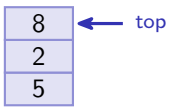
Data Types & Structures

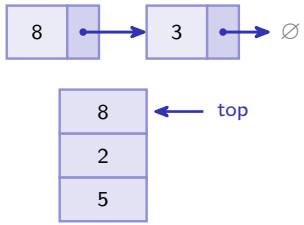
AS Computer Science ▪ Topic 10

A-Level Computer Science



What we will cover

1. Data types & records
2. Arrays
3. Files
4. Stacks, queues, linked lists
5. Implementing with arrays
6. Recap

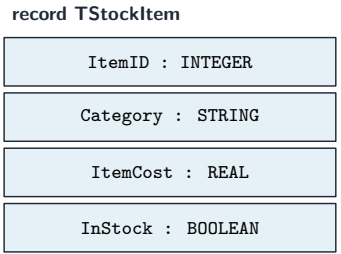


Data types and records

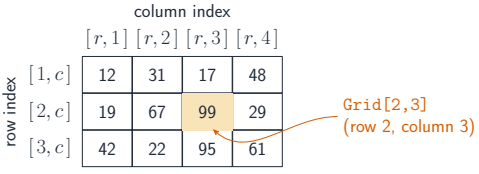
Pick the smallest precise **data type** 数据类型:

- `INTEGER`, `REAL`, `STRING`
- `CHAR`, `BOOLEAN`, `DATE`

A **record** 记录 groups several types under one name (dot notation `Item1.Category`).

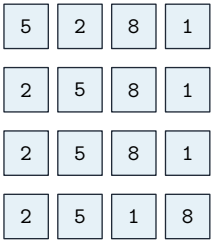


Arrays



2-D: a table, indexed `[row, column]`

Searching and sorting



Worked example: find the largest

```
DECLARE Marks : ARRAY[1:4] OF INTEGER
Max ← Marks[1]
FOR i ← 2 TO 4
  IF Marks[i] > Max THEN
    Max ← Marks[i]
  ENDIF
NEXT i
OUTPUT Max
```

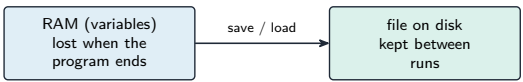
[1]	[2]	[3]	[4]
7	3	9	2

i	Marks[i]	Max
–	–	7
2	3	7
3	9	**9**
4	2	9

output: **9**

Start `Max` at the first element, not at 0 – then compare from the second.

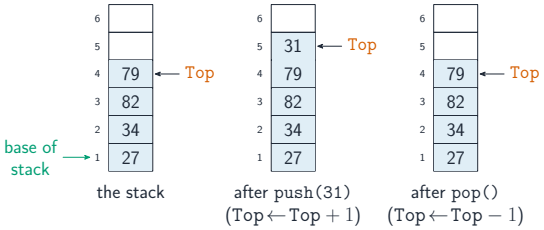
Files



A **file** 文件 keeps data on disk between runs (RAM is lost at exit).

- `OPENFILE` for READ/WRITE/APPEND
- `EOF` tests the **end of file** 文件结束
- always `CLOSEFILE` – or buffered writes are lost

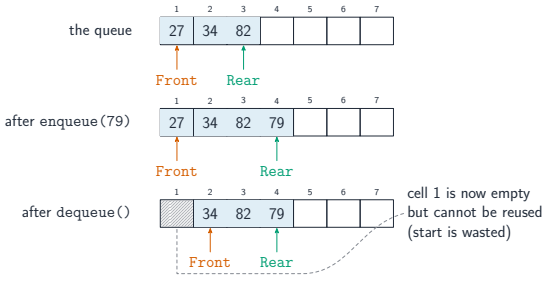
Abstract data types: the stack



A **stack** 栈 is **LIFO** 后进先出: `push` and `pop` happen at the top – undo history, procedure calls.

A stack is last-in, first-out

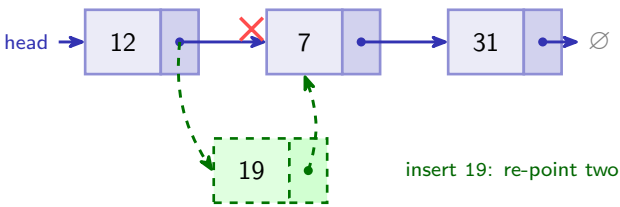
Abstract data types: the queue



A queue is first-in, first-out

A **queue** 队列 is **FIFO** 先进先出: enqueue at the back, dequeue at the front – print **spooling** 排队打印, scheduling.

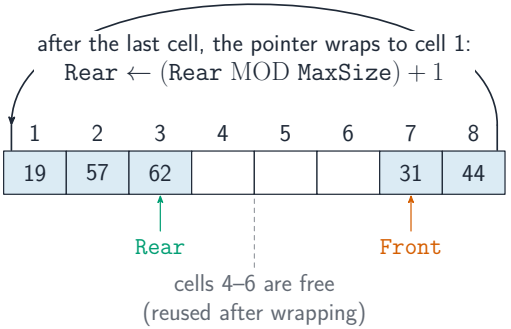
Linked list



A **linked list** 链表 chains **nodes** 节点, each a value + a **pointer** 指针 to the next.

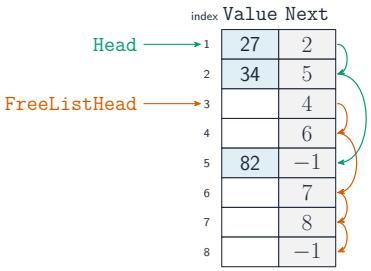
- + cheap insert/delete – nothing shifts
- − slow random access

ADTs with arrays



- stack: a `Top` index; full = **overflow** 溢出, empty = **underflow** 下溢
- queue: a **circular array** 循环数组 – pointers wrap with `(p MOD MaxSize) + 1`

Linked list in an array



Store nodes in an array of records, each with a `Next` index. A **free list** 空闲列表 chains the unused slots – linked-structure flexibility with static allocation.

Topic 10 – key takeaways

1 Records & arrays

records mix types; arrays hold one type

2 Files

persist data; open, read/write, close

3 Stack & queue

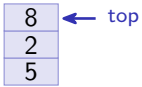
LIFO vs FIFO

4 Linked list

nodes with pointers; cheap insert/delete

Check yourself

1. Which holds several different types together?
2. Which ADT is FIFO?
3. Why use a circular array for a queue?
4. One advantage of a linked list over an array?



Answers **1.** a record **2.** a queue **3.** reuse the wrapped-round cells **4.** cheap insertion/deletion