ICCE 2018 Tutorial

Role of Unplugged Activities in CT Education

Chee-Kit Looi

National Institute of Education, Nanyang Technological University Singapore

Outline

- Intro to some unplugged activities
- Discussion on research agenda for unplugged activities

Let’s quickly run through some “unplugged” activities for CT (Computational Thinking)!

Examples of activities

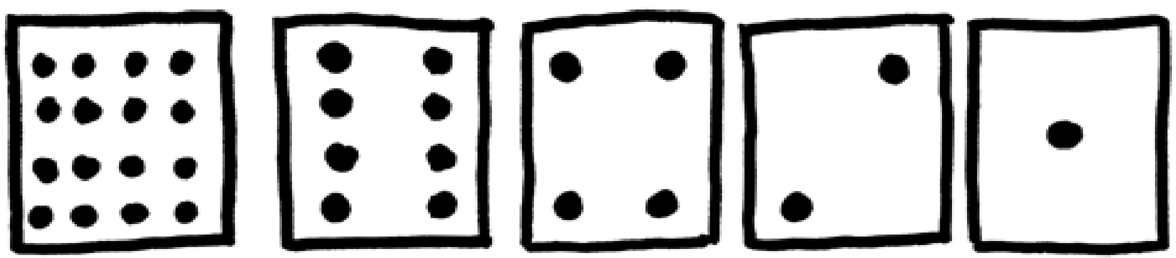
1. Binary Numbers
2. Sorting Algorithm & Sorting networks
3. I can READ your mind
4. Parity Bit Checking
5. Deadlock avoidance

Acknowledgements: 1, 2, 4 and 5 from CS Unplugged https://csunplugged.org/en/
3 from Prof Hon Wai Leong of NUS, SIngapore

Activity 1: Binary Numbers

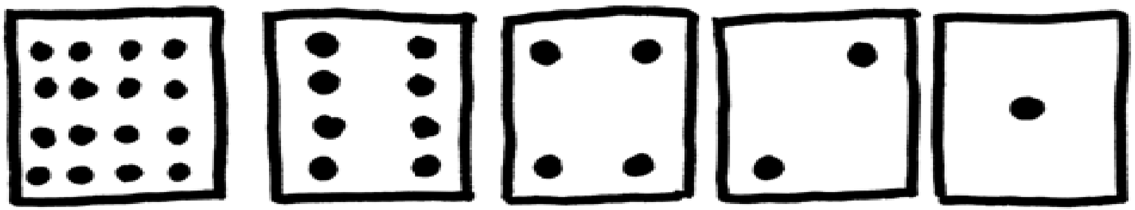
- What is base ten ?
- What is base two (binary number)?
- Why do computers use binary?

Activity 1: Binary Numbers



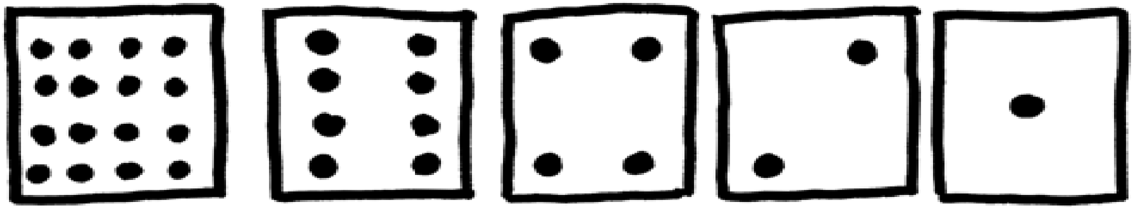
- How would you represent more numbers with the given cards?

Activity 1: Binary Numbers



- Find out how to get 3, 12, and 19.
- Is there more than one way to get any number?
- What is the biggest number you can make? What is the smallest?
- Is there any number you cannot make between the smallest and biggest numbers?

