CONSTRUIT 2017

International Conference

MAKING, THINKING AND LEARNING IN THE DIGITAL AGE

PROGRAMME
13-16 JULY 2017

UNIVERSITY OF WARWICK UK
DEPARTMENT OF COMPUTER SCIENCE

Co-funded by the Erasmus+ Programme of the European Union

WARWICK THE UNIVERSITY OF WARWICK

Helix5

THE UNIVERSITY OF EDINBURGH

UNIVERSITAS COMENIANA BRATISLAVENSIS MCMXIX

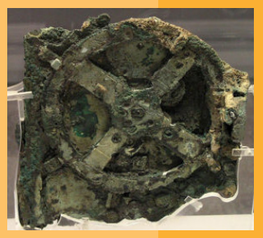
EDUMOTIVA EUROPEAN LAB for EDUCATIONAL TECHNOLOGY

Making, Thinking and Learning in the Digital Age

Learning and making have always been intimately related. Consider, for instance, the roles of the abacus, the sundial, the remarkable Antikythera mechanism, and the astrolabe, in making sense of arithmetic, time and the cosmos. Such artefacts are the precursors of a wide range of instruments and devices in the sciences and humanities that have had a transformative impact on how we construct, and conceive, the world. They include objects associated with new conceptual insights, such as the slide rule that followed the invention of logarithms, specific practical applications, such as Lord Kelvin's machine for tide prediction, and more speculative models (sometimes called 'construals') developed to clarify and communicate provisional understanding of unfamiliar phenomena, such as Faraday devised in his experiments on electromagnetism.

Portrait of Faraday in his thirties

The advent of the computer has had a major impact on the nature of the artefacts that can be constructed to support learning. Open-ended support for exploratory and creative activities is provided by software tools for special-purpose design and management roles, such as spreadsheets, CAD environments and geographical information systems. Computer support for design and making has stimulated new and broader perspectives on software development where the emphasis is on establishing connections in experience, as an instrumentalist does through looking, listening and learning, rather than on abstract computational thinking alone. In this way, computing has revolutionised many practices of the pre-digital age – but perhaps not all. For instance, is it conceivable that computing could have facilitated the making, thinking and learning that led Faraday from his personal, provisional understanding to the construction of the first electric motor?



Part of the Antikythera mechanism

The significance for learning of the thought-processes that accompany writing a computer program was recognised by Seymour Papert in connection with his theory of 'constructionism'. For Papert, a program was first and foremost an 'object-to-think-with' with rich associations that embraced "cultural presence, embedded knowledge, and the possibility for personal identification". He considered many possible ways in which objects-to-think-with might find digital expression: most notably as Logo programs, as microworlds and as educational robots. Further proposals for digital environments to support construction have been based on other programming principles, such as dynamic geometry and spreadsheets, on other languages, such as Scratch and Wolfram, and on other technologies, such as virtual and augmented realities.

Kelvin's tide predictor

CONSTRUIT 2017 is the first conference of its kind. As well as reflecting on the theme of making, thinking and learning from many perspectives, it will introduce a new practice called 'making construals' for developing flexible examples of things-to-think-with. The conference is the culmination of an EU Erasmus+ Project CONSTRUIT! which itself arose out of the long-standing Empirical Modelling Project at the University of Warwick. In both projects the supporting environment is based on a novel, broader approach to computing which is arguably closer to how humans think and learn than classical approaches have been. The vision for the conference is that it be a forum for engaging with a wide range of learning environments, a showcase for some of the achievements of the project, and a launch pad for new CONSTRUIT! initiatives. We welcome everyone with interests in computing and education to participate.

Seymour Papert (1928 – 2016)

CONTENTS

PROGRAMME

Day/Session	Time	Room	Content
THURSDAY – Principles and Practices			
Registration from 12 noon in the Computer Science Atrium			
Lunch 12.30			
Thursday 1: 13/7/2017	13.30 - 14.00	CS1.04	Introduction (Mike Joy, Meurig Beynon, Steve Russ and the CONSTRUIT! Team)
Thursday 2: 13/7/2017	14.00 - 14.50	CS1.04	Matti Tedre (UEF)
			Many Paths to Computational Thinking
			Abstract | Biography
Tea Break			
Thursday 3: 13/7/2017	15.15 - 17.25	CS1.04	Programming and Making Construals
			Construals for computational reasoning (Errol Thompson)
			Hangman in Python (Duncan Maidens)
			In honour of Seymour Papert : “Empirical Modelling” of Logo in Forth (Hans-Joachim Petsche)
			Program traces as things-to-think-with (Andrew Kay)
			Which way to assess computational thinking? (Andrew Csizmadia, Helen Boulton and Gren Ireson)
			An Intended Use Makes Programming Hard (Nick Pope)
Thursday 4: 13/7/2017	17.30 - 18.15	CS1.04	Richard Cartwright (CTO and Founder, StreamPunk Media Ltd)
			The modern software development process is making construals?
			Abstract | Biography

Day/Session	Time	Room	Content
FRIDAY – Open Educational Resources			
			Teachers’ Day
Friday 1: 14/7/2017	09.00 - 10.00	CS1.04	Orientation
			Introduction (Steve Russ) illustrated by the oscilloscope construal (Eemil Tarnanen)
			Daniel K. Schneider (TECFA, University of Geneva)
			Making for Teachers: a New Chance for Simple Physical Learning Objects
			Abstract | Biography
			Coffee Break
Friday 2: 14/7/2017	10.20 - 12.45	CS1.04	Introducing the CONSTRUIT 2017 laboratory (The CONSTRUIT team)
			Workshop Round Robin A
		CS0.01	CAD and 3D Printing (Diane Burton & Margaret Low, WMG / Warwick)
		CS0.03	SHIVA: Virtual sculpting and 3D printing for SEN children (Oleg Fryazinov, NCCA / Bournemouth)
		CS0.06	Art of Programming: Turning Human Solutions into Program Solutions (Dave White, UCL)
		CS0.07	An introduction to the Environment for Making Construals (The CONSTRUIT team)
			Lunch Break
Friday 3: 14/7/2017	13.45 - 15.10		Workshop Round Robin B
		CS0.03	Making construals as part of digital handcrafts at secondary schools: the eCraft2Learn project (Ilkka Jormanainen & Tapani Toivonen, UEF, Finland)
		CS0.01	Internal Combustion Engine (Nick Pope & Steve Russ)
		CS0.06	Generating Sound with JS-EDEN (Elizabeth Hudnott & Jonny Foss)
Friday 4: 14/7/2017	15.40 - 17.00	CS1.04	Making and Teaching
			How to Improve Your Students’ Results with Maker Spaces (Dave Catlin, Valiant Technology)
			Supporting the transition from block to text based programming languages (Andrew Csizmadia and Mark Dorling)
			Interactive Historical Map (Angeliki Theodosi)
			Piloting the JS-Eden environment for “making construals” with Greek teachers (Rene Alimisi, Emmanouil Zoulias and Dimitris Alimisis)
Friday 5: 14/7/2017	17.00 - 18.00	CS1.04	Richard Windle, Mike Taylor (The HELM Team, Nottingham)
			Syngergies between Construals and open educational resources in health
			Abstract | Biography

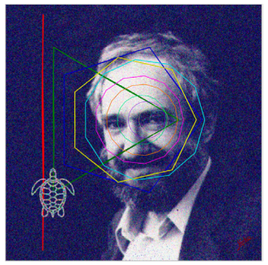
PROGRAMME

Day/Session	Time	Room	Content
SATURDAY – Broader Educational Context ... in parallel with ... The CONSTRUIT 2017 Laboratory*			
Saturday 1: 15/7/2017	09.00 - 10.15	CS1.04	Orientation
			Learning Empirical Modelling: Computational Thinking or Constructionism? (Antony Harfield)
			Charles Crook (LSRI, Nottingham)
			On making a video lecture – and other digital expositions
			Abstract | Biography
			Coffee Break
Saturday 2: 15/7/2017	10.45 - 12.00	CS1.04	Mathematics and Music
			Graphical / textual translation in algebra (Anders Kluge and Ellef Fange Gjellestad)
			Construing and building resilience: a case study of the use of Grid Algebra (Sue Johnston-Wilder)
			Learning to play musical instruments through construal of the Artiphon, a digital musical instrument (Jonny Foss)
			Lunch Break
Saturday 3: 15/7/2017	13.00 - 15.00	CS1.04	Humanities Computing
			Willard McCarty (KCL)
			Construing with the Machine
			Abstract | Biography
			Thinking with Things Made of Metal: A Case Study in Natural Knowledge Construction (John Laudun, Louisiana University)
			Artcasting as a thing-to-think-with: inventive digital education methods (Jen Ross and Hamish Macleod)
			What happens when Creative Technologists bring a Researcher into their team? (Sarah Eagle)
			Tea Break
Saturday 4: 15/7/2017	15.30 - 17.10	CS1.04	Playfulness as Genre in Constructivist Learning (Piet Kommers (Helix5) and Hamish Macleod (University of Edinburgh))
			Designing toys for teaching about systems (Hamish Todd)
			Transmedia Digital Storytelling - Let's build our story! (Vanda Sousa)
			Towards Distant Reading of Translations: Chasing the Sound Wave of Repetitions across Languages in Resistance to the Flat Logic of Text Analysis (Gabriele Salciute Civiliene)
Saturday 5: 15/7/2017	17.15 - 18.00	CS1.04	Frank James (Royal Institution and UCL)
			Construing Humphry Davy
			Abstract | Biography

*The 'Broader Educational Context' programme for Saturday July 16th will run in parallel with a range of 'CONSTRUIT 2017 Laboratory' activities to be organised in the laboratories bordering the Computer Science Atrium. These activities will be oriented towards illustrating and exploring the potential of 'making construals' as a way of developing open educational resources in relation to other approaches. They will include workshops for schoolteachers and pupils, demonstrations of things-to-think-with and construals contributed by delegates and members of the CONSTRUIT! project team, discussions of specalist topics led by project team members and opportunities for open-ended informal interactions with the project team. The resources to be used in CONSTRUIT 2017 Laboratory activities will draw on the contributions listed in the table below, on additional resources proposed by delegates before or during the conference and on other resources identified in the Topic Groups construal that is to be further developed as appropriate in the coming weeks. The nature and scheduling of the CONSTRUIT 2017 Laboratory activities will be set out in broad terms in advance of the conference but will always retain some flexibility to take account of the interests of delegates as they emerge.

