Shell commands: tsort

- tsort **lets you sort data based on any ordering criteria you wish**
    - **In fact, <u>the data provide the definition of the ordering criteria</u>**
        - So tsort does not [necessarily] sort in alphabetic or collating sequence
- **Suppose a couple, let’s call them John and Jane, are discussing where they are interested in taking their vacation**
    - **They agree to use** tsort **to help them make their final decisions; a conversation might go like this:**

So where would you like to travel to?

Well, I’ve always wanted to see Germany and France.

Mmm. Me too. Which would you put higher on your list?

Ahh, France, I think.

OK. (enters this in a file called ’vacation’: France Germany)
I’d like to visit Indonesia or Hawaii.

Which is higher on your list?

I think Indonesia. (enters: Indondesia Hawaii)

How would you rank Indonesia compared to France?

I guess I’d put Indonesia first. (enters: Indonesia France)

Oh, don’t forget England.

Right, right. (enters: England England)

And Japan. In fact, put Japan ahead of France.

OK. (enters: Japan France)

Well, let’s see where we are so far.

Shell commands: tsort, continued

❒ **The file at this point reads:**

```
France       Germany
Indonesia    Hawaii
Indonesia    France
England      England
Japan        France
```

❒ **If John and Jane used** tsort **now, it would look like this:**

```
tsort     vacation

Indonesia
Japan
France
Germany
Hawaii
England
```

❒ **By extension, you can see [hopefully] how this could be very handy, especially when the list gets quite long**

Shell commands: tsort, continued

❒ tsort **works like this:**

- ♦ **The input consists of pairs of entries; these pairs define the ordering you want**
  - ✗ If the values are different, the first is to precede the second
  - ✗ If the values are the same, it is just an indication of presence
- ♦ tsort **creates a list using this ordering and writes it to stdout**
  - ✗ Duplicates are omitted
  - ✗ Entries that are not ordered appear at the end
  - ✗ If you have a circular / logic error in your data pairings, you get an error message
    - ➤ For example, if you had **France Germany** as one entry and **Germany France** as another
    - ➤ It even catches more circular errors