Activity 1: Binary Numbers



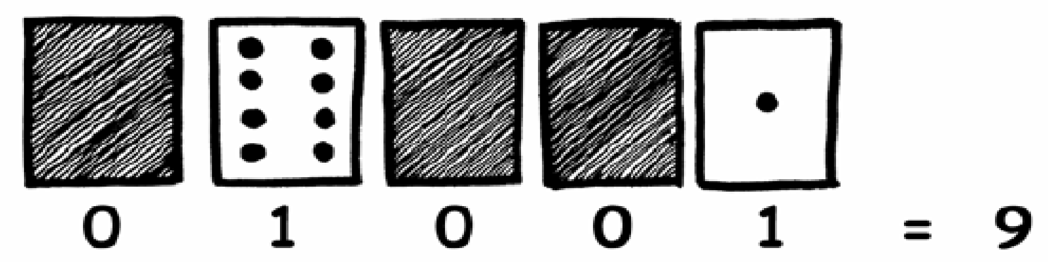
- Ask one participant’s birthday.
- What day of the month were you born? What is your birth date in binary?
- Find out what your birthday in binary.
- Do you notice that the number of the dots on the cards?
- (Each card has twice as many as the card to its right.)

Activity 1: Binary Numbers

- Note down the corresponding binary number-- The binary system uses zero and one to represent whether a card is faced up or not.

  0 – shows that a card is hidden

  1 – means that you can see the dots
- For example:



Activity 1: Binary Numbers

Watch the video

https://www.youtube.com/watch?v=b6vHZ95XDwU

Activity 2: Sorting Algorithms

- Why is sorting important in our lives?
- We need to put lists into some sort of order
  - for example names in alphabetical order, appointments or email by dates.
- Brainstorm all places where putting things into order is important.
  - What would happen if these things were not in order?

