

Program Design

Lecture 2

Programs and Algorithms

- A Program is a set of instructions written in a particular programming language that the computer is able to understand.
- An algorithm consists of a set of explicit unambiguous and finite steps, when carried out for a given set of initial conditions, produce the corresponding output and terminate in a finite time.
- Essentially and program is a set of algorithms written in specific programming languages.

Syntax and Semantics

- Syntax dictates the structure of sentences or commands.
- Semantics refers to the meaning of what is being stated in the language.
- Syntax of computer languages ensure that statements are unambiguous – unlike natural languages.

Examples of Ambiguity

Syntactical ambiguity:

“The chicken is ready to eat”



Examples of Ambiguity

Syntactical ambiguity:

“Why don’t they serve seagulls in restaurants?”



Examples of Ambiguity

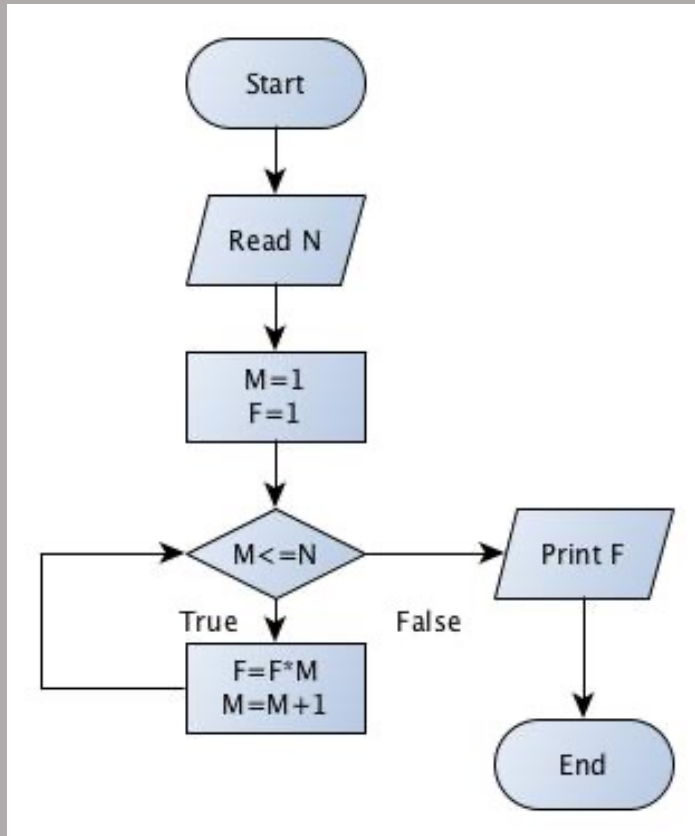
Lexical Ambiguity:

“Did you see the match last night”



- In this course we are more concerned with semantics than syntax.
- As part of design we will express the semantics in flowcharts and pseudo code.

Flowchart



Pseudocode

Program

total = 0

grade_counter = 1

While grade_counter <= 10

Input next_grade

total = total + next_grade

**grade_counter = grade_counter
+1**

End While

class_average = total /10

Print class_average

End Program

Algorithm

- An algorithm consists of a set of explicit unambiguous and finite steps, when carried out for a given set of initial conditions, produce the corresponding output and terminate in a finite time.

Biggest Number

- Which is the biggest number from these 3 values?
- 17, 64, 18

How did you do it?

- Can you give a list of instructions that if followed will give us the biggest number?

Dictionary Algorithm

- Find a word in a dictionary – there is no index or table of contents.
- How do we find a particular word in such a large set?
- We take advantage of the structured order of the data.
- On the other hand if we knew the meaning of the word and were trying to find the word a dictionary would not be too much use. That is a more difficult problem!
- Suggested algorithm?

Instructions

1. To start put your left hand on the front cover and right hand on the back cover.
2. Open the Dictionary approximately on the middle page between your hands. Check first word on left page and last word on the right page.
3. If the word we are looking for is between the two words stop and find the word on the page and stop when found.
4. If the word is in the right hand part of the book mark the current page with you left hand, if it is in the left half mark page with your right hand (the word will be in the section your hands). Go to step 2

Dictionary Look Up

Word = 'antidisestablishmentarianism'

LeftPage = 1

RightPage = LastPage

found = 0

WHILE Word not found

 FirstWord = first word on LeftPage

 LastWord = last word on RightPage

 IF (FirstWord is less than or equal to Word and LastWord is greater than or equal Word)

 Find word on page

 found = 1

 ENDIF

 CurrentPage = round((LeftPage + RightPage)/2)

 IF (FirstWord is greater than Word)

 RightPage = CurrentPage

 ENDIF

 IF (LastWord is less than Word)

 LeftPage = CurrentPage

 ENDIF

ENDWHILE

- Loop – this loop has a condition to end it – while some statement is true keep executing the loop.
- To make this loop useful we have had to form the instruction in a way that it is repeatable – every repetition getting closer to the goal.
- Inside the loop there are more conditions – depending on the ‘truth’ of the conditions different operations may be performed.
- Find word on page – is another algorithm here we have just put in a high-level instruction.

Other Solutions

- Other possible solutions –
 - Can we refine the algorithm slightly?
 - Brute force - start at the first word and keep checking in sequential order until
 - Use the first letter to aid our guess
 - Assumptions about distribution

This Course

- How can we represent data to make it useable on a computer for this algorithm? We don't give the computer a book do we?
- Learning how to use a series of explicit instructions to solve problems
 - Explicit unambiguous statements
 - Conditional statements
 - Iterations