This page is for

Teachers

Parents and carers

Digital Technologies @ Home
Unplugged activities for students

This activity is for years 7-8 and 9-10

Card Switches

Thanks to Girls Programming Network for providing this content

This activity teaches…

Computers need to sort things into order all the time, for example, if you use the ‘sort by’ filter on a website the computer has to read through everything on the page and re-sort it depending on your selection. This activity gets you to think like a computer, using specific steps to compare and sort cards.

As well as sorting into order, you are also going to scramble the cards up as best you can! The more switches it takes you to sort them back into order, the better job you did scrambling. For example, a set requiring only one switch to be sorted has not been scrambled very well.

It is targeted towards students in years 7-8 and 9-10, with different discussion points separating the band levels. It can be expanded to include Insertion Sort and Quicksort if required.

It is expected to take **one hour** of class time. Print pages 2-4 and 8 for students (instructions, example walkthrough and card cut-outs).

You will need…

Pen, paper and scissors

Getting started (read this with your child):

We’re going to try and scramble number cards as much as we can. After the cards are scrambled, we are going to count how many ‘pairwise switches’ it takes to sort them back into ascending order. What is a pairwise switch? It’s when you switch the places of a pair of cards that are next to each other. The more pairwise switches it takes to sort the cards out, the more scrambled the cards were.

First we will practice with some random scrambles, then we will try and make the most scrambled set possible.

Step by step

1. Have the student construct their cards and do their first unscrambling using the switches. If you would like to demonstrate try a scramble of 4, 1, 2, 3. It is a simple change (swap 4 and 1) but demonstrates the way the bubble sort ‘floats’ the highest number to the top after the first round of sorting.
2. Randomly scramble and sort a set of cards as a group, making sure you count the number of switches you make. The higher the number, the more scrambled the cards were.
3. Ask students to try and find the most scrambled sequence of cards. You are aiming to have the most number of pairwise switches you can.
4. When time is up, come back together and demonstrate the scramble and number of switches required.
5. Discuss which one is the ultimate Scramble and why.
6. Have the students turn the Pairwise Switch method into a pseudocode algorithm. (Consider specifying if the algorithms should be for an unknown number of cards and if it should count switches)
7. Have them validate their pseudocode by sorting a partner's scrambled cards with the partners pseudocode.

See a demonstration

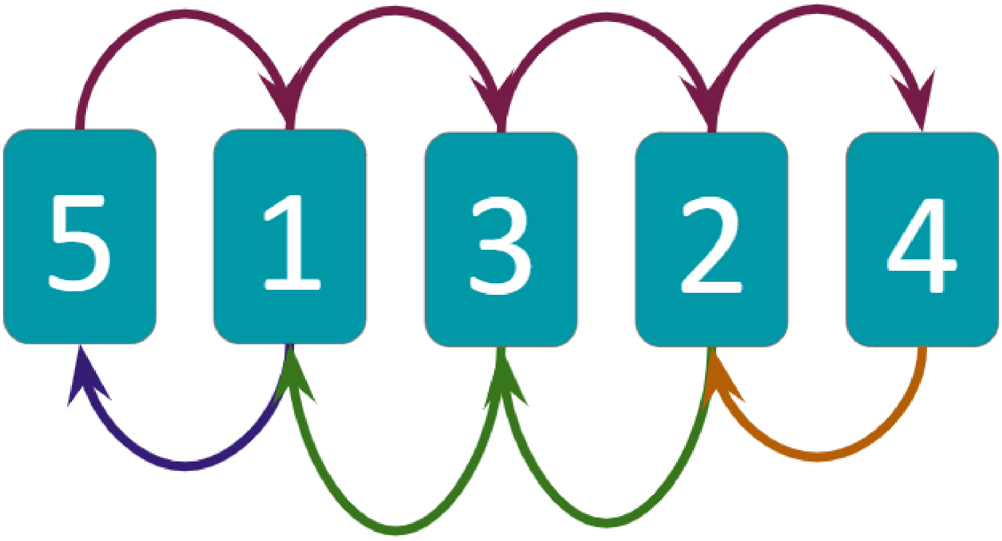
This video demonstrates the example below.
cmp.ac/switchesvid

This page is for

Students

Card Switches

Learn how to use pairwise switches to sort cards into order and count how scrambled they are.