The CONSTRUIT 2017 Laboratory – OERs, exhibits, posters, and construals
Workshops for schoolteachers and pupils (Russell Boyatt & Chris Hall) To be given at 10.15-12.00, 13.15-15.00, 15.45-17.30: see further information on schedule, topics and registration
Splinter group discussions on selected topics: see the The Topic Groups construal These sessions willl feature short informal presentations and demos to be organised to promote discussion.
Programs and Construals (10.45 - 12.00)
Comparing the process of creating and using little interactive things-to-think-with (Peter Tomcsanyi & Michal Winczer)
Digital Reality, Design and 3D printing (13.00 - 15.00)
Digital Sculpting: Modelling, Animation and 3D Printing (Valery Adzhiev)
Learning in Digital Reality Environments – Benefits of Virtual Reality and Mixed Reality (Devon Allcoat, Freeha Azmat & Kim Stansfield)
Educational Robotics (15.30 - 17.10)
How to Engage Students by Making Robot Movies (Dave Catlin)
Other resources contributed by delegates for demonstration and discussion:
A digital portrait of Seymour Papert (Tim Smithers)
Experimenting on Regular Polygons in Primary School through programming (Angeliki Theodosi)
Implementation of a trigonometric circle construal combined with physical computing (Emmanouil Zoulias, Rene Alimisi and Dimitris Alimisis)
Representing Digit by Digit Algorithms (Ellef Fange Gjelstad)
The 'oscilloscope' construal (Eemil Tarnanen)
Riga Schools Making, Thinking and Learning in the Digital Age (Lasma Lancmane & Lolita Meza)
Dancing Roamers (Dave Catlin)

Day/Session	Time	Room	Content
SUNDAY – Looking to the Future			
Sunday 1: 16/7/2017	09.00 - 11.00	CS104	Review and Prospects
			Future directions: making construals to evaluate and inform government policy (Karl King)
			Brexit EM Construal (Soha Maad)
			Tree-based Neural Network for Educational Data Mining and Cluster Analysis: Construit Approach (Tapani Toivonen, Ilkka Jormanainen and Markku Tukianen)
			Web-Based Knowledge Sharing Adoption Among Academics in Saudi Arabian Higher Education Institutions (Nouf Almujally & Mike Joy)
			Coffee Break
Sunday 2: 16/7/2017	11.30 - 13.00	CS104	Closing session
Lunch			



Painting With a Turtle to See More

A computational painting made to celebrate and remember the life of Seymour Papert.

Synopsis

Seymour Papert was born on February 29, 1928, and died on July 31, 2016. He was a visionary, humanist, mathematician, pioneer computer scientist, and one of the first to demonstrate and promote the use of computers and programming to teach children (and thus also adults) how to do mathematics, how to learn to think for themselves, and how to use their imagination in creative and constructive ways.

He was the Father of Logo, the programming language I use to make all my paintings, including, of course, this painting. It is my small way to celebrate and to remember someone who made many contributions to so many people's lives.

Papert showed children how to have fun with a Turtle that can be made to draw things. I've used to a little of this here, with what I call the Prime Polygons, the first eleven. Things that Turtles are good at drawing. For this painting the Turtle paints these polygons, with a thin fine brush.

There's lots of mathematics in this painting, more than a million points (each with a hole in the middle to let some colouring through), many finely painted lines, but there's nothing random here.

Afriko

Donostia, October 2016

Painting:
Work 62.02:
Painting With a Turtle to See More

Size: 980mm x 980mm

Programmed using ACSLogo
developed by Alan Smith
running under OS 10.6
on an Apple MacBook Pro

Printed by
Reproducciones Igara S. Coop
Donostia / San Sebastián
Using an Epson Stylus Pro 7800
(at 2,880 x 1,440 dpi)
With Epson UltraCrome K3 Ink
On EFI Cert Proof Paper 6225XF
Semimatt (260 g/m^2)
And technical help from
Juan Mari

Email: `afriko@cantab.net`

Computational Paintings
Works by
Afriko

KEYNOTE SPEAKERS

Matti Tedre (UEF)
Many Paths to Computational Thinking

Matti Tedre is a Professor at the School of Computing, University of Eastern Finland. He is the author of *The Science of Computing: Shaping a Discipline* (Taylor and Francis, 2014). He has held various professorships in five countries, and is currently working on the philosophy of computer science, ICT4D, computer science education, and educational technology.

Abstract

Computational thinking (CT) is a popular phrase that refers to computing's disciplinary ways of thinking and practicing. We acquire those ideas and habits of mind through our work in designing software, simulations, and systems, and they offer powerful mental tools for people who design computations. Although many central ideas of CT are centuries old, a conception of CT as a new way of thinking started to emerge only after the birth of modern computing. CT started to gain wider currency in the academia after the emergence of computational sciences. It also enabled thoroughly empiricist visions for learning and knowledge construction, yet those ideas were not broadly embraced until the 2000s, when a new wave of CT spawned initiatives and educational reforms across the education sector in many countries. This talk presents an overview of the historical currents from which CT has developed as well as some challenges to and misconceptions of CT.

Richard Cartwright (CTO and Founder, StreamPunk Media Ltd)
The modern software development process is making construals?

Dr Richard Cartwright is CTO and founder of Streampunk Media, an open-source startup company based in Scotland. Richard has a PhD in Computer Science from the University of Warwick, where he researched the topic of Empirical Modelling applied to Computer Aided Design. Now a software architect in the media and entertainment industry, he has previously worked at the BBC, Snell Advanced Media / Quantel and as Technical Steering Committee Chair of the Advanced Media Workflow Association (AMWA.tv). Recognised as a go-to expert on, and designer of, highly-scalable, software-only systems, Richard believes it is essential to write code every day. Using code fragments as a model to communicate, he advocates the use of techniques including functional programming, object-orientation, serverless architecture, agile methodologies, and more.

Abstract

Computer science follows a course of simplifying the development of business logic through ever increasing abstractions from the underlying hardware. Creating abstraction layers enables domain experts to optimise mappings from logical to physical platforms. Beyond languages / compilers, networking protocols, the world wide web and virtualisation / cloud, the artefacts of modern software development are adapting to an emerging world of serverless computing and the Internet of Things (IoT). This talk presents observations of modern software development and design – a expanding universe of executable and/or animated boxes and lines – where asynchronous distributed system configuration is replacing imperative, sequential programs. With examples from Amazon Web Services, IBM's Node-RED flow-based graphical programming language and more, this talk illustrates how developers can both experiment with and communicate their ideas through models in web browsers and/or fragments of code in Read-Eval-Print Loops (REPLs). Effectively, is this a process of building construals that either fail fast or - with the addition of tests and a "dev ops" process - get deployed live?

Daniel K. Schneider (TECFA, University of Geneva) Making for Teachers: a New Chance for Simple Physical Learning Objects

Daniel K. Schneider is an associate professor at TECFA, a research and teaching unit in the Faculty of Psychology and Educational Sciences, University of Geneva, Switzerland. Holding a PhD in political science, he has been working in educational technology since 1988 and participated in various innovative pedagogical and technological projects. He has been a prime mover towards the introduction of creative pedagogical strategies and ICT technologies. His long-term R&D interests focus on modular, flexible and open Internet architectures supporting rich and effective educational designs. His current interests include digital design and fabrication in education, student monitoring, e-learning competence and learning in citizen science. Within TECFA's "blended" master program in educational technology (MALTT), he teaches educational information and communication systems, digital design and fabrication, foundations of educational technology, and research methodology. His personal homepage is at http://tecfa.unige.ch/DKSand he coordinates EduTechWiki (http://edutechwiki.unige.ch).

Abstract

This contribution considers how digital manufacturing technologies such as 3D printing and laser cutting could contribute to enrich teacher-made teaching and learning tools. Until now, "making" technologies have been presented and studied as a medium for learning activities or as a subject to be taught but not as a means for teachers to create tangible objects for learners. We shall discuss the experience of two project-based courses where participants made "constructive kits" and "tools for group activities" with a 3D printer and a laser cutter. Based on these field trials we shall examine affordances of these technologies for teacher-created materials. We also will discuss how teachers could be trained and whether "making for teaching" has the potential to transform existing teaching practice.

Richard Windle, Mike Taylor (The HELM Team, Nottingham) Synergies between Construals and open educational resources in health

Dr. Richard Windle is an Associate Professor for Health E-learning at the School of Health Sciences at the University of Nottingham. He co-leads a large research and development team called "HELM: health e-learning and media". The HELM team has been developing and releasing open educational multimedia content for over 15 years and have released over 250 resources in that time. These are being used in over 50 countries worldwide across a range of institutions and contexts. The research of the team, currently rated as having impact of international significance, focuses on the pedagogical qualities and effectiveness of community-derived learning content and is supported by funding from research councils, third sector partners, charities and the EU amongst others.

Mike Taylor is a Learning Technologist and Media Developer and is currently working for the School of Health Sciences at the University of Nottingham. Mike has an MSc in Information Technology and has worked in various development roles within the education sector since the late 1990's. He has been involved in the design and development of Re-usable Learning Objects (RLOs) for over the past ten years. In September 2016, HELM won the UK learning technologist team of the year award, voted for by the UK learning technology community.

Abstract

Reusable Learning objects (RLOs) are small, standalone multimedia learning resources that are designed to be usable in a range of different contexts. We have been developing and openly releasing such resources for nearly 20 years, focusing on the area of health. They are currently being used in over 50 countries worldwide (http://www.nottingham.ac.uk/helmopen/). As well as the concept of open education, there are many other similarities and synergies between the concepts of Construals and RLOs. The media within RLOs are used to cover very focused aspects of a subject, but function as a means to challenge and extend thinking around a particular theme. At the heart of their development is the concept of co-creation and engagement with a community of practice around a particular subject. For us, this is often a way to engage with different stakeholder groups, such as patients or service users, enabling them to share their experiences and expertise. The concept of "making as learning" often involves engaging the intended learner-groups in the development process also. Moreover, the RLOs themselves are often focused around a particular artefact, such as a video, talking head or piece of artwork that the author may use as a vehicle to tell their story. This concept of narratives also links learning of concepts to everyday experience, the virtual world with reality, these concepts are also in common with Construals. Finally, like Construals, the end products are interactive in nature and engage the learner in an active learning process. In this presentation we will demonstrate some of our RLOs, discuss the development process and explore the links with Construals. We will challenge the participants to consider how they might utilise this format in their sphere of learning.

Charles Crook (LSRI, Nottingham)
On making a video lecture – and other digital expositions

Charles Crook was formerly Director of the Nottingham Learning Sciences Research Institute and Editor of the Journal of Computer-Assisted Learning. He is a developmental psychologist by background but has longstanding research interests around new media in educational contexts – with particular attention to how new technology can support and extend the social experience of learning. Currently he is also Visiting Professor at the University of Bristol and East China Normal University.

Abstract

This lecture starts with some reflections on lecturing. It then moves to illustrate the different designs whereby a traditional lecture can be rendered in video format. And, finally, considers the challenges that such video re-mediation presents to a learner's educational experience. These concerns will be a launching point for a more general discussion of video-based exposition and, in particular, how voice and image are most effectively integrated when designing expository artefacts. Examples of both commonplace and neglected design formats will be critically presented for general discussion.