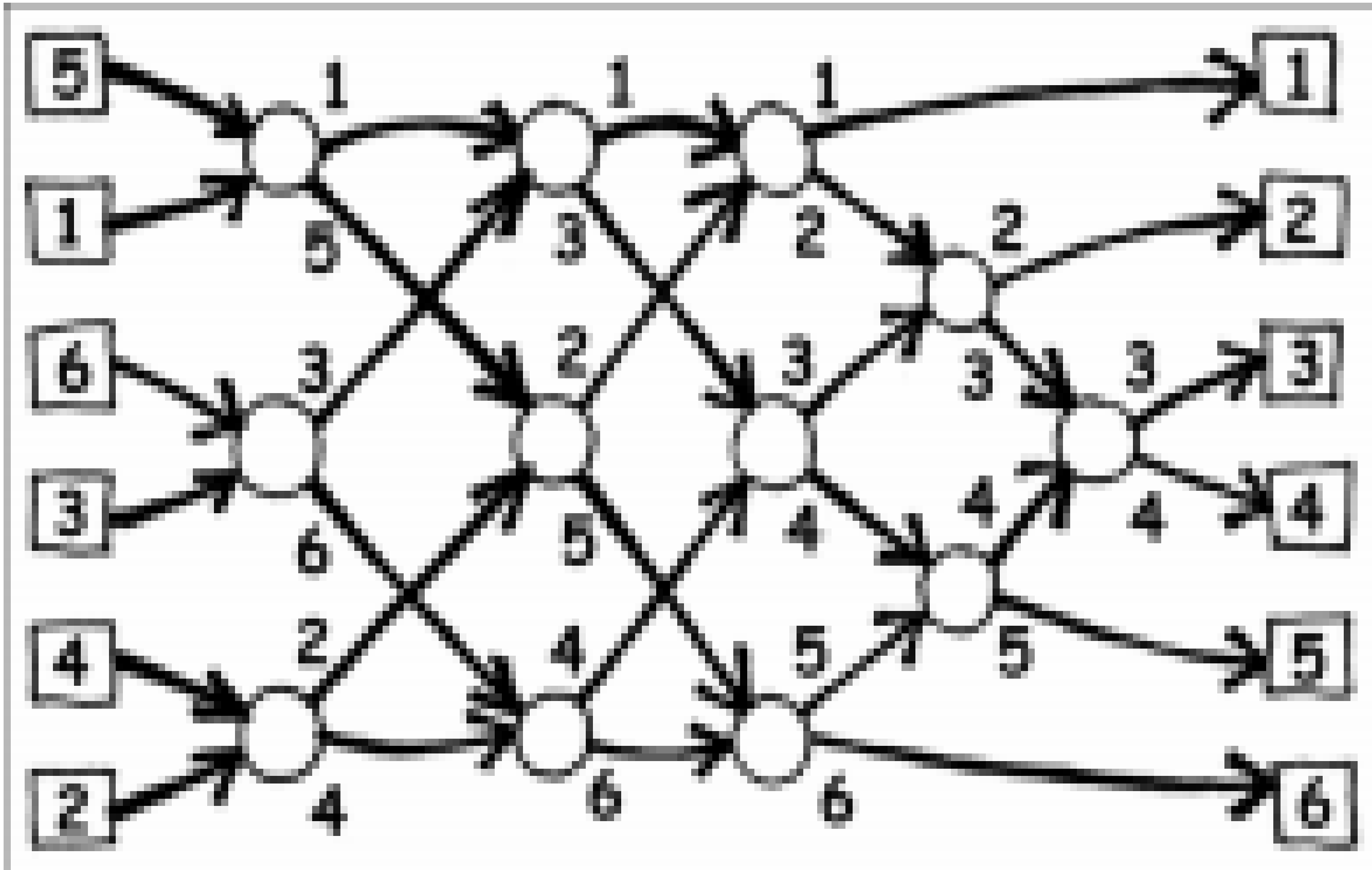
Selection Sort
Quick Sort

- https://www.youtube.com/watch?v=cVMKXKoGu_Y&feature=youtu.be
- How many comparisons you need to make to sort 6 objects into order.
  - What about 7 objects?
  - What about 20?

Activity 2: Sorting Networks

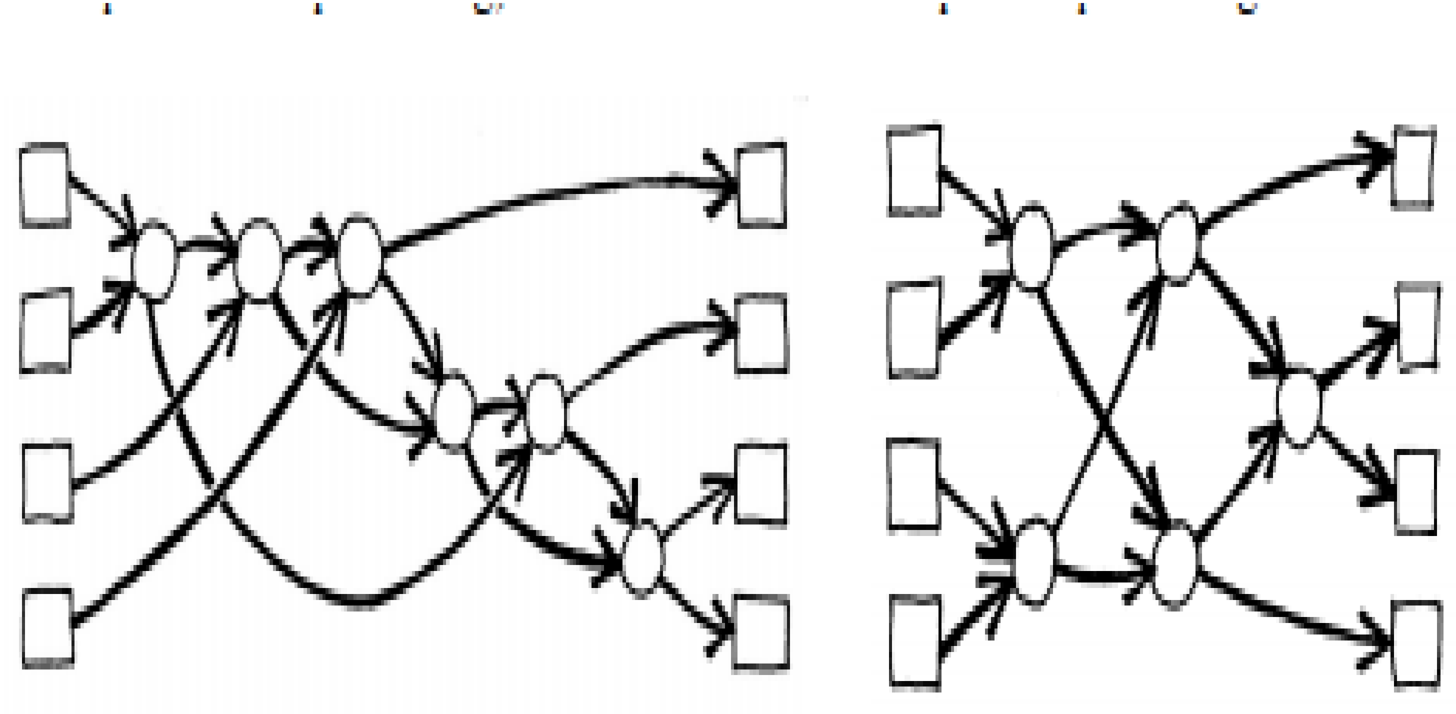
- How to do a more efficient sorting?
- How to speed up computational processing?
- Imagine one person digging a ditch ten meters long. If ten people each dug one meter of the ditch the task would be completed much faster.