How to sort your cards using pairwise switches

Step 1

Cut an A4 sheet of paper into 12 cards. Write the numbers 1 to 6 (one number per card) onto one side of each. This should give you 2 sets of cards that go from 1 to 6 (or print and cut out the last page of this activity).

Step 2

Scramble your cards and lay them out.

Step 3

It's time to unscramble using Pairwise Switches. To do a Pairwise Switch, start at the leftmost pair of cards. **Do a switch when the card on the left is bigger than the card on the right.** Record how many times you make a switch.

Step 4

Work your way across the cards. When you reach the end, check the order of the cards. If they aren't in order yet, go back to the start of the cards and work through again. Keep going back to start until they're in the right order!

Step 5

Now that you know how to sort them, try and find what combination of cards takes the most switched to unscramble. Try different combinations and see how many switches they take to solve. Make sure you keep a record of which combinations you try.

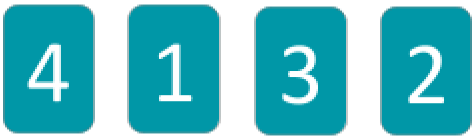
Step 6

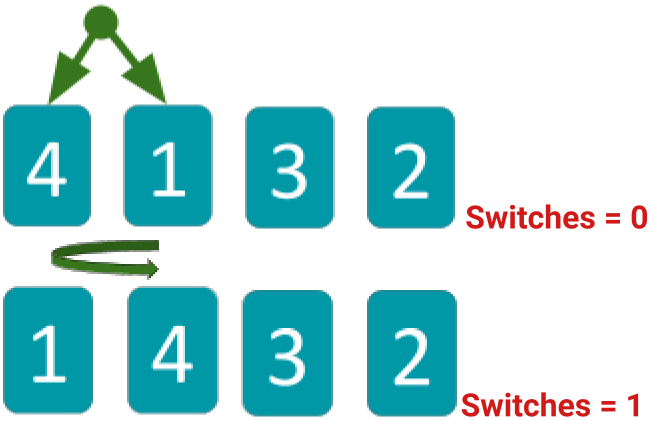
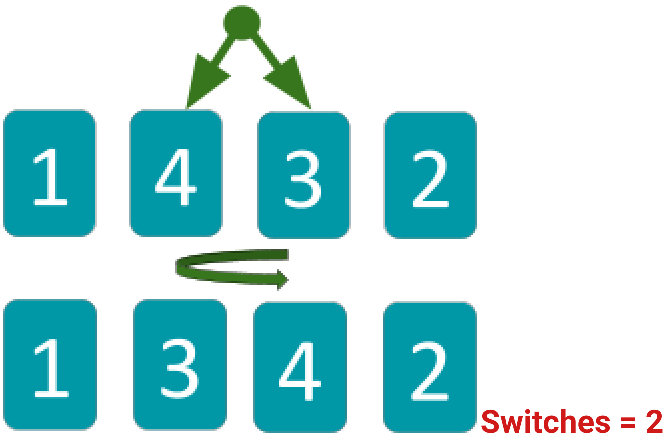
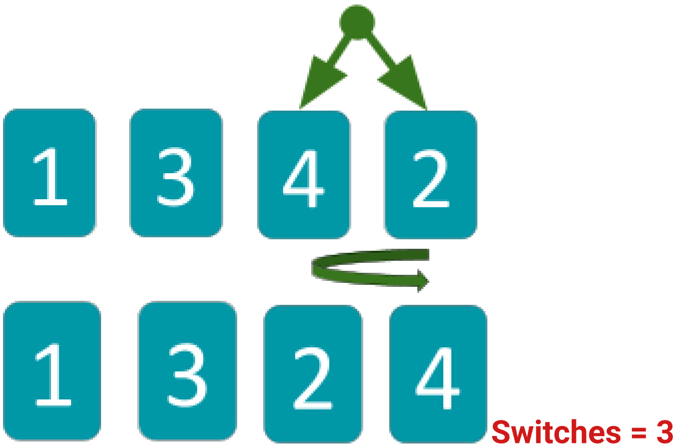
Now it's time to turn your pairwise switch method into pseudocode. See if you can write the pseudocode to sort your set of cards. Can you write the pseudocode to sort a set of cards that you don't know the length of?

