Computational thinking and Problem-solving

A-Level Computer Science · Vocabulary in context

Word	Example
insertion sort /ɪn'sɜ:ʃn sɔ:t/	**Insertion sort** puts each new item in the right place.
bubble sort /'bʌbl sɔ:t/	**Bubble sort** swaps two items next to each other if they are in the wrong order.
binary search /'baɪnəri sɜ:tʃ/	**Binary search** halves the list each time.
array /ə'reɪ/	An **array** stores many values under one name.
linear search /'lɪnɪə sɜ:tʃ/	A **linear search** checks each item one by one.
linked list /lɪŋkt lɪst/	In a **linked list**, each item points to the next one.
in place /ɪn pleɪs/	Bubble sort works **in place**: it sorts the list without making a copy.
stable /'steɪbl/	A **stable** sort keeps equal items in the same order.
stack /stæk/	The last item put on the **stack** is the first one taken off.
queue /kju:/	A **queue** is first in, first out.
node /nəʊd/	Each **node** holds a value and a pointer.
pointers /'pɔɪntəz/	The tree uses left and right **pointers**.
binary tree /'baɪnəri tri:/	Each node in a **binary tree** has at most two children.

exercises.lol exercises.com.cn

Word	Example
dictionary /'dɪkʃənəri/	A **dictionary** stores pairs of keys and values.
circular queue /'sɜ:kjʊlə kju:/	A **circular queue** reuses the space at the front.
free list /fri: lɪst/	The **free list** keeps track of empty nodes.
time complexity /taɪm kəm'pleksɪti/	The **time complexity** shows how the time grows with the input.
Big-O notation /bɪg əʊ nəʊ'teɪʃn/	Linear search is O(n) in **Big-O notation**. O(n)
space complexity /speɪs kəm'pleksɪti/	The **space complexity** shows how much memory the algorithm needs.
recursion /rɪ'kɜ:ʃn/	**Recursion** means a function calls itself.
call stack /kɔ:l stæk/	Each function call is added to the **call stack**.
base case /beɪs keɪs/	The **base case** stops the recursion.
recursive case /rɪ'kɜ:sɪv keɪs/	In the **recursive case**, the function calls itself again.
factorial /fæk'tɔ:rɪəl/	The **factorial** of 4 is $4 \times 3 \times 2 \times 1 = 24$. 4 $4 \times 3 \times 2 \times 1 = 24$
divide-and-conquer /dɪ'vaɪd ænd 'kɒŋkə/	Binary search is a **divide-and-conquer** algorithm.
general case /'dʒenərəl keɪs/	The **general case** is the rule for all the other inputs.
parameters /pə'ræmɪtəz/	The procedure has two **parameters**: a name and an age.
stack overflow /stæk ˌəʊvə'fləʊ/	Recursion with no base case causes a **stack overflow**.

Word	Example
memoisation /ˌmeməʊaɪˈzeɪʃn/	**Memoisation** saves answers so they are not worked out again.
local variables /ˈləʊkl ˈveərɪəblz/	**Local variables** disappear when the function ends.
stack frame /stæk freɪm/	Each call gets its own **stack frame**.
return address /rɪˈtɜːn əˈdres/	The **return address** shows where to go back after the call.

exercises.lol exercises.com.cn