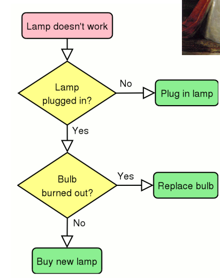
Algorithms

CS 231
Dianna Xu

1

Algorithm

- Step-by-step method for performing some action
- An algorithm is a sequence of finite instructions, often used for calculation (Wikipedia)



2

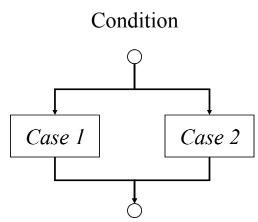
Pseudo code

- Variables
  - Memory storage
  - Size and type
- Assignment statement
  - x **:=** expr

3

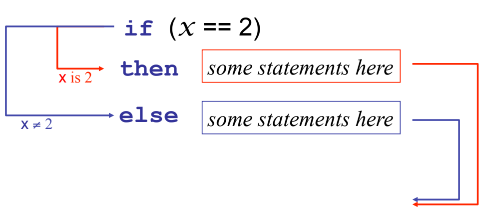
Conditional

- **if** (condition) **then** s1 **else** s2



4

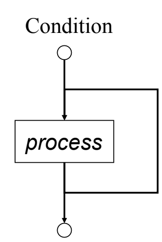
Structure



5

while loop

while (condition)
body
end while



6

for loop

```
for (i:= initial expr to final expr )
      body
next i
```


Greatest common divisor

- *a*, *b* ϵ *Z*, *a*≠0, *b*≠0, gcd(*a*, *b*) is the integer d with the following properties:
  - *d* | *a* and *d* | *b*
  - ∀ *c* ϵ *Z*, if *c* | *a* and *c* | *b*, then *c* ≤ *d*

Lemmas

- Lemma 1:
  - If *r* is a positive integer, then gcd(*r*, 0) = *r*
- Lemma 2:
  - Given *a*, *b* ϵ *Z*, with *b*≠0, and *q*, *r* ϵ *Z* such that: *a* = *bq* + *r*
  - Then gcd(*a*, *b*) = gcd(*b*, *r*)
    - Show gcd(*a*, *b*) ≤ gcd(*b*, *r*)
    - And gcd(*b*, *r*) ≤ gcd(*a*, *b*)

The Euclidean Algorithm

- By hand: gcd(123, 456)
  - 456 = 123*3 + 87 → gcd(456, 123) = gcd(123, 87)
  - 123 = 87*1 + 36 → gcd(123, 87) = gcd(87, 36)
  - 87 = 36*2 + 15 → gcd(87, 36) = gcd(36, 15)
  - 36 = 15*2 + 6 → gcd(36, 15) = gcd(15, 6)
  - 15 = 6*2 + 3 → gcd(15, 6) = gcd(6, 3)
  - 6 = 2*3 → gcd(6, 3) = 3

Algorithm: Euclidean

```
Input: A, B (A, B in Z, A > B ≥ 0)
Algorithm Body:
  a:= A, b:= B, r:= B
  while (b ≠ 0)
     r:=a mod b
     a:=b
     b:=r
  end while
Output: gcd := a
```


Properties of an Algorithm

- Input
- Output
- Definiteness
- Correctness
- Finiteness
- Effectiveness
- Generality

Finding the Max Element

- Give an algorithm for finding the largest value in a finite sequence of integers

Input: $a_1, a_2 \ldots a_n \in Z$

Algorithm Body:

$max :=$ a_1

for ($i :=$ 2 **to** n)

  if ($max < a_i$) **then** $max := a_i$

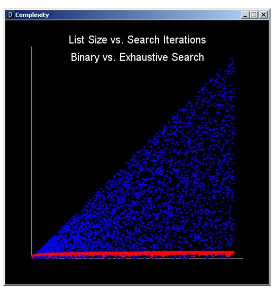
next i

Output: max

Searching

- Locating an element in an ordered list
- 1 2 3 5 6 7 8 10 12 13 15 16 18 19 20 22
- Linear Search – Sequential Search
- Binary Search
  - 1 2 3 5 6 7 8 10 12 13 15 16 18 19 20 22
  - 12 13 15 16 18 19 20 22
  - 18 19 20 22
  - 18 19

Running Time



Sorting

- Ordering the elements of a list
- Bubble Sort
  - compare each pair and swap if necessary
- Insertion Sort
  - the front of the list is kept in order
  - the sorted list starts with 1 element, the first
  - each successive element is compared and inserted into the correct position in the sorted list

Greedy Algorithm

- Optimization problems – to find a solution to the given problem that either maximizes or minimizes the value of some parameter
- The simplest approach – greedy
  - select the best available choice at each step
  - does not consider consequences of all sequences
  - solution is not always optimal