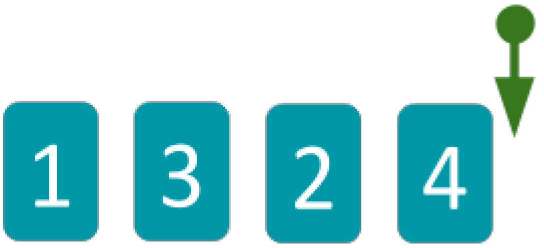
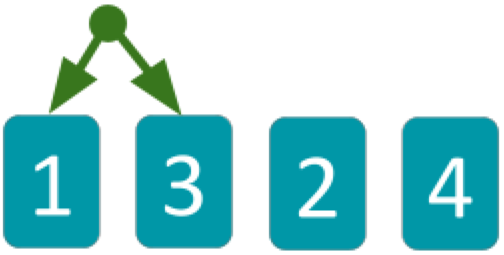
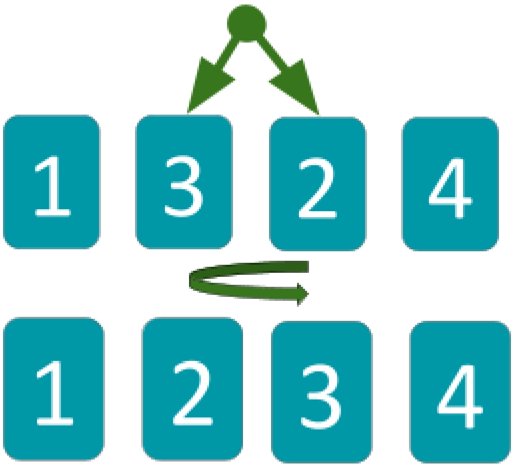
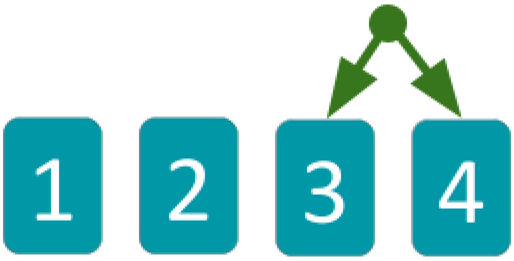
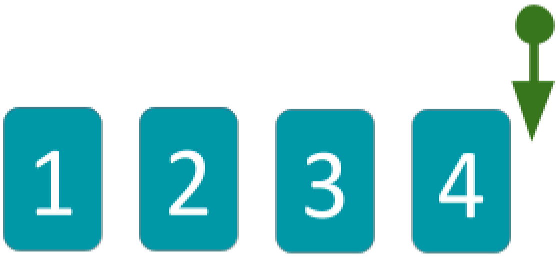
Step 7

Swap pseudocode with another student and trial each other's pseudocode: does it work? Can it be improved?

Let's unscramble one set of cards together.



Look at the first 2 cards, is 4 bigger than 1? Yes! Let's switch them.	
Look at the next 2 cards. Is 4 bigger than 3? Yes! Let's switch!	
Look at the next 2 cards. Is 4 bigger than 2? Yes! Let's switch!	

We are at the end of the row. Are the cards sorted? No. We need to go back to the start.	
Look at the first 2 cards again. Is 1 bigger than 3? No! So we leave the cards where they are.	
Next 2 cards. Is 3 bigger than 2? Yes! Let's switch.	 **Switches = 4**
Is 3 bigger than 4? No leave them	
Are the cards sorted? Yes! We are done! And it took us **4** pairwise switches to sort the cards!	

This page is for

Teachers

Parents and carers

Answer key

Choose if you want to print this for your kids or keep it to yourself!

The combination that will require the most pairwise switches is a reversed order of the original.

This will take **15 switches** to sort into ascending order.

To understand why, you should know that Bubble Sort is so named because it ‘floats’ the highest number to the top with each pass over the cards (like a bubble).

With that in mind, the way to make it take the longest is to move the highest numbers the furthest from the top of the list as you can.

Once the highest number is the furthest from the top, the second highest should be second furthest from the top etc etc.

There may be disagreement from students that reverse order is not exactly ‘scrambled’, it’s a good opportunity to ask them why they think that. It could be because the linguistic meaning of ‘scrambled’ to them means ‘without any order’. You can explain that in some cases, that is what scrambled means. But in this case, scrambled is being defined as “as many pairwise swaps away from the original order as possible.“

Keep the conversation going - answers (pg 7)

(9-10) Can you write your sort as an algorithm?

This could be done using pseudocode or in a programming language students are familiar with.

