Using Interface Design to Develop Computational Thinking Skills

Ana C. Calderon
Department of Computing
Cardiff Metropolitan University, UK
acalderon@cardiffmet.ac.uk

Tom Crick
Department of Computing
Cardiff Metropolitan University, UK
tcrick@cardiffmet.ac.uk

ABSTRACT

Human-computer interaction is a long established sub-discipline of computer science. While there has been significant focus on the importance of developing computational thinking skills, there appears to be a gap in the literature in using HCI principles, analysis and design as a framework for doing so. We present the first step to identify methodologies for systematically introducing HCI to pupils from an early age, presenting a commentary for their prospective future application, comparing to similar approach as other foundational aspects of computer science in developing computational thinking skills that have been considered for the past decade.

Categories and Subject Descriptors

K.3.2 [**Computers & Education**]: Computer and Information Science Education—*Computer Science Education*

Keywords

Computer Science Education; Computational Thinking; HCI; Design; Interaction

1. INTRODUCTION

Computational thinking is increasingly valued for its significance in the teaching of foundations of computer science as well as its ability to aid broader problem-solving skills across a range of subjects [2, 3]. HCI is a well established research and application area in computer science, but it is also a multi-disciplinary domain which, we argue, has a number of important links to computational thinking. Interactivity and design has been a somewhat undervalued theme during the growth of computational thinking as a high-value skill, with an associated gap in the literature. Some proof-of-concept solution methodologies are detailed, targeted at particular aspects of HCI, presented in the form of sessions accessible to children of primary school age.

WIPSCE 2015 London, UK

2. SUGGESTED SESSIONS

We now give methodologies, in the form of activity-based learning sessions, to teach elements crucial to HCI to secondary school students. These are ideas are influence by Norman [8], Rogers [10] and Shneiderman [11].

2.1 Session A

The first session we describe is intended to cover negative transfer as well as the principle of affordance, and we break down the explanation not by the flow of the session, but by the intended learning outcome, this is intended to facilitate also the explanation of the effects making the presentation accessible outside the HCI community. The session was devised with the use of a "doll house" specifically designed in a self-contradictory manner. The students are given clues to navigate around the house and their goal is to find a pebble. To find the hidden pebble, the students must follow instructions and activities specified in the cards. This follows similar patterns as a popular game show, and the reason is to help the students view this as a fun activity rather than a typical learning session (via traditional classroom environments).

Principle of affordance

Affordances are the product of agents and their environment [5], gien any agent-environment combination a affordance may or may not exist. If one does the agent needs to be aware of it, however for most HCI researchers when affordance is mentioned it is typically assumed that the agent is aware of it, calls this a perceived affordance [7].

The very first instruction given to students is to simply enter the house, the entrance consists of a front door, porch, both doors have a door knob, the first door must be pushed whereas the second door needs to be pulled. Students are likely to try to turn the knob the first time and push the second. At this stage in the session, an explanation is due, on how past experiences can affect learning of new tasks. In addition most students are likely to attempt to turn the knob, which makes it an excellent point to teach students about the principle of affordance.

Explaining transfer effects applied to design methods in HCI

Negative transfer[6, 9, 13] is a term in behavioural psychology to describe how new task learning can be negatively affected by knowledge of similar or related tasks. This has consequences to design of interactive designs [12] For example,[1] uses a simulated experiment (based on real industrial

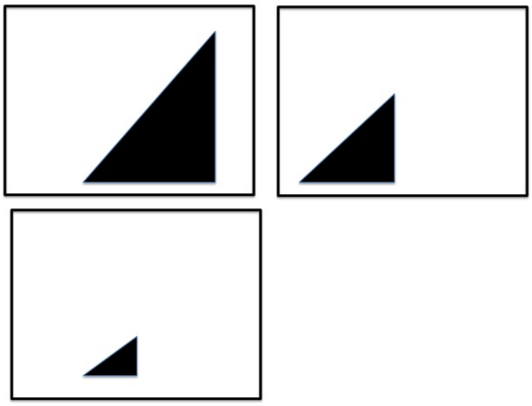

Figure 1: Example of representing portions via text.

work environments) to support a hypothesis of how interface changes are less likely to cause accidents when it limits changes to former interaction patterns.

Amongst the instructions, pupils are asked to fill up a watering can in the kitchen, water the plants, return to the house and clean the same watering can in the downstairs guest bathroom. The first tap used must be turned clockwise in order to let out water, conversely the second tap must be turned anti-clockwise for the same goal to be achieved. This is the second intentional opportunity during the activity, in which teachers can explain transfer effects and start a discussion bout its importance to design.

2.2 Session B

The second session is based on a similar activity as published in [4] where students are given cooking ingredients lists to transform into symbols.