Willard McCarty (KCL)
Construing *with* the Machine

Willard McCarty is Professor of Humanities Computing, King's College London, has adjunct positions at Western Sydney and North Carolina State universities and is Fellow of the Royal Anthropological Institute (London). He is Editor of the journal Interdisciplinary Science Reviews (2008-) and founding Editor of the online seminar Humanist (1987-). He is recipient of the Canadian Award for Outstanding Achievement, Computing in the Arts and Humanities (2005), the Richard W. Lyman Award, Rockefeller Foundation (2006) and the Roberto Busa Award, Alliance of Digital Humanities Organizations (2013). He is author of Humanities Computing (2014/2005), the first comprehensive theoretical study of the field. His current book project is an historical and philosophical study of the relation between computing and the human sciences. See www.mccarty.org.uk/.

Abstract

Writing about 'artificial life', Marilyn Strathern once observed that such phrases "depend on a balance between the synthesis that produces a novel entity and the analytical differentiation of elements without which the combination would not be visible" (1992: 2). Much the same could be said of 'artificial intelligence', which without the sense of a balancing act does little more than name a technical enterprise whose achievements quickly become part of the furniture. Roger Shank and Lawrence Birnbaum have noted that deep questions of intelligence implied by its name "are rarely part of the intellectual debate about the possibilities for AI... [and] are often not part of the debate about human capabilities either" (2009/1994: 79). As Donna Haraway writes (with reference to Strathern), "It matters what thoughts think thoughts... what knowledges know knowledges... what relations relate relations... what worlds world worlds" (2016: 35). What thoughts, then, what knowledges, relations and worlds follow from the yoking together (Samuel Johnson would have said, a violent one) of 'artificial' and 'intelligence'? How do we stay aware of and develop its semantic balancing act? How do we avoid "Single vision & Newtons sleep" (Blake1802)? The beginning I take is to imagine as thoroughly and concretely as possible, with as much attention to the engineering and mathematics as someone severely undereducated in those subjects can give it, what a fully realised AI would be like. I see two ways to go from there. The first follows the well-trodden lead of Alex Garland's *Ex Machina* (2014), Charlie Brooker's *Black Mirror* (2011-) and a host of others into Masahiro Mori's "uncanny valley" (Mori 2012/1970; Kageki 2012). There we find opportunities to ask rich historical and philosophical questions. The other, less trodden path is to treat our imagined creature anthropologically, as an other faced by the other that is us, as a digital native, and then set about *with* it (or with *whom*) to investigate David Gooding's *construals*, his "flexible, quasi-linguistic messengers between the perceptual and the conceptual" (1986: 208). This is the tack I take here. Avoiding all manner of prognostication, I do not speculate on where this tack might take us, whether to a destination we aim for, or to some other. Rather I consider the grand challenges and hard questions it raises, and ask for help with them.

===

Blake, William. Letter to Thomas Butt, 22 November 1802.

Gooding, David. 1986. "How do scientists reach agreement about novel observations?" *Studies in History and Philosophy of Science*. Part A 17 (2): 205-30.

Haraway, Donna J. 2016. "Tentacular Thinking: Anthropocene, Capitalocene, Chthulucene". In *Staying with the Trouble: Making Kin in the Chthulucene*. 30-57. Durham NC: Duke University Press

Frank James (Royal Institution and UCL) Construing Humphry Davy

Frank James's main research concentrates on the physical sciences in the late eighteenth and nineteenth centuries and how they relate to other areas of society and culture, for example art, business, media, religion, technology and the military. He edited the *Correspondence of Michael Faraday*, published in six volumes between 1991 and 2012, and a number of essay collections including '*The Common Purposes of Life*' – a set of essays on the Royal Institution. His *Michael Faraday: A Very Short Introduction* was published in 2010 by OUP who the following year also published his sesquicentenary edition of Faraday's *Chemical History of a Candle*. His current research is on the practical work of Humphry Davy, including his work on nitrous oxide, agricultural chemistry, mineralogy, the miners' safety lamp, analysis of ancient Roman pigments and his attempts to unroll chemically the papyri excavated from Herculaneum.

Abstract

This talk will commence with recollecting David Gooding and his work, particularly how his ideas concerning construals emerged out of his historical work on Michael Faraday, especially in regard to replication. The second part of the talk will look at how construals might be of use in interpreting the work of Humphry Davy, using his discovery of the physiological effects of nitrous oxide.

THURSDAY

Principles and Practices

Construals for computational reasoning

Errol Thompson
Aston University, United Kingdom
e.l.thompson1@aston.ac.uk

Keywords
construal
computational reasoning
game strategies
variation theory

Abstract
The basis of the construal work conducted by Beynon and colleagues is the application of a constructionist understanding of learning. The learner constructs a model that aids them in understanding the phenomena being studied. A key component of Beynon et al's approach is the use of observables so that the learner can see the changes in what they are modelling. The question being asked in this paper is where does such building begin? Does it come through building the model or experimenting with its operation? For a module on problem solving and computational reasoning, we have been using a range of strategy games. The focus however isn't being able to play the game but understanding how to construct strategies to play the games based on computational constructs and ideas.

Variation theory argues that we should expose the variations in the critical aspects of the phenomenon being studied so that the learner is able to observe the features of the phenomenon. Applying the principles of variation theory to computational reasoning means trying to expose the critical aspects of the computational techniques. Combining this with those of using observables in construals, this paper explores how we might construct construals that enable the learner to develop computational based game playing strategies. To illustrate these issues, we will use two games that have predictable outcomes for those knowledgeable about the games. These are the game Nim and tic-tac-toe (OXO games).

The objective of using Nim is to expose the ideas of invariants as they apply in developing computational algorithms. Since the playing strategy can be exposed quickly and predictions of the game outcome made from the current state of the game, we endeavor to encourage the learner to think about how they might express a rule for the state of the game that can be used to determine the next game action. Through modifying the rules of the game additional variation is added to the game to see whether adaption of the computational strategy can be applied in more complex variants of the game.

For tic-tac-toe (OXO games), the goal is to explore alternative computational techniques for describing game play strategies. These include issues of game state trees, rule based approaches, and concepts from test-driven development and behavior-driven development. How does what is exposed impact the way the learner understands and interprets the game playing strategy?

From exploring these two games, we seek to draw some conclusions about what makes a good construal and the features that are appropriate. We also seek to explore what should be observable and how this impacts the variations that we should use to explore a phenomenon such as computational reasoning.

Hangman in Python

Duncan Maidens
Birmingham City University
duncan.maidens@bcu.ac.uk

Keywords
coding
computational thinking
Python
Hangman

Abstract
With coding on the National curricula for school, there is a hive of activity in teaching coding and teaching teachers to teach coding. The range of abilities and variation in prior knowledge is making this teaching a real challenge.

Whilst many believe it is a particular language and the syntax that they struggle with, it is the author's opinion that it is the fundamentals of computation thinking where the problems lie.

The session will look at addressing this issue by tackling a simple hangman game and discuss strategies to bring out the fundamental concepts needed.

In honour of Seymour Papert : “Empirical Modelling” of Logo in Forth

Hans-Joachim Petsche
Department of Philosophy, Potsdam University, Germany
petsche@gmx.de

Keywords

Papert
Logo
Forth
Empirical Modelling
program language translation

Abstract

Forth is nice and flexible but to a philosopher Logo is the more impressive language. When I first read about Logo in the late eighties I was very fascinated by it, however I could only get Forth for my PC. Both are relatives of Lisp, but Forth has a reverse Polish notation (RPN) where Logo has an infix notation. Logo allows top down programming, Forth only bottom up. Logo enables recursive programming, Forth does not. Logo includes a turtle graphic, Forth has nothing comparable. So what to do if you can’t get Logo and have no information about its inner architecture? This should be a case of “empirical modelling”. How can you model the observable results of the behavior of Logo in terms of Forth?

Two recursive programs – one for drawing a binary tree and one for symbolic differentiation – should serve as yardstick for measuring the success of rebuilding Logo in Forth.

The first problem was to replace the RPN by a infix notation. For this purpose we had to separate data from operations by means of a second stack and find the agency which would do that. We found two kinds of agents: arithmetic and logic operators which will push themselves on the operation stack and all kinds of brackets (including “Enter”) which will start to collect the operators and end with the execution of the operations. This agency with a second stack gives Forth the desired Logo-like infix notation.

Now we had to teach Forth the top down programming. Therefore we needed to redefine the word “abort” which gives an error-message if we try to compile an unknown word. The redefined “abort” will put all unknown words in a temporary dictionary. Additionally we observed the need for some other agents “create” and “:” (which start compiling new words) as well as “variable” and “constant” (which hold data). These new agents perform their normal functions but also have the additional ability to test if the new word is already in the temporary dictionary. If so, the temporary word is replaced by the final. If some words remain undefined after compiling we get an error message.

The last significant problem was to implement recursive programming. To enable this we created a new type of ‘agent’: recursive variables. This new type of variable – which does not exist in Logo – has its own stack for its values depending on the depth of recursion. If you call such a variable it looks at its recursion pointer and selects the corresponding value. So the variable is an agent to manage its own recursive values.

Modelling the rest of the Logo functions caused no serious problems. A test shows that our program works in a Logo-like input-output manner.

This was the first and only implementation of a Logo-like program in the former GDR.

Program traces as things-to-think-with

Andrew Kay
Aston University
kaya3@aston.ac.uk

Keywords

computer science
programming
education
learning theory
constructionism

Abstract

This work proposes a suite of new learning activities for computer programming students, supported by educational theory. The proposed activities use a new educational software tool to record, review and manipulate program traces. A prototype of this tool, named CESL, is also presented.

When using the tool, students interact with a visual drag-and-drop environment to perform program operations such as assigning values to variables, creating objects and arrays, and calling methods. The tool responds to these interactions by visualising the effects on the program state, and printing the snippets of code corresponding to each operation.

The sequence of program states and code snippets is recorded by the tool, and constitutes a program trace. Unlike conventional tracing tools such as debuggers, traces are created directly by the student rather than by executing pre-existing code.

These traces can be used as constructionist \things-to-think-with” in further learning activities; for example, traces could be reviewed and replayed, or compared with others recorded by solving dierent instances of the same problem. Branching, loops and recursion each correspond with structural properties of program traces, which could be identied to assist the abstraction of actions into algorithms.

It would also be possible to use student-recorded traces as test-cases for student-written code. The intermediate program states of a conventional execution trace could be compared with those from a trace recorded by the tool; where they diverge, the line of code responsible can be identied, and the expected behaviour presented, along with the code snippet achieving this behaviour as an example.

As traces recorded using the tool are embodiments of actions performed by the student, the ability of students to remember these actions is expected to reduce the cognitive load of the proposed learning activities, compared with using only traces recorded by program execution.

The tool is expected to enable students to learn concrete programming concepts without dependence on syntax, to develop familiarity with the allowed operations for each type of entity, and to practice composing program behaviours into algorithmic plans.

This submission to the CONSTRUIT 2017 conference is accompanied by a video demonstration of the prototype tool, which can be found online at http://andrewkay.name/construit2017.

Which way to assess computational thinking?

