO*.

de Vries, R. and Rentfrow, J. (2016). A winning personality: The effects of background on personality and earnings.

Drury, E. (2013). A guide to vertical teaching: Advice from experts and teachers who use vertical teaching in their schools. *The Guardian*.

DuCharme, B. (2013). *Learning SPARQL*. O'Reilly UK Ltd.

Embury, S. (2019). Comp23311 software engineering 1.

Facepalm, O. (2015). Dad dancing, the number one way to embarrass children. *The Daily Telegraph*.

Fauvel, J., Flood, R., and Wilson, R., editors (2006). *Music and Mathematics: From Pythagoras to Fractals*. Oxford University Press.

Fincher, S. and Finlay, J. (2016). Computing graduate employability: Sharing practice.

Fincher, S., Finlay, J., and Davies, S. (2017). Building a graduate employability community in computing: the GECCO workshops.

Fincher, S. and Petre, M., editors (2004). *Computer Science Education Research*. Taylor & Francis.

Fogarty, T. (2015). Hackathons are for beginners. *medium.com*.

Friedman, S. and Laurison, D. (2020). *The Class Ceiling: Why it Pays to be Privileged*. Policy Press.

Fry, H., Ketteridge, S., and Marshall, S., editors (2014). *A Handbook for Teaching and Learning in Higher Education*. Taylor & Francis Ltd.

Furber, S. (2012). *Shutdown or restart? The way forward for computing in UK schools*. Royal Society.

Furber, S. (2017). *After the reboot: computing education in UK schools*. Royal Society.

Garner, R. (2012). Almost 50 per cent of adults can't do basic maths (that means half).

Gowers, T. (2016). Maths isn't the problem - the way it's taught is.

Green, F. and Kynaston, D. (2019). *Engines of Privilege: Britain's Private School Problem*. Bloomsbury Publishing.

Halfacree, G. (2019). *The Official Raspberry Pi Beginner's Guide*. Raspberry Pi Press, 3 edition.

Hodson, R. (2015). Wikipedians reach out to academics. *Nature*.

Hull, D. (2015). Improving the troubled relationship between scientists and wikipedia. In *First Wikipedia Science Conference, Wellcome Trust, London*. Figshare.

Hull, D. (2017). Wikipedia at the royal society: The good, the bad and the ugly. In *Physiology Friday: The Physiological Society, Hodgkin Huxley House, London*. Figshare.

Hull, D. (2019a). Debugging your CV, linkedin, job search etc.

Hull, D. (2019b). Getting started with computing education projects.

Hull, D. (2019c). The northern software house: tech employers in the north west #NotJustLondon. *github*.

Hull, D., Pettifer, S. R., and Kell, D. B. (2008). Defrosting the digital library: Bibliographic tools for the next generation web. *PLOS Computational Biology*, 4(10):e1000204.

Hull, D., Wolstencroft, K., Stevens, R. D., Goble, C. A., Pocock, M. R., Li, P., and Oinn, T. (2006). Taverna: a tool for building and running workflows of services. *Nucleic Acids Research*, 34(Web Server):W729–W732.

Klassner, F. and Anderson, S. D. (2003). LEGO mindstorms: Not just for k-12 anymore. *IEEE Robotics & Automation Magazine*, 10(2):12–18.

Kubica, J. (2012). *Computational Fairy Tales*. CreateSpace Independent Publishing Platform.

Leatherdale, D. (2019). What3words: The app that can save your life.

Mandilaras, A. (2004). Industrial placement and degree performance: Evidence from a british higher institution. *International Review of Economics Education*, 3(1):39–51.

Mohammad-Qureshi, S. and Hull, D. (2019). Learn to edit wikipedia: Thursday 17th october, university of manchester. *Wiki Loves Scientists*.

Moller, F. and Powell, S. (2019). Teaching computing via a school placement. In *Proceedings of the 3rd Conference on Computing Education Practice - CEP '19*. ACM Press.

Morris, T. (2019a). Comp30030 third year project laboratory syllabus: Joint honours.

Morris, T. (2019b). Comp30040 third year project laboratory syllabus: Single honours.

Oinn, T., Greenwood, M., Addis, M., Alpdemir, M. N., Ferris, J., Glover, K., Goble, C., Goderis, A., Hull, D., Marvin, D., Li, P., Lord, P., Pocock, M. R., Senger, M., Stevens, R., Wipat, A., and Wroe, C. (2006). Taverna: lessons in creating a workflow environment for the life sciences. *Concurrency and Computation: Practice and Experience*, 18(10):1067–1100.