A Python example:

```
#Define the Bubble sort function, which takes an array of objects
def bubbleSort(arr):
    n = len(arr)
    # Traverse through all array elements
    for i in range(n):
        # Last i elements are already in place
        for j in range(0, n-i-1):
            # traverse the array from 0 to n-i-1
            # Swap if the element found is greater
            # than the next element
            if arr[j] > arr[j+1] :
                arr[j], arr[j+1] = arr[j+1], arr[j]
```

(9-10) How could you optimise this bubble sort to be faster?

By adding a check to stop you from checking the already sorted cards. For example, a variable called ‘unsorted_length’ that gets shorted by 1 every time you go over the cards

(9-10) Is this a fast way to sort these cards? Can you think of a faster way?

This is designed to make the students think about alternatives to bubble sort (which is actually an inefficient sort, taking a lot of time)

(9-10) starting from a sorted set, what's the most scrambled you can make the cards using only 3 changes? Why?

This refers to the solution which states that the best way to scramble for a bubble sort is to move the highest number to the position farthest from its sorted position, the same for second highest , etc.

(9-10) Research Insertion Sort and Quicksort. Can you demonstrate those sorts with cards? What is the highest scramble combination you can make for these sorts? Watch the video link and read the explanation resources cited above.

- Insertion Sort - The worst case for Insertion Sort is actually the same as the Bubble sort. The set of cards sorted into reverse order will take the longest amount of time to sort. Each time the insertion sort collects the next card, it has to swap it with every preceding card, so while it does not cover the entire set of cards over and over, it still has to touch and swap each e=card in the dec.
- Quicksort - Given the random (sometimes, depends on implementation) assignment of the pivot position, it is not just a matter of the cards being in a certain order, it is also a matter of the pivot position. In a smaller set like the cards it doesn't make a huge amount of difference. In a larger set it can make a relatively unshuffled set take much longer to sort through.

  The worst case scenario for a quicksort would be a reverse order sorted set of cards with a pivot that sits at the highest or lowest point each iteration.

This page is for

Parents and carers

Teachers

Want more?

Here are some further activities, online resources, assessment ideas and curriculum references.

Keep the conversation going

- What strategy did you use to scramble your cards?
- (7-8) If these cards were letters instead of numbers, how would you sort them?
- (7-8) If these cards were animals instead of numbers, how would you sort them?
- (7-8) Did you know that you were using a Bubble Sort to sort these cards?
- (7-8) Why do you think it is called bubble sort?
- (9-10) Can you write your sort as an algorithm in a programming language?
- (9-10) How could you optimise this bubble sort to be faster?
- (9-10) Is this a fast way to sort these cards? Can you think of a faster way?
- (9-10) starting from a sorted set, what's the most scrambled you can make the cards using only 3 changes? Why?
- (9-10) Research Insertion Sort and Quicksort. (Links below, or you can find your own) Can you demonstrate those sorts with cards? What is the highest scramble combination you can make for these sorts?

Keep learning

For High School students interested in different sorting algorithms

- An text based introduction to quicksort https://medium.com/karuna-sehgal/a-quick-explanation-of-quick-sort-7d8e2563629b
- An text based introduction to Insertion sort https://medium.com/karuna-sehgal/an-introduction-to-insertion-sort-16b97619697
- A good overview video of the sorts , but with an emphasis on time taken to do them. https://www.youtube.com/watch?v=WaNLJf8xzC4

Some more resources for students interested in watching sorts in action.

- http://sorting.at/
- https://www.youtube.com/user/AlgoRythmics
- https://visualgo.net/bn/sorting

For teachers creating a portfolio of learning or considering this task for assessment

Present students with a sequence of objects that need sorting into an order. Have the student sort the objects and determine how many swaps are needed to complete the sort.

Note: It is not recommended to use numbers or letters as sorting objects as there are a number of tools available online that will sort and count the swaps for you. Choose a random set of images, provide the students with the scrambled order to work from and the unscrambled one for reference.

Linking it back to the Australian Curriculum: Digital Technologies

Algorithms

(7-8) Design algorithms represented diagrammatically and in English, and trace algorithms to predict output for a given input and to identify errors (ACTDIP029 - see cmp.ac/algorithms)

(9-10) Design algorithms represented diagrammatically and in structured English and validate algorithms and programs through tracing and test cases (ACTDIP040 - see cmp.ac/algorithms)

Refer to aca.edu.au/curriculum for more curriculum information.

Students

1	1
2	2
3	3
4	4
5	5
6	6