Andrew Csizmadia[1], Helen Boulton[2], Gren Ireson[2]

[1]Newman University, UK
[2]Nottingham Trent University, UK
a.p.csizmadia@newman.ac.uk, helen.boulton@ntu.ac.uk, gre.ireson@ntu.ac.uk

Keywords

Computational thinking
Framework
Asessment

Abstract

Computational thinking occupies a pivotal position at the heart of the computing curriculum that is taught in both primary and secondary state schools in England. It is proposed to play an equally pivotal role in the computer science curriculums that have been or are to be implemented in other countries throughout the world. A systematic review of the literature associated with computational thinking clearly indicates that there is little agreement amongst the computer science education community about what computational thinking encompasses, and there is a great deal of discussion and debate about strategies for assessing computational thinking.

The first part of the paper will critically examine the continuing evolution of the definition of the term computational thinking looking back beyond Wing's seminal article (2006) to identify the roots of computational thinking. Then it identifies the definitions of computational thinking that have flowered since 2006. The second part of the paper will present the computational thinking frameworks that have been theorised. Each framework will be introduced and their key dimensions highlighted: concepts, practices, perspectives and participants. The third part of the paper outlines the approaches to assessing computational thinking, which include project portfolio analysis and digital artefact analysis both using automated tools utilising lint-like analysis, artefact-based interviews and design scenarios. The paper concludes with suggestions for formatively assessing computational thinking for both learners and teachers.

An Intended Use Makes Programming Hard

Nicolas Pope
University of Warwick
n.w.pope@warwick.ac.uk

Keywords

construals
programming
adaptation
reuse
structure

Abstract

Much emphasis within the CONSTRUIT! project is placed on the idea that it is not programming but something complementary and more about understanding. Considering the Construit scripting language as a programming paradigm, however, produces several helpful insights into problems that are still faced in the software world and can clarify why Construit is claimed to be different.

One key quality is that at the point of writing a Construit script there is no need for an intended use of the script beyond that moment, in stark contrast to traditional programming. How does Construit manage this? Programmers writing code cannot always know how it will be used, and yet despite this they assume they do because the tools they have require it. Structured programming, object-orientation, encapsulation and all related techniques involve designing intended use into the software and using this to organise the program into modules with fixed interfaces and to organise the collaboration of the development team. Programmers love this just as mathematicians love a beautiful formula and physicists love an elegant theory. It is the artistic appreciation of a beautiful abstraction and lures us into a false sense of security with brittle results.

Embedded structure requires an intended use, so if there can be no reliable intended use in advance there can be no embedded structural solution to organising the programming activity (without managing refactor hell). Where does this leave us? No objects? No procedures? Or inventing an intended use? That latter seems popular, but this paper explores viable options used by Construit.

If most elements of an unstructured state are declaratively given by formulae, then a dependency analysis of these formulae allows structure to be extracted dynamically at a later date. The Construit implementation already does this to automatically identify class and object patterns in order to optimise the underlying JavaScript implementation (which does something similar also). Further, a concept called scoping allows a user of another piece of script to abstract it and ask questions, which in effect (and internally literally) creates functions dynamically at the point-of-use. The original developer can focus on one concrete definition at a time and yet, without realising, they are allowing arbitrary functions and data-structures to be generalised from it in a highly adaptable and reusable way.

So could this be a holy grail of design-less programming at large scale? Not just yet, but the task is not a hopeless one and, as shown in this paper, going back to the world of unstructured programming may have huge benefits

FRIDAY

Open Educational Resources

Computer Aided Design and 3D printing

Diane Burton and Margaret Low
Warwick Manufacturing Group, University of Warwick
d.s.burton@warwick.ac.uk, m.j.low@warwick.ac.uk

Keywords

computer aided design
3D printing
numeracy skills
geometry
Key Stage 3

Abstract

This workshop draws on our experience of working with Key Stage 3 children in local schools over the past four years using the teaching of computer aided design (CAD) and 3D printing as a vehicle to develop key skills and confidence, particularly in those children from more disadvantaged backgrounds.

The workshop will introduce some of the teaching resources for CAD which were developed for use in schools. They include video tutorials, tutorial sheets, measuring tools and modelling materials.

The evaluation of these resources indicated that the children involved felt they had developed their numeracy skills because they had used numbers and geometry in a practical situation. The children indicated that they had gained more awareness of how they learned things, and how a number of different media could be used together to help them to learn.

Our teaching resources are freely available online.

SHIVA project and beyond: accessible virtual sculpting and 3D printing

Oleg Fryazinov, Leigh Mcloughlin, Alexander Pasko, Michelle Wu, Valery Adzhiev
Bournemouth University, UK
ofryazinov@bournemouth.ac.uk, lmcloughlin@bournemouth.ac.uk, apasko@bournemouth.ac.uk, mwu@bournemouth.ac.uk, vadzhiev@bournemouth.ac.uk

Keywords

Accessibility
3D modelling
3D printing
disability
SHIVA

Abstract

Involvement of people with disabilities in the artistic activities, especially those requiring physical actions, is a difficult task. Digital technology provides such an opportunity but only to some extent, as the majority of the current accessible solutions focus more on the access rather than on the content.

In this work we discuss the accessible solutions developed in the SHIVA project that has produced prototype accessible virtual sculpting tools and 3D printing for children with complex disabilities to allow them to express themselves creatively. The easy to use accessible modelling software tools allowed for the creation of sophisticated 3D objects which then were digitally fabricated by using 3D printing technology. Eventually, this allowed students with disabilities to create physical sculptures even without being able to use their own physical movements. Even though the SHIVA project was successfully deployed in a special school in the UK, technical barriers were identified during the trialling period which becomes a part of further research and development into accessible 3D modelling and accessible 3D printing solutions.

We will present in action the software that uses assistive technologies such as eye gaze tracking and a touch screen monitor to create 3D sculpting shapes which then can be brought to life as physical artefacts using 3D printing.

Art of Programming: Turning Human Solutions into Program Solutions

Dave White
University College London, UK
d.white@cs.ucl.ac.uk

Keywords

Graphical programming language
text-based programming language
transition process
computational thinking
unplugged activity
Computer Science Education
Scratch
Logo
Python

Abstract

This workshop introduces Learning, Thinking and Programming in ISPY as an activity appropriate for key stages KS2 – KS5.

Finding and experimenting with different human solutions to a problem sometimes provides a basis for, or serves as a pointer to, devising and implementing algorithms for programmed solutions. In this workshop, we introduce the fertile problem domains of simple geometry of regular shapes in the 2-D plane, and the Knight's Tour on different boards in which to investigate/practise this idea.

For Papert, a program was first and foremost an 'object-to-think-with'... to inspire what comes next... ISPY is such an environment: to try out human solutions for drawing in 2 Dimensions and converting them to program solutions. Using turtle/sprite basic motion instructions with repetition and tailored user-defined shape functions by pressing buttons, the beginner can think about and learn to produce program solutions. In ISPY there is no drag and stick, no IDE, no text input, no syntax errors or messages, no correct, save and run cycle. Just press and go. For more experienced users, kt-app is an interactive board for experimenting with human solutions for the simple knight's tour to spark possible program solutions.

Introducing the Environment for Making Construals

The CONSTRUIT! team

Keywords

the Construit environment
making construals
observables
dependencies
agents

Abstract

This workshop will introduce the web-based Construit environment for making construals, guiding you around the interface and the script language used, along with a demonstration by example of our key concepts of observable, dependency and agency. Some simple exercises will illustrate the basic skills used in developing open educational resources such as the Arduino, Internal Combustion Engine and Artiphon construals that will feature in follow-up workshops.

Making construals as part of digital handcrafts at secondary schools: perspectives from the eCraft2Learn project

Ilkka Jormanainen and Tapani Toivonen
University of Eastern Finland, Joensuu, Finland

Keywords

maker movement
Arduino
physical computing

Abstract

The new Construit environment makes it easy to use physical computing extensions with construals. In this presentation, we will walk through an example where an Arduino microcontroller is used to control sensors and actuators as part of a construal. The examples are driven by the spirit of the H2020 funded eCraft2Learn project (http://project.ecraft2learn.eu/), where possibilities of the maker movement are explored in the context of formal and informal secondary school education.

An Internal Combustion Engine construal

Nicolas Pope
University of Warwick
n.w.pope@warwick.ac.uk

Keywords
construal
the Construit environment
internal combustion engine
open educational resource
engineering

Abstract
This workshop illustrates how a construal of an internal combustion engine can be developed for use as an open educational resource. Observables and dependencies in the model are used to express the relationships between key engine parameters, such as the cylinder and piston size, and connect the circular motion of the wheel with the linear motion of the cylinder. The principles behind the combustion engine are exposed by considering how the wheels can be driven by manual interventions to change the pressure in the combustion chamber. Automating the agency in the construal gives insight into the engine cycle, the role of friction and concepts such as 'engine braking' and 'stalling'.

Making Music in the Construit environment

Elizabeth Hudnott and Jonny Foss
University of Warwick
lizzy.hudnott@gmail.com, jonathan.foss@warwick.ac.uk

Keywords
construal
the Construit environment
music
musical instrument
Artiphon
Theremin
piano

Abstract
This workshop illustrates how a musical instrument can be construed as a means to create a dependency between physical gestures and sounds. This principle is exemplified in construals of several different types of musical instrument: such as the piano, the Theremin and the Artiphon – and in a simple 'DIY' instrument that can be readily developed in the Construit environment.

How to Improve Your Students' Results with Maker Spaces

Dave Catlin
Valiant Technology Ltd, United Kingdom
dave@valiant-technology.com

Keywords
Automaton
Mechanical Circus
Fleet School
Maker Project
Maker Movement
Maker Space
Design Technology
Design and Make
Constructionism

Abstract
An account of how to create an excellent educational environment with Maker spaces.

If you search for definitions for Maker Space you'll find words and phrases like enthusiasm, shared interest and technology. It represents a belief in the value of tinkering. Society has always respected the tinkerer and education longs to bring their spirit to the classroom. Is it possible? How would you do it? This paper is an account of how Trevor Thomson, a teacher at Fleet school in rural England, succeeded in achieving these goals. It records how he worked with his 11-year old students to create a spectacular mechanical circus. More importantly, how he created a Maker Space which helped deliver the whole curriculum, improve student test scores without teaching to test or suffocating the Maker Space spirit of adventure.

Supporting the transition from block to text based programming languages

Andrew Csizmadia[1] and Mark Dorling[2]
[1]Newman University, UK
[2]Queen's Mary University of London, UK
a.p.csizmadia@newman.ac.uk, mail@markdorling.net

Keywords

Computational Thinking
Transition
Pedagogical approaches

Abstract

This talk will help attendees to answer four burning questions facing most teachers teaching computer science at both primary and secondary education:

What are the key skills that novice programmers need to master to become creative, independent and resilient programmers? How do we effectively scaffold our support for learners in transitioning from block-to-text based languages e.g. From Scratch to Snap and then Python, in secondary education? How do you teach learners of all ages to program effectively, and what pedagogies can we reuse from other subjects? What academic research literature should I begin reading to inform my practice?

In this interactive talk, attendees will develop a solid interpretation of what programming constructs are expected to be taught to learners in each educational setting (primary and secondary); identifying the transition points from block to text-based languages e.g. from Scratch to Snap and then to Python. The most relevant academic literature and experience will be shared with attendees in easy to understand byte sized chunks, so they are aware of the foundations that we build upon.

