Introduction to Algorithms

ICS 2– Introduction to Computer Programming

What is an algorithm?

… a well-defined procedure that allows an agent to solve a problem.

Note: often the agent is a computer or a robot…

Example algorithms

- Cooking a dish
- Making a peanut-butter jelly sandwich
- Shampooing hair
- Programming a VCR
- Making a pie

Example

Is this an algorithm?

- Step 1: Wet hair
- Step 2: Lather
- Step 3: Rinse
- Step 4: Repeat

Would you manage to wash your hair with this algorithm? How about a robot? Why (not)?

Algorithms

An algorithm must:

1. Be well-ordered and unambigous
2. Each operation must be effectively computable
3. Terminate.

Example

Problem: Find and print the 100th prime number

- A prime number is a whole number not evenly divisible by any other number other than 1 and itself

Algorithm (?):

1. Generate a list of all prime numbers L1, L2, L3,…
2. Sort the list into ascending order
3. Print out the 100th element in this list

Is this an algorithm?

Types of Operations

- Basic operations
  - Wet hair
  - Rinse
- Conditional operations
  - If batter is too dry add water
- Repeat/looping operations
  - Repeat step 1 and 2 three times
  - Repeat steps 2,3,4,…10 until batter becomes soft.

Algorithm

- How to come up with an algorithm?
  - That is, how to discover an algorithm underlying a problem?
  - Problem solving

- How to represent an algorithm?
  - In English??
  - In a programming language??

Example

- Problem: Given two positive integers, compute their greatest common divisor

- Euclid's algorithm:
  - Step 1: Get two positive integer values from the user
  - Step 2: Assign M and N the value of the larger and smaller of the two input values, respectively
  - Step 3: Divide M by N, and call the remainder R
  - Step 4: If R is not 0, then assign M the value of N, assign the value of R, and return to step 2; otherwise, the greatest common divisor is the value currently assigned to N

Coming up with algorithms..

- How do people think????

- Puzzle:
  - Before A, B, C and D ran a race they made the following predictions:
    - A predicted that B would win
    - B predicted that D would be last
    - C predicted that A would be third
    - D predicted that A's prediction would be correct.
  - Only one of these predictions was true, and this was the prediction made by the winner.

  In what order did A, B, C, D finish the race?

Example

- Problem: Adding two n-digit numbers

```
    7597831 +
    1287525
------------------
    8885356
```

How would you write an algorithm to solve this problem? Assume the basic operation is adding one-digit numbers.

Expressing algorithms

- Is natural language good?
  - For daily life, yes…but for CS/IT it lacks structure and would be hard to follow
  - Too rich, ambiguous, depends on context

- How about a programming language?
  - Good, but not when we try to solve a problem... we want to think at an abstract level
  - It shifts the emphasis from how to solve the problem to tedious details of syntax and grammar.

Pseudocode

- Pseudocode = English but looks like programming
- Good compromise
  - Simple, readable, no rules, don't worry about punctuation.
  - Lets you think at an abstract level about the problem.
  - Contains only instructions that have a well-defined structure and resemble programming languages

Pseudocode

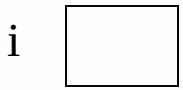
- Basic (primitive) operations
  - Read the input from user
  - Print the output to the user
  - Cary out basic arithmetical computations
- Conditional operations
  - Execute an operation if a condition is true
- Repeat operations
  - Execute a block of operation multiple times until a certain condition is met

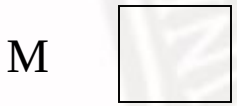
Variables

Variable

- A named memory location that can store a value
- Think of it as a box into which you can store a value, and from which you can retrieve a value

Examples:

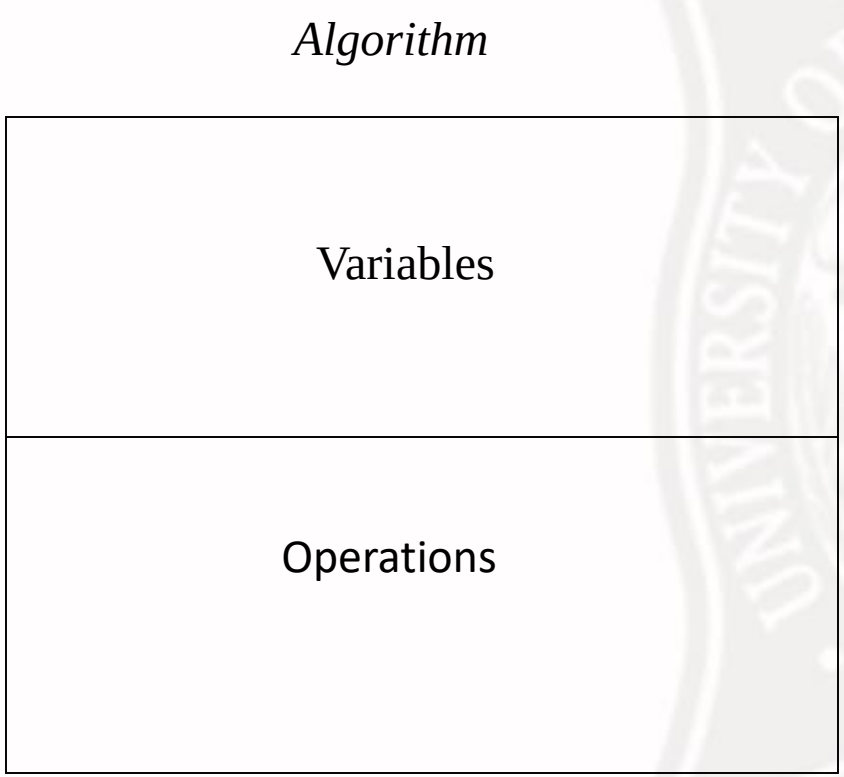




Example of operations

- `Set the value of i  to 3`
- `Set the value of M to i*3 + 12`
- `Set the value of i to i+10`

A model for visualizing an algorithm



An algorithm consists of operations that involve variables

Primitive operations

- Get input from user
  - `Get the value of x from user`
- Assign values to variables using basic arithmetic operations
  - `Set the value of x to 3`
  - `Set the value of y to x/10`
  - `Set the value of z to x +25`
- Print output to user
  - `Print the value of y, z to the user`

Example 1

Problem: For any three numbers input by the user, compute their sum and average and output them

Example of algorithm in pseudocode:

```
Variables: a,b,c, sum, avg
```

- Get the values of a, b, c from user
- Set avg to (a+b+c)/3
- Set sum to (a+b+c)
- Print sum and avg

Example 2

Problem: Given any value of radius from the user, compute and print the circumference of a circle with that radius

Algorithm in pseudocode:

```
variables: r, c
1. Get the value of r from user
2. Set c to 2 * pi * r
3. Print "The circumference of your
circle is " c
```