Activity 2: Sorting Networks



Activity 2: Sorting Networks

- Discuss which of the following sorting networks will be faster(serial versus parallel)



Activity 2: Sorting Networks

- https://www.youtube.com/watch?v=30WcPnvfiKE

Activity 3: I can READ your mind!

Think of a number from 1 to 31.
And I can read your mind ☺

Activity 3: I can READ your mind!

Is your number here ?

1	3	5	7	9
11	13	15	17	19
21	23	25	27	29
31				

Activity 3: I can READ your mind!

Is your number here ?

2	3	6	7	10
11	14	15	18	19
22	23	26	27	30
31				

Activity 3: I can READ your mind!

Is your number here ?

4	5	6	7	12
13	14	15	20	21
22	23	28	29	30
31				

Activity 3: I can READ your mind!

Is your number here ?

8	9	10	11	12
13	14	15	24	25
26	27	28	29	30
31				

Activity 3: I can READ your mind!

Is your number here ?

16	17	18	19	20
21	22	23	24	25
26	27	28	29	30
31				

Activity 3: I can READ your mind!

Your number is -- ! ! !

Do you know the magic behind it

Actually you already told me your number in binary.

Activity 3: I can READ your mind!

Is your number here ? = is your binary number with first digit 1 ?

1 00001	3 00011	5 00101	7 00111	9 01001
11 01011	13 01101	15 01111	17 10001	19 10011
21 10101	23 10111	25 11001	27 11011	29 11101
31 11111				

Activity 3: I can READ your mind!

Is your number here ? = is your binary number with second digit 1 ?

2 00010	3 00011	6 00110	7 00111	10 01010
11 01011	14 01110	15 01111	18 10010	19 10011
22 10110	23 10111	26 11010	27 11011	30 11110
31 11111				

Activity 3: I can READ your mind!

Is your number here ? = is your binary number with third digit 1 ?

4 00100	5 00101	6 00110	7 00111	12 01100
13 01101	14 01110	15 01111	20 10100	21 10101
22 10110	23 10111	28 11100	29 11101	30 11110
31 11111				

Activity 3: I can READ your mind!

Is your number here ? = is your binary number with fourth digit 1 ?

8 01000	9 01001	10 01010	11 01011	12 01100
13 01101	14 01110	15 01111	24 11000	25 11001
26 11010	27 11011	28 11100	29 11101	30 11110
31 11111				

Activity 3: I can READ your mind!

Is your number here ? = is your binary number with fifth digit 1 ?

16 10000	17 10001	18 10010	19 10011	20 10100
21 10101	22 10110	23 10111	24 11000	25 11001
26 11010	27 11011	28 11100	29 11101	30 11110
31 11111				

Activity 3: I can READ your mind!

Now we convert binary to decimal, i.e. adding up the left up corner numbers where you say yes !

$$(10101)_2 = 1\times2^4 + 0\times2^3 + 1\times2^2 + 0\times2^1 + 1\times2^0$$

$$(10101)_2 = 1\times16 + 0\times8 + 1\times4 + 0\times2 + 1\times1$$

$$= 16 + 4 + 1 = 21$$

How many tables do I need if we do guessing from 1 to 100? Can you create the table?

Activity 4: Parity Bit Checking

- The world is noisy place, and errors can occur whenever information is stored or transmitted.

- How could we determine when errors have occurred?

Activity 4: Parity Bit Checking

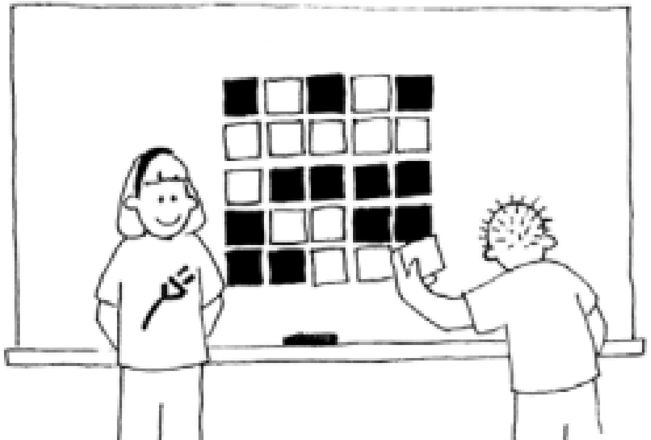
- Error detection techniques add extra parity bits to data to determine when errors have occurred

- This activity is a magic trick which most audiences find intriguing. In the trick the demonstrator is “magically” able to figure which one out of dozens of cards has been turned over, using the same methods that computers use to figure out if an error has occurred in data storage.