This will be illustrated by sharing tried and tested activities/games adapted from other subjects to support the development of the 13 programming skills needed by learners to become creative, independent and resilient programmers. The presenters will demonstrate to attendees how these activities can be mapped to the most appropriate computational thinking concepts (i.e. abstraction, decomposition, algorithmic thinking, pattern recognition, evaluation, generalisation) to indicate which computational thinking concepts concept learners are having an opportunity to demonstrate mastery of. These activities will be classified around our idea of construct, change and create.

Interactive Historical Map

Angeliki D. Theodosi
Ralleia Experimental Primary Schools
atheo@sch.gr

Keywords

STEAM
Programming
Primary School
Robotics
History
Scratch
Makey-Makey

Abstract

The term STEAM usually describes the connection between Art and Technology in Education. What if the project is about History and Literature rather than Mathematics? How may different subjects of the curriculum in Primary School form an interesting interdisciplinary project in order to engage pupils into learning and deepen their knowledge of the individual themes? The presented interdisciplinary project has taken place during the academic year 2016-2017 in Ralleia Experimental Primary Schools in Greece.Twenty (20) pupils of 5th and 6th grade were participated to create an Interactive Historical Map of a battle (The Sacred town of Messolonghi). Historical events were studied from pupils' book (Koliopoulos J., Michailides J., Kallianiotis A., Minaoglu Ch.). In addition, pupils found extra information in encyclopedias and on the Internet about parallel connected and/or triggered by the sengie of Messolongi events, such as the battle of Kleesova. Pupils processed the sources and they've created themselves the material used on their Interactive Historical Map. The referenced Historical Map was actually a painting of P. Zografos (1836-1839) represented the siege of Mesolongi (1825-1826) based on the descriptions of General I. Makrigiannis (Ioanni Makrygianni-Panagioti Zografou Images of the Independence Struggle, 2009). Pupils have reproduced the painting, wired it with copper tape, connected it with Makey-Makey board (Makey-Makey) and programmed it with Scratch (Scratch (scratch.mit.edu)) in order to create an interactive map. Pupils were totally engaged in the project and expressed their excitement. Furthermore, this project was presented in a teachers workshop where through the interview the participated teachers evaluated the project as a useful teaching strategy.

References

Ioanni Makrygianni-Panagioti Zografou Images of the Strugle. (2009, 07 10). Argolikos Archival Library History and Culture. Argos, Greece. Retrieved 06 29, 2017, from goo.gl/P2PuE9
Koliopoulos J., Michailides J., Kallianiotis A., Minaoglu Ch. (n.d.). History, 6th grade. Athens, Greece. Retrieved from http://ebooks.edu.gr/modules/ebook/show.php/DSDIM-F114/744/4870,22240/
Makey-Makey. (n.d.). Retrieved 03 22, 2017, from http://www.makeymakey.com/
(n.d.). Scratch (scratch.mit.edu).
The Sacred town of Messolonghi. (n.d.). Retrieved 6 29, 2017, from http://messolonghi.elea.gr/uk-page.htm

Piloting the Construit environment for “making construals” with Greek teachers

Rene Alimisi, Emmanouil Zoulias and Dimitris Alimisis
Edumotiva, Greece
ralimisi@gmail.com, ezoulias@teemail.gr, alimisis@otenet.gr

Keywords
making construals
teacher training
evaluation
school education

Abstract
Construit (also known as JS-Eden) is an environment for learners to build software artefacts that relies on construction by ‘making construals’ using observables and dependencies (Beynon 2012, Harfield et al., 2016). Construit is proposed as an alternative to procedural or object-oriented constructionist environments. In this paper we present a small scale case study that was carried out in three stages over a three years period in the context of the Construit! Erasmus+ project.

As part of the CONSTRUIT! Erasmus+ project, three workshops were organised in Athens, Greece, with the aim to familiarize teachers with the construal making process, to test the viability of teaching using Construit in the classroom and the pedagogical value of the Construit environment.

Construit was introduced to 55 secondary and vocational school teachers through four seminars for making construals. The three workshop-based seminars aimed at familiarizing teachers with no prior experience of Construit with the construal making concepts, the environment for making construals (Construit) as well as with the kind of artifacts they could create with Construit.

At the beginning of the workshops, some time was spent to explain the basic theoretical foundations; the workshops continued with demonstrations of already developed construals which could be parameterised by the teachers; emphasis was put on explaining the main components of Construit; then, some time was allocated to hands-on activities; the workshop concluded with an open discussion where the teachers were encouraged to talk about their experience and to express their opinions on the applicability of construal making in their teaching practices. After the seminar the teachers were also kindly asked to provide us with written feedback through an online questionnaire that included both closed and open questions.

Given the small sample, secure conclusions cannot be easily drawn. However, the feedback gathered shows that the CONSTRUIT! training seminars immersed teachers to an explorative and reflective computing process. The feature of the “immediate feedback” embodied in the environment was also highlighted by the teachers as well as the opportunity for visualization, customization and parameterisation. Some teachers appear to be skeptical about the applicability of construal making in their teaching practices while others are more positive and indicate the ways according to which it can be better supported towards this goal. They also highlight the need for additional supporting resources both on technical issues (i.e. user guides, manuals, repository enrichment, construals linked to the school curricula) as well as on pedagogical ones (i.e. lesson plans, worksheets). Although their feedback suggests that there is still room for improvements to be made to Construit’s user interface in order to meet school students’ needs, the principles of constructionism by making construals can be readily applied in a classroom for domain learning. Teachers’ feedback provided has informed significantly the development of several Construit versions and led to the final refined Construit environment.

SATURDAY

Broader Educational Context

... in parallel with ...

The CONSTRUIT 2017 Laboratory

Learning Empirical Modelling: Constructionism or Computational Thinking?

Antony Harfield
Department of Computer Science & IT
Faculty of Science, Naresuan University
Phitsanulok, Thailand 65000
antonyh@nu.ac.th

Keywords

Empirical Modelling
constructionism
computational thinking

Abstract

Computational Thinking is a skill we endeavour to teach young people so that they can solve the algorithmic problems in computers and the world. Constructionism emphasises that building things in the world (including computational artefacts) is an effective way to learn. Both Computational Thinking and Constructionism are highly relevant in the classroom, the former having been the focus of a popular "learn to code" movement within the last 5 years, and the latter having established a worldwide community of practitioners and researchers over the last 30 years. Within the scope of the EU Erasmus funded CONSTRUIT! project, research by partners has demonstrated practical examples of how construals can be used to learn Computational Thinking and as a support for Constructionism.

On reflection, where does the work of CONSTRUIT! sit in relation to Constructionism and Computational Thinking? On the one hand, "making construals" is fundamentally Constructionist in nature (e.g. the incremental building of models based on a learner's experience) and involves many of the same concepts and processes that Computational Thinking advocates teaching children (e.g. the concepts of "decomposition, patterns, abstraction" and the activities of "tinkering, creating, debugging"). On the other hand however, construals are different from the procedural programs that result from the majority of Computational Thinking and Constructionist activities, specifically in the way they are constructed and in their rich possibility for exploration.

This presentation explores practical examples of "making construals" with the aim to position the principles and application of Empirical Modelling in relationship to Computational Thinking and Constructionism.

Construing and building resilience: a case study of the use of Grid Algebra

Sue Johnston-Wilder
University of Warwick
sue.johnston-wilder@warwick.ac.uk

Keywords

ICT-enhanced mathematics education
construal
learning algebra
mathematical resilience
student experience
Grid Algebra

Abstract

In this paper, I seek to weave together the work on construals with my own work on the affective domain, seeking to overcome mathematics anxiety, exclusion and under-attainment; I do so by describing a transformation in learning using a carefully designed artefact, namely Grid Algebra. Grid Algebra 'exploits the computer and associated technologies to offer an interactive experience to its user'.

I give examples of how Grid Algebra uses 'computer-related technology to stage interactive experience of unprecedented richness and subtlety, unprecedented' in that youngsters who had previously found algebra TIRED (tedious, isolated, rote, elitist and depersonalised) began running to lessons. I describe how an 'open-ended context for interaction' in an algebraic construal allowed for human agency in a domain traditionally taught to passive learners.

The paper draws upon data from a PhD (Lugalia, 2015) set in the context of a whole year group learning early secondary algebra in Kenya. I examine the effect of introducing the artefact, combined with dialogic teaching, upon a year group: on pupils' interest in algebra, their involvement and engagement in mathematical learning, their conceptual understanding, their attainment and their mathematical resilience. I lay emphasis on collaboration, articulation, agency, accessibility and variety mediating the use of the artefact. Using Grid Algebra led to many pupils' engagement, enjoyment, new confidence, and eagerness to participate in mathematics. It also led to noticeable increases in attainment and observable attainment-enhancing behaviours in other lessons.

I conclude that construals can indeed have a significant impact on student experience, effort and attainment. Additionally, emphasising affective aspects which reinforce ICT use in mathematics instruction can create an enabling environment for active pupil learning.

Learning to play musical instruments through construal of the Artiphon, a digital musical instrument

Jonathan Foss
University of Warwick
jonathan.foss@warwick.ac.uk

Keywords
musical instrument
construal
Artiphon

Abstract
This presentation will demonstrate a construal that was created to help the maker to learn the notes on the fretboard of a stringed musical instrument. The 'Artiphon' construal[1] allows the maker to visually learn different scales on a musical instrument, and can be used to form a deeper understanding of relationships and patterns that exist between the notes on a fretboard.

Being able to visualise these patterns (e.g. scales, chords, or arpeggios) on an actual instrument can help the musician to both memorise and improvise music. The construal acts as an interactive diagram, showing the musician how to play a variety of different scales and chords. The comments in the construal describe how notes on a fretboard can be thought of numerically, and how mathematical patterns can help to teach music. The 'Artiphon' construal highlights the importance of mathematics in music, and demonstrates simple mathematical relationships that can be used to help beginner musicians to learn to play.

The presentation will also demonstrate an innovative digital musical instrument called the 'Artiphon'[2], which can be configured to behave in different ways according to the preference of the musician. When a musician plays the Artiphon, musical signals are sent to the computer in real-time (via MIDI), which then processes the information and generates the appropriate sound. The keys on the Artiphon are arranged in a 6 by 12 grid in the shape of part of a guitar's fretboard. The mapping of these keys (like the tuning of an instrument) can be redefined instantly, allowing the musician to play the instrument like (for example) a guitar, a keyboard, a violin or a ukulele at the touch of a button.

The 'Artiphon' construal was developed to assist the author with learning how to play the Artiphon, but can easily be adapted to help students to learn more traditional stringed instruments. The construal allows the musician to change a variety of different parameters,including the tuning of the instrument (e.g., guitar tuning or violin tuning) and the displayed scale and a highlighted chord. When such parameters are changed, the construal immediately (through dependency) recalculates note positions and updates the display so that the musician can learn how to play their selected scale on the strings/keys of their selected instrument.