Ovenden, M. (2019). Manhattan-chester: Unprecedented residential building in manchester is happening - very high up. nine towers of over 25 storeys have appeared in the past three years, so what's fuelling the change? *BBC Radio 4*.

Papert, S. (1980). *Mindstorms: children, computers, and powerful ideas*. Basic Books.

Proffitt, M., editor (2018). *Leveraging Wikipedia: Connecting Communities of Knowledge*. American Library Association.

Rajan, A. (2019). How to break into the elite.

Raymond, E. S. (1999). *The Cathedral & the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. O'Reilly Media.

Reagle Jr., J. M. (2010). *Good Faith Collaboration: The Culture of Wikipedia*. The MIT Press.

Resnick, M., Silverman, B., Kafai, Y., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., and Silver, J. (2009). Scratch: Programming for all. *Communications of the ACM*, 52(11):60.

Robinson, K. (2006). Do schools kill creativity?

Robinson, K. (2010). Changing education paradigms.

Rosenthal, J. (2005). The magical mathematics of music.

Sattler, U. (2019). Comp10120 first year team project.

Sentance, S. (2018). Recent developments in computer science education research.

Sentance, S., Barendsen, E., and Schulte, C., editors (2018). *Computer Science Education: Perspectives on Teaching and Learning in School*. Bloomsbury Academic.

Sentance, S., Waite, J., Hodges, S., MacLeod, E., and Yeomans, L. (2017). Creating cool stuff: Pupils' experience of the BBC micro:bit. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education - SIGCSE '17*. ACM Press.

Shadbolt, N. (2016). An independent review of computer science degree accreditation and graduate employability.

Sharwood, S. (2019). Ibm asks remaining staff to take career advice from HR-bot. *The Register*.

Singh, S. (2005). The undergraduate ambassador scheme (UAS).

Smedley, R. (2019). Conquer the command line: The raspberry pi terminal guide.

Smith, N., Sutcliffe, C., and Sandvik, L. (2014). Code club: bringing programming to UK primary schools through scratch. In *Proceedings of the 45th ACM technical symposium on Computer science education - SIGCSE '14*. ACM Press.

Stacey, A. (2009). How to respond to "i was never much good at maths at school".

Stephenson, C. (2018). Pre-college computer science education: A survey of the field.

Swan, C., Beale, C., Avroutine, I., Hodgson, J., Waller, D., Smith-Nunes, G., Kershaw, J., Saeed, S., Dixon, L., Surrall, A., and Pitt, J. (2013). Cambridge GCSE computing MOOC. Technical report.

Swinton, J. (2019). *Alan Turing's Manchester*. Infang Publishing.

Thiele, I., Swainston, N., Fleming, R. M. T., Hoppe, A., Sahoo, S., Aurich, M. K., Haraldsdottir, H., Mo, M. L., Rolfsson, O., Stobbe, M. D., Thorleifsson, S. G., Agren, R., Bölling, C., Bordel, S., Chavali, A. K., Dobson, P., Dunn, W. B., Endler, L., Hala, D., Hucka, M., Hull, D., Jameson, D., Jamshidi, N., Jonsson, J. J., Juty, N., Keating, S., Nookaew, I., Novère, N. L., Malys, N., Mazein, A., Papin, J. A., Price, N. D., Selkov, E., Sigurdsson, M. I., Simeonidis, E., Sonnenschein, N., Smallbone, K., Sorokin, A., van Beek, J. H. G. M., Weichart, D., Goryanin, I., Nielsen, J., Westerhoff, H. V., Kell, D. B., Mendes, P., and Palsson, B. Ø. (2013). A community-driven global reconstruction of human metabolism. *Nature Biotechnology*, 31(5):419–425.

Turing, A. M. (1937). On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, s2-42(1):230–265.

Turki, H., Shafee, T., Taieb, M. A. H., Aouicha, M. B., Vrandečić, D., Das, D., and Hamdi, H. (2019). Wikidata: A large-scale collaborative ontological medical database. *Journal of Biomedical Informatics*, 99.

Vrandečić, D. and Krötzsch, M. (2014). Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Wainwright, O. (2019). Welcome to manc-hattan: how the city sold its soul for luxury skyscrapers. *The Guardian*.

Warner, J. and Guo, P. J. (2017). Hack.edu: Examining how college hackathons are perceived by student attendees and non-attendees. In *Proceedings of the 2017 ACM Conference on International Computing Education Research - ICER '17*. ACM Press.

Wickham, H. and Grolemund, G. (2017). *R for Data Science*. O'Reilly UK Ltd.