Activity 4: Parity Bit Checking

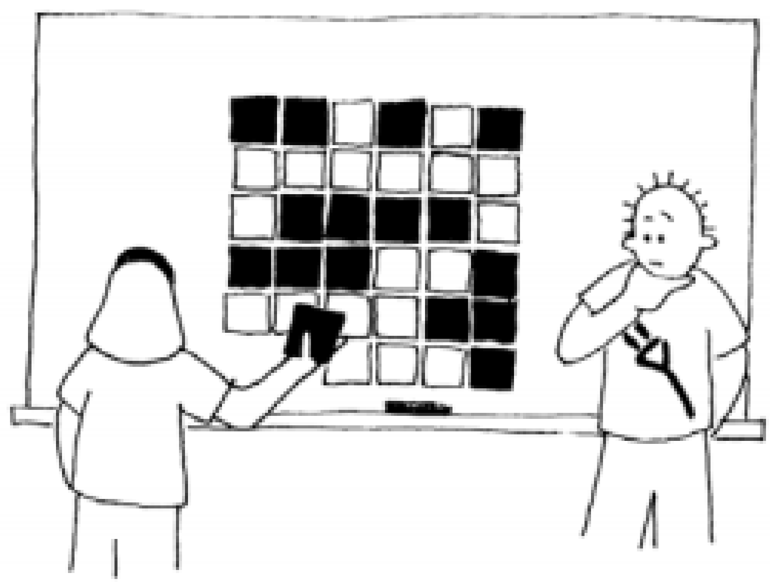
Here the magic begins!

I will need a pile of identical, two-sided cards. Choose a participant to lay out the cards in a 5 × 5 square, with a random mixture of sides showing.



Activity 4: Parity Bit Checking

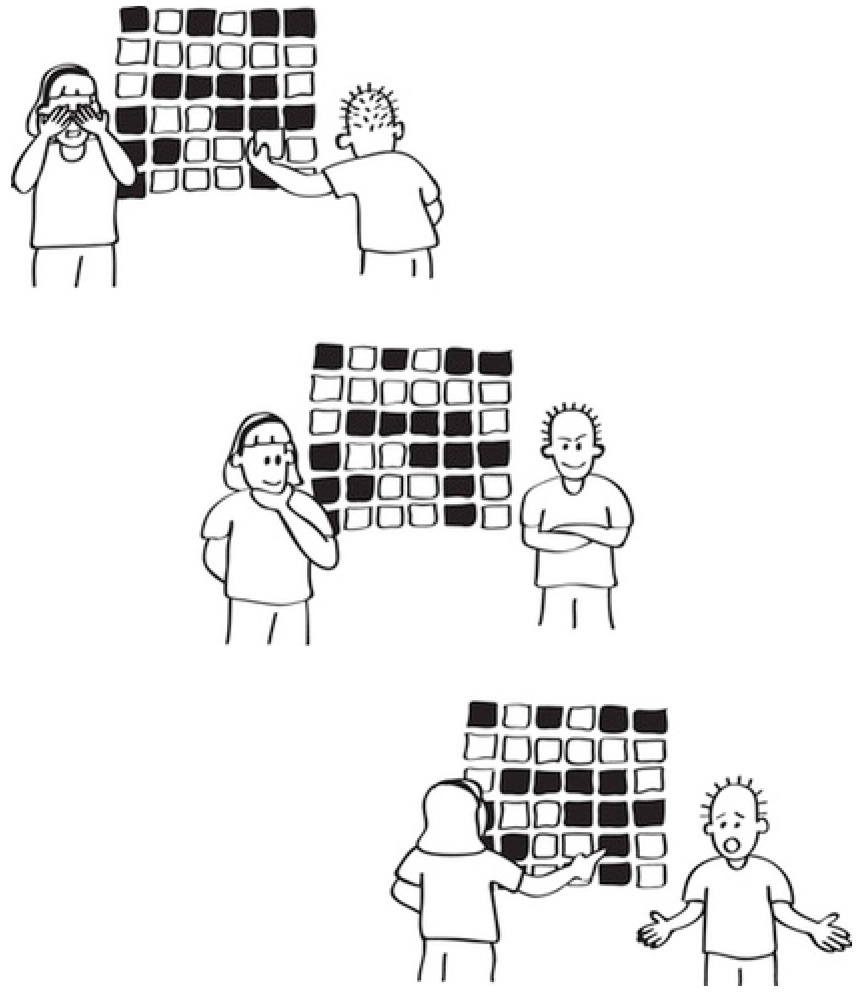
I casually add another row and column, “just to make it a bit harder”, and form 6 × 6 square.



