

Summary

Summary

► $\text{current} = \text{current} \rightarrow \text{next}$

Summary

- ▶ `current = current→next`
- ▶ The beautiful Binary Tree Search. A real need for recursion.

Summary

- ▶ `current = current→next`
- ▶ The beautiful Binary Tree Search. A real need for recursion.
- ▶ Understanding why `qsort()` is $O(n \log n)$.

Summary

- ▶ `current = current→next`
- ▶ The beautiful Binary Tree Search. A real need for recursion.
- ▶ Understanding why `qsort()` is $O(n \log n)$.
- ▶ von Mises paradox. The joy of hashing.

Summary

- ▶ `current = current→next`
- ▶ The beautiful Binary Tree Search. A real need for recursion.
- ▶ Understanding why `qsort()` is $O(n \log n)$.
- ▶ von Mises paradox. The joy of hashing.
- ▶ The realisation that infix notation is evil.

Summary

Summary

- ▶ Things we skipped in ANSI C : varargs, ##, unions, bit fields

Summary

- ▶ Things we skipped in ANSI C : varargs, ##, unions, bit fields
- ▶ Things we skipped in Algorithms: Skip lists, Self-Balancing Trees, MinMax Heaps, Cuckoo Hashing

Summary

- ▶ Things we skipped in ANSI C : varargs, ##, unions, bit fields
- ▶ Things we skipped in Algorithms: Skip lists, Self-Balancing Trees, MinMax Heaps, Cuckoo Hashing
- ▶ *Hello World*

Summary

- ▶ Things we skipped in ANSI C : varargs, ##, unions, bit fields
- ▶ Things we skipped in Algorithms: Skip lists, Self-Balancing Trees, MinMax Heaps, Cuckoo Hashing
- ▶ *Hello World*
- ▶ Why I teach this unit.

Summary

- ▶ Things we skipped in ANSI C : varargs, `##`, unions, bit fields
- ▶ Things we skipped in Algorithms: Skip lists, Self-Balancing Trees, MinMax Heaps, Cuckoo Hashing
- ▶ *Hello World*
- ▶ Why I teach this unit.
- ▶ Projects in Computer Vision / Deep Learning