The presentation will also show how JS-Eden can use the MIDI signals sent by the Artiphon to dynamically modify the values of the observables in the construal, instantly updating the visual display. The presentation will also explain how such behaviour can be used to support rehearsal and performances.

[1]http://jseden.dcs.warwick.ac.uk/construit/?load=37&vid=2256
[2]https://artiphon.com

Thinking with Things Made of Metal: A Case Study in Natural Language Construction

John Laudun
University of Louisiana, USA
johnlaudun@gmail.com

Keywords
folk culture
tradition
creativity
networks

Abstract
The recent rise in interest in practice as theory or procedural knowledge in the hard sciences has revealed something the humanities explored beginning in the sixties: that knowledge is constructed brick by brick and often by a multitude of bricklayers, some of whom are not even present in the moment. As Seymour Papert made clear in his push to create a space within which learning was also play, the importance of thinking of thinking as not merely propositions is not only terribly important for the advancement of science but also for the advancement of many social causes. This paper is based on seven years of research to understand the nature of the networks of ideas and people that gave brith to the crawfish boat, an amphibious boat that revolutionized the agricultural economy of south Louisiana. The boat, I argue, represents the ongoing process of traditional knowledge, based entirely in daily and seasonal practices, as it seeks the best fit for human endeavors on a particular landscape. The form of the current boat is not the same as early implementations: each iteration became a thing-to-think-with allowing participants within the network to further their thinking -- such is also the case for the PTO ditcher, which is highlighted as a further example of this particular network (as Latour imagined such things). While some might consider the fabricators simply craftsmen, and contrast them, as Richard Sennett does in his recent work, with designers who are divorced from their work by the computer, this paper argues that given the nature of their practice, and their own increasing adoption of technology in service of their work, makers like these may offer us a glimpse into how the two worlds, of thinking with minds and thinking with hands, can be glimpsed intersecting in an actual, living practice.

Artcasting as a thing-to-think-with: inventive digital education methods

Jen Ross and Hamish Macleod
University of Edinburgh, United Kingdom
jen.ross@ed.ac.uk, h.a.macleod@ed.ac.uk

Keywords
inventive method
playful interaction
cultural heritage
digital education
mobile technology

Abstract
Michael (2012) urges researchers and practitioners to embrace what he calls 'inventive problem-making', and he is not alone in calling for a more open, future-facing, and creative approach to methodology â€“ such calls are being heard across disciplines in the arts, design and social sciences. Speculative or inventive approaches involve envisioning and creating futures, to provoke new ways of thinking and to bring particular ideas or issues into focus. This talk focuses on a recent development – Artcasting – as an inventive method, and connects this to principles of playful interactions found in game design (Whitton 2014).

The AHRC-funded Artcasting project created a mobile application that invited visitors to selected gallery exhibitions to choose an artwork and digitally 'cast' it on a trajectory to a new location, adding information about their choice of cast and their associations with the artwork, and potentially re-encountering their own or other artcasts in the future. With an emphasis on movement, trajectory and imagination, it offered a way of experiencing a gallery exhibition as mobile and open to new interpretations and encounters. Artcasting was designed to simultaneously help visitors make imaginative connections with artworks, and help galleries and museums understand how people are experiencing and engaging with their exhibitions, both in and beyond the gallery. By using the development of an app as not simply the expression of research findings but as an object to think with, and a method in its own right, Artcasting engaged inventively with 'curiosity, critique, doubt, unintended consequences, and emergent properties of technologies in use' (Ross 2016, p.2).

This talk will introduce Artcasting and show how viewing it as a digital 'object to think with' offered new insights into the possibilities and limitations of measuring and understanding engagement. Taking this as a starting point, and drawing on Gee's (2003) principles of probing, discovery and material intelligence, we will discuss other possible visions for Artcasting, to demonstrate how inventiveness in digital methods offers useful modes through which to extend conversations with communities and audiences about the futures we want to enact.

References:

Gee, J. P. (2003). What Video Games Have to Teach Us About Learning and Literacy. Basingstoke: Macmillan.
Michael, M. (2012). 'What Are We Busy Doing?' Engaging the Idiot. Science, Technology & Human Values, 37(5), 528-554.
Ross, J. (2016). Speculative method in digital education research. Learning, Media and Technology, Online First, 1-16.
Whitton, N. (2014). Digital games and learning: Research and theory. Abingdon: Routledge.

What happens when Creative Technologists bring a Researcher into their Team?

Sarah Eagle
Graduate School of Education, University of Bristol
s.eagle@bristol.ac.uk

Keywords
Creative Technologists
Maker Communities
Academic Psychology
Collaboration

Abstract
This presentation offers a reflection on the collaboration between a group of creative technologists, commissioned to provide an installation in a public library, and an academic researcher with a specialism in human learning, and a background in Experimental Psychology. The starting point for the installation was the book, itself a 'thing to think with'. The creative technologists aimed to stimulate thinking about the collective power of books to stimulate human emotions and behavior, to inspire interest in 'digital making', and in the power of creative robotics to provoke fundamental human emotions. Funding from Arts Council England enabled them to employ a researcher, who studied the public response to the installation alongside the aims to produce a 'thing to think with', and then examined these in relation to research from psychology and allied fields. The findings from this study were able to indicate the possibilities for the installation, its context and the components, to be 'things to think with' for people who encountered it. Through the partnership, the creative technologists gained a new insight that went on to support their wider work.

Playfulness as Genre in Constructivist Learning

Hamish MacLeod[1] and Piet Kommers[2]
[1]University of Edinburgh, UK, [2]Helix5, The Netherlands
H.A.Macleod@ed.ac.uk, p.a.m.kommers@utwente.nl

Keywords

playing
learning
'flipped' approaches
making construals
guided discovery
exploratory learning
active learning

Abstract

Playing (rather than gaming) is an uncontrived mind set with a very short iteration cycle between 'action', 'experience' and 'action' again. Its agenda is to find out what a particular world offers in terms of 'new' phenomena, how to build a certain relationship with its main entities, find out dependencies and meet its innate characters. In the context of Construit! we suggest the "making of construals" perspective as very much a special case of (guided) discovery, exploratory, or active learning. Any of these frameworks might guide design of activities that could support a "flipped" approach. Related to this, we see the thinking about "playful learning" as helpful here. In any "flipped" regime, one is engaging much more one's learners to think and behave as researchers – they must work in advance, and bring "hypotheses" which they can test by questioning peers and teachers. "Making construals" might be offered as a discipline to help them to prepare for such activity.

Designing toys for teaching about systems

Hamish Todd
University of Edinburgh
hamish.todd1@gmail.com

Keywords

Education
Design
Programming
Toys
Games
Intuition

Abstract

Many toys exist that embody phenomena found in science. Almost all children learn about symmetry from making paper snowflakes and about surface area minimization from bubbles. A toy can be surprising, tactile, and challenging, increasing our intuition for, and enjoyment of, the thing they embody.

Old toys were limited to physical materials, but today toys can be designed for computers to embody any phenomenon that the designer wishes. But designing them to make the most of human intuitions is a non-trivial task. In this presentation we relate our experiences designing illustrative toys, and in particular why we have decided to stay away from what we call "action at a distance" in their design.

References

Gardner, Martin. "Mathematical games: The fantastic combinations of John Conway's new solitaire game 'life'." Scientific American 223.4 (1970): 120-123.

Transmedia Digital Storytelling – Let's build our story!

Vanda Sousa
Escola Superior de Comunicação Social-Instituto Politécnico de Lisboa – ESCS IPL
vandamariasousa62@hotmail.com

Keywords

Digital Humanities
Trans media
Storytelling
Digital storytelling
active ageing
inter generational dialogue

Abstract

Combining the concept of communicational act as performative of identity, as Benveniste argued, the concepts of classical and digital transmedia narratives as constructors of natural narratological ecosystems with strong social merchandising potential, as Jenkins argues, we'll clarify digital transmedia narratives as a tool in the field of active ageing in the new communicational paradigm where the reader becomes cooperative and makes construals of the text's meanings, highlighting digital transmedia storytelling as a facilitator of the inter-generational communication process.

Towards Distant Reading of Translations: Chasing the Sound Wave of Repetitions across Languages in Resistance to the Flat Logic of Text Analysis

Gabriele Salciute Civiliene
Department of Digital Humanities, King's College London, UK
gabriele.salciute-civiliene@kcl.ac.uk

Keywords

word repetitions
frequency analysis
translation
cross-linguistic distant reading
intratextual space
intersemiotic issues

Abstract

In this paper I reflect on my recent PhD work on distant reading across languages. At its very centre is my making of the theory that interrogates a reductionist approach to texts as plane figures, as encountered in mainstream translation theories. We are not short of visions of texts as multidimensional entities that extend into other texts; that travel from one semiotic system to another; that transcend the opposition between intellectual artefacts and the world (i.e. Derrida's general text that contains texts, thought and institutions). But vision is often ahead of practice: there is a gap between theories of elastic text and attempts to analyze it as such.

One such chasm persists in mainstream translation theories that describe translators as avoiding repetitions out of universal aesthetic discontent. The counts of words lost and replaced with synonyms in translation are taken as behavioural evidence. And here we run into a deep intersemiotic problem. We count what is in plain view, which is nothing more than graphic codes replacing each other on the surface of a text. It is easy to surrender to this flat logic of construing inherently complex human responses to frequencies as switches between the omittable and retainable, thus computing them as discrete counts. But translations, like other texts, are not produced as simplistically as these linguistic descriptions make them seem; neither do they come by chance. They rather evolve gradually through convergence between chance and choice, just like Dawkins's natural designoids (1996).

Word repetitions have variably lasting effects on us, according to experimental psychology (Salciute-Civiliene 2016). My theory envisages them as reverberating sound waves reflected from the walls of spaces that we create through our reading. We turn texts into performative constructs just like an orchestra turns the arrays of musical symbols into sound sculptures. Their dynamic effects depend on context, but also on the internal physics of texts, including the force of word frequencies and distances that separate them. High frequencies enlarge our emotions; distances among them amplify their subliminal workings.

My things to think with are words, but also spaces in between and in place of them as well as my iconoclastic mistrust in what those words and spaces appear to be on the surface. Instead of chasing reasons in ghostly contexts and fleeting moments, I am constructing the intratextual space inhabited by material words and spaces as well as by evocative objects like our emotions, memories and fantasies. Computing this hybrid and complex entity takes much more imaging and imagining than the flat logic of linguistic frequency analysis allows. This reconstruction is voyeuristically satisfying in a Barthesian sense according to which reading is a neurotic response to the "hallucinated form of the text" (1975, p. 63). Deceptively unimaginative words and cryptic spaces give us a glimpse of translators' textual fetishism. In my consideration of the above, I will focus on how visual thinking and the visuals I produced to externalize this logic were instrumental in creating new ways of looking at the problem that was thought long resolved in Translation Studies.

