

# Python 3 Cheatsheet: Specialized Algorithms

---

## Fast Power

Calculate  $x^n$  in  $O(\log n)$ .

```
def fastPow(x, n):
    res = 1
    while n > 0:
        if n % 2 == 1:
            res *= x
        x *= x
        n //= 2
    return res
```

## Basic Calculator (Stack)

Evaluate  $3 + 2 * 2$ .

```
def calculate(s):
    stack, num, sign = [], 0, '+'
    for i, c in enumerate(s + '+'):
        if c.isdigit():
            num = num * 10 + int(c)
        if c in '+-*/' or i == len(s):
            if sign == '+': stack.append(num)
            elif sign == '-': stack.append(-num)
            elif sign == '*': stack.append(stack.pop() * num)
            elif sign == '/': stack.append(int(stack.pop() / num))
            sign, num = c, 0
    return sum(stack)
```

## Reservoir Sampling

Pick random element from stream of unknown size.

```
# k=1
count = 0
res = None
for x in stream:
    count += 1
    if random.randint(1, count) == 1:
        res = x
```

## Cyclic Sort

Sort 1 to N in  $O(N)$  time &  $O(1)$  space.

```
i = 0
while i < len(nums):
    j = nums[i] - 1 # Correct index
    if nums[i] != nums[j]:
        nums[i], nums[j] = nums[j], nums[i]
    else:
        i += 1
```

## Boyer-Moore Voting (Majority Element)

```
count = 0
candidate = None
for num in nums:
    if count == 0:
        candidate = num
    count += (1 if num == candidate else -1)
return candidate
```