Design for recognition

As the above section demonstrate, learning how to achieve certain tasks will have consequences to learning new tasks. As humans we are constantly learning new skills and it is thus essential to design intuitive interactive devises. The following two sessions are aimed at highlighting the importance of designing to minimise the amount of learning required.

For this, pupils are placed into two groups, the first group is given three lists: ingredients, portions and instructions. They must find a way to abstract away from the textual information and draw icons and symbols that will aid the second group of pupils put together a recipe. The list is small, “chocolate, milk, butter” and the portions are simple “a little”, “a lot”, “a big piece”, “a medium piece”, the instructions are also simple “start mixing”, “stop mixing”, “start pouring”, etc. The students should enjoy themselves as they try to discover whether how close to the original recipe the second group’s recipe is. After the session is complete the teacher should emphasise their learning of difficult but important it is to design interfaces with icons that are meaningful. Examples of solutions to text abstraction can be found in 1 (representing small, medium and large piece).

Iterative Design

Iterative design: After determining the users, tasks, and empirical measurements to include, perform the following iterative design steps:

- Design the user interface;
- Test;
- Analyse results;
- Repeat;
- Repeat the iterative design process until a sensible, user-friendly interface is created.

In addition, this session can be structured to teach other design-relevant methodologies to children, namely iterative design. This is achieved by allowing the pupils second and third attempts at their representation of the text given to them (we expect more than three attempts would be frustrating for the pupil, and three are enough to illustrate the principle of iteration in design). The pupils must be told that the iteration is intended to improve the design and should be repeated until a good and sensible set of symbols is achieved (as would happen with an interface with programmers). This should be followed by a short formative session on iterative design, highlighting the main concepts in a language accessible to the particular age group, for instance:

1. What is the target audience of your interactive software? What is the goal of your software? What empirical testing can be done to check for accuracy in achieving the tasks post creation of your interface?
2. Design your interface
3. Test your interface
4. Interpret results
5. Iterate (repeat)
6. End when satisfied with resulting interface

The sessions described are intended as examples of principles relevant to HCI, this is the beginning of a list of potentially several principles that are important to HCI, we mean for this sessions to be seen as illustrative examples rather than containing a comprehensive list of principles.

3. CONCLUSIONS

We have presented a methodology for introducing rigorous design principles and theories relevant to human-computer interaction to young children. This is exemplified with two sessions developed encompassing computational thinking. We hypothesise that HCI can be embedded as a valuable part of developing computational thinking skills in young students, and that these will have a positive impact more broadly across their future learning, not just for computer science. Furthermore, we reiterate the wider societal benefits of developing these broad computational skills, both as baseline digital competencies to ensure a digitally engaged citizenry, as well as high value skills for the economies of the future. Immediate future work will consist of an implementation of the sessions with a cohort of students to investigate their feasibility as part of a scheme of work and how they progress over an academic period.

4. REFERENCES

[1] D. Besnard and L. Cacitti. Interface changes causing accidents. an empirical study of negative transfer. *International Journal of Human-Computer Studies*, 62(1):105 – 125, 2005.

[2] N. Brown, M. Kölling, T. Crick, S. Peyton Jones, S. Humphreys, and S. Sentance. Bringing Computer Science Back Into Schools: Lessons from the UK. In *Proceedings of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE 2013)*, pages 269–274. ACM Press, 2013.

[3] N. Brown, S. Sentance, T. Crick, and S. Humphreys. Restart: The Resurgence of Computer Science in UK Schools. *ACM Transactions on Computer Science Education*, 14(2):1–22, 2014.

[4] B. T. Fellows M. and W. I. Computer science unplugged. `http://csunplugged.org/`, September 2005.

[5] J. J. Gibson. The theory of affordances. *Hilldale, USA*, 1977.

[6] Lunchin and Lunchin. *How People Learn*. 1970.

[7] D. A. Norman. Affordance, conventions, and design. *interactions*, 6(3):38–43, 1999.

[8] D. A. Norman. *The design of everyday things*. Basic books, 2002.

[9] S. J. Pan and Q. Yang. A survey on transfer learning. *Knowledge and Data Engineering, IEEE Transactions on*, 22(10):1345–1359, 2010.

[10] Y. Rogers, H. Sharp, and J. Preece. *Interaction design: beyond human-computer interaction*. John Wiley & Sons, 2011.

[11] B. Shneiderman. *Designing the user interface-strategies for effective human-computer interaction*. Pearson Education India, 1986.

[12] Y. Wærn. Varieties of learning to use computer tools. *Computers in human behavior*, 9(2):323–339, 1993.

[13] D. J. Woltz, M. K. Gardner, and B. G. Bell. Negative transfer errors in sequential cognitive skills: Strong-but-wrong sequence application. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 26(3):601, 2000.