



# Bubble sort

**The easiest, least efficient sort**

---



---

# Bubble sort

- The bubble sort goes through the list looking at adjacent elements and switching them if necessary.
  - Each time through is called a ‘**pass**’
  - Each time it looks at two elements to see if they need switching is called a ‘**comparison**’
  - Each switching is called a ‘**swap**’
-



# Example

- h n a b d is our list
- compare h n, they are in order
- h n a b d
- compare n a, they should be switched
- h a n b d, compare n b, so switch
- h a b n d. compare n d, so switch
- h a b d n – End of Pass 1



---

## Pass 2

- h a b d n - compare h a so switch
  - a h b d n - compare h b so switch
  - a b h d n - compare h d so switch
  - a b d h n - compare h n , in order
-



---

## Pass 3

- a b d h n , a b in order
  - a b d h n , b d in order
  - a b d h n , d h in order
  - a b d h n , h n in order
- 
- When no swaps are done we know we are finished.
-



---

# Efficiency

- We want to minimize the number of comparisons and swaps.
  - So we count them to see how efficient our sort was.
  - In this case, our 3 pass each took 4 for **12 comparisons**, and we made **6 swaps**.
-