SATURDAY

The CONSTRUIT 2017 Laboratory

Graphical / textual translation in algebra

Anders Kluge[1] and Ellef Fange Gjelstad[2]
[1]University of Oslo, [2]Kikora as
anders.kluge@iped.uio.no

Keywords

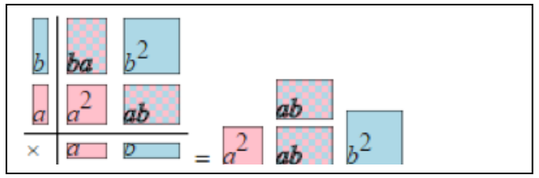
computer science
programming
education
learning theory
constructionism

Abstract

One challenge students are confronted with as they struggle to learn mathematics is the transition from concrete arithmetic to the more abstract notions of general algebra (Kieran, 2007). In learning addition and subtraction the students may get concrete representations they can work with to add two quantities, or to subtract one from the other. Already when multiplication and division are introduced, more abstract thinking is required from the students, and when they move over to general quantities expressed as A's and B's or unknowns expressed in X's and Y's, the abstract thinking becomes and increasingly problematic barrier for many students (Devlin 2013). The idea of a translator between textual and graphical expressions is an effort to soften the transition from the concrete world of physical representations and over to more abstract thinking, and to give the students tools to think with. Using the translation software, the students should be able to write algebraic expressions, and simultaneously see the result as a graphical expression, and vice versa. And more importantly, they should be able to change the graphical expression and immediately see the textual consequences. This will give the students another type of stimulation to learn, engaging in an expressive and dynamic activity to gradually acquire the abstract notions inherent in algebra. In science learning digital representations is researched, used and developed (see e.g. Smetana and Bell 2012). In mathematics the efforts have been turned more towards games (see e.g. Ke 2008) and towards dynamic geometry (Hohenwarter et al. 2009). Compared to dynamic geometry, the textual/graphical translation software is directed more towards expressive power and student experimentation with different visualisations.

A simple example to illustrate the idea:
Textual: (a+b)2 = a2 + 2ab + b2

Graphical:



One of the challenges with this approach is that mapping of abstract entities to a graphical expression gives them to some extent a "value", i.e. a size that makes it possible to assign a value to them. Another is to maintain a both expressive and a logical consistency between the textual and graphic expression.

References

Devlin, K. (2013): The Music of Math Games. American Scientist. Volume 101, Number 2 Page: 87-91.
Hohenwarter, J., Hohenwarter, M. & Lavicza, Z. (2009). Introducing Dynamic Mathematics Software to Secondary School Teachers: the Case of GeoGebra. Journal of Computers in Mathematics and Science Teaching, 28(2), 135-146. Chesapeake, VA:
Ke, F. (2008): A case study of computer gaming for math: Engaged learning from gameplay? Computers & Education 51. pp 1609-1620
Kieran, C. (2007). Learning and teaching algebra at the middle school through college levels. In F. K. Lester (Ed.), Second Handbook of Research on Mathematics Teaching and Learning (pp. 707-762). Charlotte, NC: Information Age.
Smetana, L. K., & Bell, R. L. (2012). Computer Simulations to Support Science Instruction and Learning: A critical review of the literature. International Journal of Science Education, 34(9) pp 1337-1370.

Comparing the process of creating and using little interactive things-to-think-with

Peter Tomcsányi and Michal Winczer
Faculty of Mathematics, Physics and Computer Science, Comenius University, Bratislava, Slovakia
tomcsanyi@slovanet.sk, winczer@fmph.uniba.sk

Keywords

things-to-think-with
programming environments
interactive educational applications

Abstract

An approach to learning to use a new tool is comparing it to some tools one already knows and finding the strengths and weaknesses of the new tool compared to the other ones.

In our paper we will apply this approach to creating and using little interactive educative applications (known as microworlds or things-to-think-with or construals, even if the meaning of these words may not be 100% the same).

We will compare the process of creating and using the same (or very similar) digital artefacts in two or more different environments including Logo and Construit. We will focus on similarities and differences of the process when using the different environments from the conceptual, technical and user experience points of view.

We hope that our work will, among other outcomes, facilitate newcomers to Construit to put the new environment in the context of their previous experience in the field of constructing things-to-think with as well as give the authors and long time users of Construit (and many Construit predecessors) new perspectives and ideas for the further development of Construit.

Digital Sculpting: Modelling, Animation and 3D Printing

Valery Adzhiev
Bournemouth University, UK
vadzhiev@bournemouth.ac.uk

Keywords

Digital Sculpting
Modelling
Animation
3D Printing
Cubism
Heterogeneous Objects

Abstract

Dynamic sculptures created with help of computer-aided design and animation tools and related to a proper artistic style are surprisingly rare. We describe an original approach to creating static and dynamic sculptures in some fine arts traditions, in particular, cubism. Creation of so-called 'impossible' artistic shapes related to M.C. Escher's style will also be outlined. The proposed approach is based on the original theoretical framework for modelling multidimensional heterogeneous objects allowing for their hybrid representation that combines polygonal meshes with functionally-based distance fields. We introduce some original concepts, such as a 4D cubist camera realising multiple projections from 4D space-time to 3D space and combining them using space-time blending to create animated sculptures. We will cover all the main phases through the full production pipeline from data acquisition to shape modelling, animation, rendering and 3D printing. This cohesive process provides artists with a proper balance between automation and control thus extending a traditional suite of rather technical, lacking art specifics, methods and tools. As it has been the case with our long-standing international project 'HyperFun', we specifically aim at educational applications of the proposed framework to provide teachers and schoolchildren with modern tools allowing for creating original artistic content in the context of art and design studies using "visual thinking" and Constructivism-based educational methodology. Some modelled, animated and 3D-printed cubist artefacts are to be presented.

Learning in Digital Reality Environments – Benefits of Virtual Reality and Mixed Reality

Devon Allcoat, Freeha Azmat, Kim Stansfield
University of Warwick, UK
d.b.allcoat@warwick.ac.uk, f.azmat.1@warwick.ac.uk, K.Stansfield@warwick.ac.uk

Keywords

virtual reality
mixed reality
education
things-to-think-with

Abstract

There are a number of applications that exist for virtual reality (VR), but most of them are for the entertainment industry. There are very few educational based applications, and the few that exist have not been widely tested. Mixed reality (MR) is even newer, with the HoloLens not commercially available yet; obtainable only for developers and businesses. Therefore, educational applications have not yet been explored to any significant extent. The aim of our research is to consider possible pedagogic benefits of learning in these digital realities. The focus is not to look solely at what we can do with virtual and mixed reality, but why we should do it, and what specific pedagogic benefits these applications may have. This research considers different forms of thinking in education, as defined by Bloom's Taxonomy; in particular comparing lower forms of learning such as "remembering" with higher orders of learning such as "creating". VR and MR both have different possible uses due to the different architecture of the technologies; therefore, we are considering the benefits and negatives of each, and how they are each best applied. This would allow us to develop educational resources which allow students to learn with digital things-to-think-with. This could then be integrated with current existing module content.

How to Engage Students by Making Robot Movies

Dave Catlin
Valiant Technology Ltd, United Kingdom
dave@valiant-technology.com

Keywords

Robotic Performing Arts
RPA
Roamer
STEM
STEAM
Robot Movies

Abstract

In 2010 Catlin wrote a paper entitled Robotic Performing Arts. In this, he explored the cultural and historical roots of robots. He claimed that our fascination with these artificial life forms inspired creation myths and stories of machines and statues coming to life. The paper also suggested these same subconscious motivations express themselves in the art forms of mimicry and puppet theatre. Robotic Performing Arts claims: robots belong to the artistic world as much as they do to science and technology. Catlin proposed that Robotic Performing Arts offered a way of using educational robots to combine the arts with technical subjects. That is, it puts art into STEM and creates STEAM.

The paper hypothesises how Robotic Performing Arts can offer educational value to students with artistic dispositions with those who prefer technical subjects. Catlin argued making robot movies created an exciting learning environment. This paper reports on several Robotic Arts Projects run in London and America. It broadly supports the original theory, with some changes and suggested improved approaches.

Experimenting on Regular Polygons in Primary School through programming

Angeliki Theodosi
Ralleia Experimental Primary Schools
atheo@sch.gr

Keywords

Regular Polygons
Programming
Construals
Scratch
Primary School

Abstract

The presented project began as an educational lesson plan in geometry for 5th (Kakadiaris Ch., Mpelitsidou N. Stefanides J., Chronopoulou G.) & 6th (Kassoti O., Kliapis P., Ikonomou Th.) grade. Through this activity, pupils had the opportunity to program a simplified construction of a regular polygon using Scratch (Scratch, 2017). The particular algorithm (Theodosi A., 2017) in order to draw a regular polygon calculates the degrees of the angle of the polygon, based on the number of angles of the polygon given each time by the user and draws the regular polygon. Based on their observations about regular polygons for a various number of angles, the pupils concluded that as the number of angles increases the polygon converges to a circle. Through this activity pupils also observed that when in their programs they increased the number of angles and/or the length of the regular polygon's side, the drawing shape was no longer a regular polygon but rather resembled a spiral – probably due to the manipulation of floating point numbers in Scratch.

After the Scratch activity, pupils experimented with a construal on regular polygons (Beynon, 2017). The particular construal was programmed based on the pupils' mathematical background. The pupils had the opportunity to interact with a more stable environment for the construction of regular polygons and to experiment with coding and objects. They also interacted with the construal and they filled an evaluation/self-evaluation form which is presented in this paper.

Implementation of a trigonometric circle construal combined with physical computing

Emmanouil Zoulias, Rene Alimisi and Dimitris Alimisis
Edumotiva, Greece
ezoulias@teemail.gr, ralimisi@gmail.com, alimisis@otenet.gr

Keywords

construal
constructivism
educational technology
physical computing
trigonometry

Abstract

This work will provide a description of a lesson plan for implementation of a “trigonometric circle” using a construal created in the Construit environment and combined with a physical interface (Makey Makey). The trigonometric circle is a very basic lesson for understanding trigonometry at high school level in Greece. Through gaining knowledge of sine, cosine and tangent, a wide area of knowledge opens on various daily life problems. Within this work we prepare a lesson plan to teach this very important knowledge. The lesson plan is based on a construal. Making a construal is a way of using the computer to help us in making sense of a situation. It is introduced as a new digital method. The prepared worksheet can be used by any student to produce an artefact of the trigonometric circle step-by-step, working at their own pace. Construit is the environment used to produce construals. The learners build software artefacts that rely on construction by “making construals” using observables, dependencies and agents. Making a construal is a way of creating interactive open educational resources (OERs). Within this work we combine the ideas of constructionism with physical computing adding the Makey Makey technology as the user interface to move the slider within the construal.