Activity 4: Parity Bit Checking

Get a participant to flip over one card only while I cover my eyes.

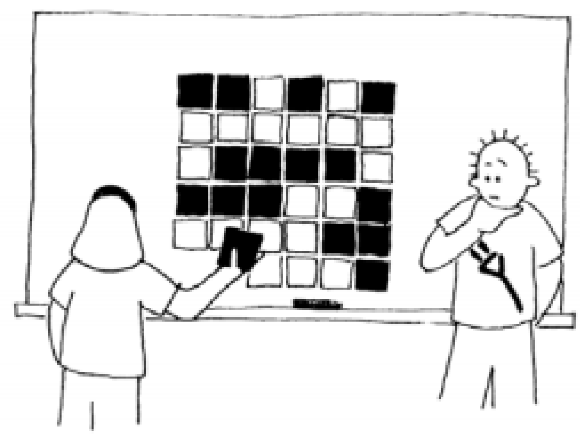
But I will still figure our which card is touched!



Activity 4: Parity Bit Checking

How the trick is done?

The 6th row and 6th column cards are the key to the trick. You must choose the extra cards to ensure that there is an even number of coloured cards in each row and column.



Activity 4: Parity Bit Checking

Computer is using the same technique to identify data error.

For Example：

1 0 1 0 1 0 1 ×
1 0 1 0 1 1 1 ×
1 1 1 0 1 0 0 ×
0 1 0 1 1 1 0 ×
1 1 0 1 0 0 1 ×
0 0 1 1 0 1 0 ×
× × × × × × ×

“×”is the parity bit that you add on.

Activity 4: Parity Bit Checking

Think about it:

1.What happens if two, or more, cards are flipped?

2. consider the lower right-hand card. If you choose it to be the correct one for the column above, then will it be correct for the row to its left?

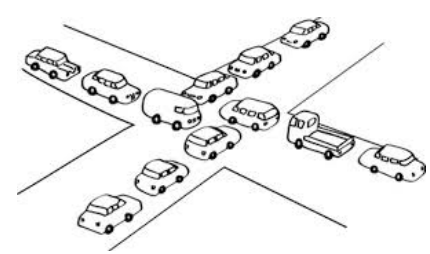
3. In this card exercise we have used even parity—using an even number of coloured cards. Can we do it with odd parity?

Activity 4: Parity Bit Checking

[Computer Science Buskers (Error Detection).mp4]

Activity 5: Deadlock Avoidance

- When you have a lot of people using one resource
  - such as cars using roads, or messages getting through the Internet), there is the possibility of “deadlock”.
- Brainstorm real-world situation of deadlock
  - a traffic jam, or
  - trying to get a lot of people through a doorway at once.



Activity 5: Deadlock Avoidance

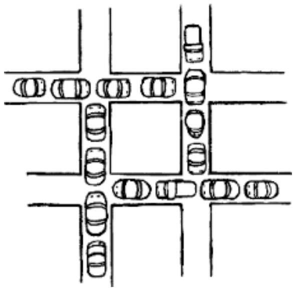
- Groups of 7 participants sit in a circle.
- The participants are labelled with a fruit tag. There are 2 fruit toys corresponding to the tag on them, except for one participant, who only has one corresponding fruit toy to ensure that there is always an empty hand.
- Distribute the fruit toys randomly to the participants in the circle. Each participant has 2 toys, except for one participant who has only one. (No participant should have a toy corresponding to their tag.)

Activity 5: Deadlock Avoidance

- The participants pass the toys around until each participant gets the toys corresponding to their tag. The 2 rules are:
- Only one toy may be held in a hand.
- A toy can only be passed to an empty hand of an immediate neighbor in the circle. (A participant can pass either of their 2 toys to their neighbor.)

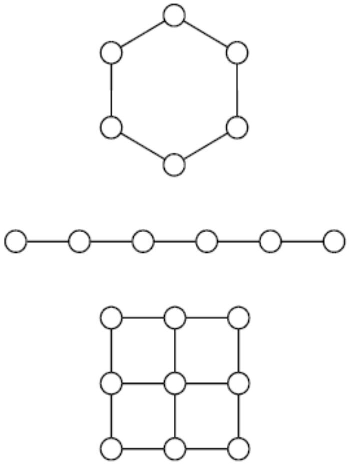
Activity 5: Deadlock Avoidance

You will quickly find that if you are "greedy" (hold onto their own doll as soon as they get them) then the group might not be able to attain its goal.
It may be necessary to emphasize that individuals don't "win" the game, but that the puzzle is solved when everyone has their dolls.



Activity 5: Deadlock Avoidance

- Try the activity with a smaller or larger circle.
- Have the participants come up with new rules.
- Carry out the activity without any talking.
- Try different configurations such as sitting in a line, or having more than 2 neighbours for some participants. Some suggestions are shown below:



Activity 5: Deadlock Avoidance

https://www.youtube.com/watch?v=WforXEBMm5k&feature=youtu.be

What strategies did the participants use to solve the problem?

How is this activity related to Computational Thinking?

Debrief/Feedback

Educational theories or frameworks aligned with the intent and purpose of unplugged activities

- Kolb’s experiential learning theory (1984)
  - unplugged activities can provide students concrete experiences into understanding how computing concepts work.
- Concrete-Pictorial-Abstract (Leong, Ho, & Cheng, 2015)
  - unplugged is in the “Concrete” stage, makes abstract concepts concrete
- Theories for Embodied cognition
  - mind and body works together to promote understanding.
  - Researchers (such as Fadjo, 2012; Román-González, Pérez-González, & Jiménez-Fernández, 2017; Sung, Ahn, & Black, 2017; Trory, 2016) have researched about embodied cognition in the development of CT

Educational theories or frameworks aligned with the intent and purpose of unplugged activities

- Constructionism posits that we learn best by constructing artifacts, algorithms and understanding
  - Researchers (such as Fowler, 2017; Futschek & Moschitz, 2010; Karadeniz, Samur, & Özden, 2013; Kotsopoulos et al., 2017; Malinverni & Parés, 2014; Schweikardt & Gross, 2007) have studied about the role of constructionism in the development of CT

Should there be a more concerted effort to study unplugged activities in formal computing education?

Should there be a more concerted effort to study unplugged activities in formal computing education?

- Yes!
- Why?
- (1) With the increased interest in computing education and developing CT, there is demand for pedagogies to teach computing to a wider audience of students, not just those who have traditionally self-selected to take a computing subject.
- (2) Unplugged activities "may" be an alternative way of developing CT in contrast to the conventional model of learning CT through programming.

Unplugged in Formal Education

- Unplugged activities have a proven track record in outreach programs
- Results are mixed as to how well they perform as part of a standard curriculum, and under what conditions (e.g., age group, subject).
- In formal education, more work needs to be done to theorize the foundations for unplugged pedagogy so that teaching and learning models can be developed and tested.

Unplugged in Formal Education

- Where should unplugged activities be integrated into a lesson sequence and for what purpose(s)?
  - To generate student interest on a topic?
  - To prime students' minds to learn a concept?
  - To anchor a students' understanding to some concrete experience?

Unplugged in Formal Education

- more classroom interventions need to be designed and tested for formal classroom learning to build evidence about
  - how to adapt or design activities to suit learning goals,
  - how to assess student learning,
  - how to train teachers to use the activities, and
  - how to connect activities to other course content, such as programming.

Call to address the demand for **evidence-based practices** to teach computational thinking and computing to a broader group of students than before!

Additional Resources

- Scratch.mit.org
- teachlondoncomputing.org
- Code.org
- csunplugged.org

The End

Acknowledgements to my colleagues Wendy Leong, Longkai Wu, Peter Seow, Mengleong How, Liu Liu and Bimlesh Wadhwa from NUS, Singapore

Contact: cheekit.looi@nie.edu.sg