References

Construal Reference: http://jseden.dcs.warwick.ac.uk/construit/?load=262
Triangles: http://jseden.dcs.warwick.ac.uk/construit/?load=51
is: http://jseden.dcs.warwick.ac.uk/construit/?load=69
Trigonometric circle definitions:
https://online.math.uh.edu/MiddleSchool/Modules/Module_4_Geometry_Spatial/Content/UnitCircleTrigonometry-TEXT.pdf
https://www.mathsisfun.com/geometry/unit-circle.html
https://en.wikipedia.org/wiki/Unit_circle

Representing Digit by Digit Algorithms

Anders Kluge[1] and Ellef Fange Gjelstad[2]
[1]University of Oslo, [2]Kikora as
anders.kluge@iped.uio.no

Keywords

representation
digit by digit
primary

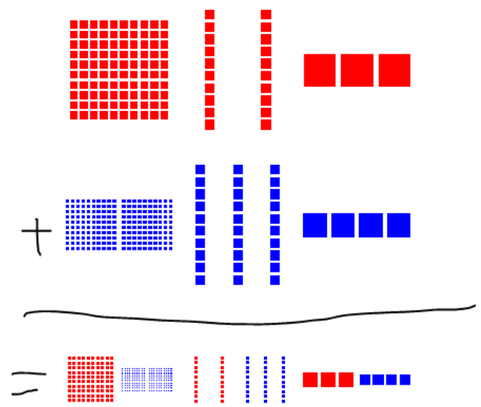
Abstract

In primary school, pupils are introduced to integers with more than one digit. They learn how to make addition, subtraction, multiplication etc. on such numbers. The first algorithm may be column addition with memory digits.
At this point, the algorithms may seem impossible to understand, and be perceived as magic and obscure operations. This creates an impression that math is about following rules. Also, it may create an impression about math as difficult stuff far from daily life.
Our (imagined) software contains three solutions to mitigate this:

- When the student has performed an algorithm, the software marks the correct and wrong digits. In the column addition, an error might be present in the sum or in the memory digits.
- When the student has performed an algorithm, the software translates the result into another representation. One possible representation is shown in the example.
- The student can perform the algorithms with pen on paper. The student can then take a picture of the paper, e.g. with a pad or phone, and an app will recognize the digits and analyse the result.

The calculation 123 + 234 = 357 could be set out like the example below, and represented graphically as shown to the right.

$$\begin{array}{rrrr} & 1 & 2 & 3 \\ & 2 & 3 & 4 \\ \hline = & 3 & 5 & 7 \\ \hline\hline \end{array}$$



One interesting challenge is to define a graphical language for the math. We would like the language to be extendable into algebra. Another is to handle errors and memory digits. A technical challenge is the text recognition.

A simple use case is students working like they are used to, on pen and paper. The students get an exercise, possibly from a text book, calculate on paper, and use their camera device to analyse the exercise. The feedback will help the student discover errors and learn from them.

Riga Schools Making, Thinking and Learning in the Digital Age

Lasma Lancmane and Lolita Meza
Education, Culture and Sports Department of Riga City Council, Latvia
lasma.lancmane@riga.lv, lolita.meza@riga.lv

Keywords
e-learning environment
Make: Learn: Share: Europe (MLSE)
ambassadors training on robotics coding mobile phone applications game development
pedagogical and psychological aspects
digital skills

Abstract
The Education, Culture and Sports Department of the Riga Council (the Department) is a supervising body of the local government, which implements the preschool and general education policy and promotes further education of teachers. The Department has jurisdiction over more than 300 institutions established by the Riga City Council: including 150 pre-school education institutions (kindergartens), 117 comprehensive elementary education and comprehensive secondary education institutions (schools).

Local government input in the technological infrastructure of municipality schools has been intense in 2011 – 2013 due to the EU funded projects when computers, interactive whiteboards, tablets were supplied. During the project time, an e-learning environment for Riga schools was established: http://macibas.e-skola.lv with the aim to establish one platform for teachers and pupils.

The currently running international project Make: Learn: Share: Europe (MLSE) is designed to develop digital skills amongst young people across Europe, in order to assist with learning and enterprise development. A consortium of partners from different EU member (Italy, Spain, Latvia and the UK) states is creating a large scale student ambassador association who run training and learning opportunities for others in schools or other local settings.

The subject areas of the training include robotics, coding, mobile phone applications, game development and others. The learning materials for these sessions, called 'Open School Units', will be openly available.

Knowledge, skills and experience in the digital era are all changing in a rapid manner – whilst this can frustrate long term planning it does give an exciting edge to new skills development. The rate of technological change means that there is a need to consider the use and application of learning on an increasingly regular basis. It is important to assess not only the advantages of digital making but also pedagogical and psychological aspects of it.

SUNDAY

Looking to the future

Future directions: making construals to evaluate and inform government policy

Karl King
Winning Moves Ltd, UK
karlking2k@gmail.com

Keywords

Making construals
Policy evaluation
Theory development
Realist evaluation
Mechanisms
CMOCs

Abstract

Theory-based evaluation approaches are becoming increasingly popular in the evaluation of government interventions (i.e. policies, programmes, initiatives, projects). This is for two main reasons:

Where it is infeasible to implement randomised control trials or quasi-experimental designs, theory-based approaches offer an alternative framework for causal inference (generative causation), providing at least some opportunity to explore whether/how an intervention influenced observed outcomes.

Where experimental designs are feasible, these allow conclusions to be drawn about whether or not an intervention worked, but, unlike theory-based approaches, provide limited insight into the specific causal mechanisms at play and how the intervention worked (or not) in particular contexts/circumstances.

Realist evaluation is particularly popular at the moment, because it addresses directly the question of what works (or not), for whom, in what respects, to what extent, in what contexts and how/why.

Realist evaluation starts and ends with a theory; the goal being to gather evidence that enables the initial theory to be tested and refined. Each theory is developed in the form of 'context, mechanism, outcome configurations' (CMOCs), with each new piece of evidence observed feeding in to developing, supporting, refuting or refining the existing theory.

Despite the complexity inherent in many of the systems government seeks to influence, theories are often developed as static depictions (think spaghetti of boxes/arrows) – and can very quickly become difficult to work with and challenging for those not involved in the theory development process to understand. There is also a distinct risk of oversimplification stemming from the difficulty of depicting complex systems.

In the view of the author, realist evaluations could commence with the evaluator making an interactive construal instead of producing a static depiction of tentative theory. This could then be linked more directly to the evidence used to develop and test the theory, and allow the computer to support the process of identifying instances where the theory (or construal) does not appear to reflect the evidence.

The author expects much of CONSTRUIT 2017 will be devoted to making construals in educational contexts, in particular to support learning in situations where the goal is for a student is to develop and improve their understanding of something already understood by their teacher. The author's personal interest is in making construals to support understanding in contexts/domains which are not well understood already, as is the case in many evaluations of government interventions. That said, the broader vision is that such construals made within evaluations could be used, subsequently, to support learning in educational contexts.

References:

[1] Pawson, Ray, and Nick Tilley. Realistic evaluation. London: Sage, 2014. Print. First published 1997
[2] Pawson, Ray. Evidence-based policy: a realist perspective. Los Angeles: Sage, 2006. Print.

A Brexit EM Construal?

Soha Maad
IGROW EEIG, Ireland
sm@sohamaad.net

Keywords

BREXIT
EM
construal
social
economic
impact
modelling
strategy
policy
insight

Abstract

British Exit of the European Union (BREXIT) is a landmark event in the history of Britain. Understanding and analysing the current and future impact of this event are of paramount importance to all economic stakeholders including government, businesses, and citizens at large in UK and globally.

This paper describes a vision for an Empirical Modelling (EM) construal of BREXIT. Such a construal would be a tool to explore social, economic, and political factors surrounding BREXIT and its impact on United Kingdom and world trade, growth, and economy. It would serve as a real world computer based modelling tool to explore a historical event that shapes the UK and the world economy. It would be a growing model of real world observables and complex dependencies between these observables. Users would interact with the construal to gain knowledge, insight, and infer strategies, policies, and future actions.

Could such a construal be developed? The author believes that Empirical Modelling can be used to analyse complex real world situations to gain world insight and infer future plans and actions. In principle, the Empirical Modelling framework, foundation and concepts is sufficiently broad. A BREXIT EM construal would be developed using an online environment and technically inspired by a range of EM construals previously prototyped using experimental modelling environments to demonstrate the full potential scope and features of Empirical Modelling.

Existing tools and technologies to implement these concepts admit various technical limitations and may fail to convey the model the experience of the real world to which EM aspires. The author nonetheless believes that a BREXIT EM construal would be uniquely well-suited to delivering a rich computer based experience of such a complex real world situation and event and analysing its local and international impact.

Tree-based Neural Network for Educational Data Mining and Cluster Analysis: Construit Approach

Tapani Toivonen, Ilkka Jormanainen, Markku Tukiainen
University of Eastern Finland, Joensuu, Finland
tapani.e.toivonen@gmail.com, ilkka.jormanainen@uef.fi, markku.tukiainen@uef.fi

Keywords

Construit
clustering
educational data mining
data mining
visualization

Abstract

The Construit environment drives our development efforts for a novel clustering method for educational data mining settings. Cluster analysis is a set of approaches that aim to label similar data instances together. Cluster analysis is widely used in educational data mining but the heterogeneous data gathered from the educational settings imposes requirements and constraints on the traditional cluster analysis algorithms that many fail to fulfill. We have developed a new cluster analysis algorithm named Neural N-Tree (N-dimensional) for educational data mining. Neural N-Tree preserves the relationships of data instances in the inter-cluster levels and helps in the visualization of high dimensional data. Our approach to cluster analysis aims to support educators to deepen their domain specific knowledge about the authentic educational settings through visualization and non-discrete clusters. Currently Neural N-Tree has been tested with various data sets with promising results. Comparison with other cluster analysis algorithms shows that Neural N-Tree provides a powerful way to reduce the dimensionality of data through visualization and knowledge about the relationships of different clusters.

Web-based Knowledge Sharing Adoption among Academics in Saudi Arabian Higher Education Institutions

Nouf Almujally and Mike Joy
Computer Science, University of Warwick
n.almujally@warwick.ac.uk, M.S.Joy@warwick.ac.uk

Keywords

Knowledge Management
Knowledge Sharing
elearning
Technology Acceptance
Higher Education Institutions
Academics

Abstract

Knowledge sharing has become a significant source of success in Knowledge Management (KM). In many organizations, knowledge management is often inadequate when it comes to web-based knowledge sharing, particularly among academics who work in Saudi universities. To ensure that a knowledge sharing system can be well implemented and used when communicating internally in an academic context, there is a need to know why academics accept or reject the use of web-based knowledge sharing systems. Therefore, the aim of this research is to determine the factors that affect academics' behaviours toward using web technology to share knowledge in Saudi Arabian higher education institutions. Furthermore, a web-based knowledge sharing adoption model is constructed based on the factors which already exist in technology acceptance theories, such as the Unified Theory of Acceptance and Use of Technology (UTAUT) and Task-Technology Fit (TTF), as well as other factors which are explored in knowledge sharing literature reviews to enrich the proposed model. Then, the model will be edited and refined using mixed method approach. The future work will expand the model and evaluate it to ensure that it fits the academics' needs.

CONSTRUIT 2017
Department of Computer Science
University of Warwick, CV4 7AL
t: +44 (0)24 7652 3193
e: construit@warwick.ac.uk
w: edumotiva.eu/construit2017