CS 10
INTRODUCTION TO COMPUTER SCIENCE

CHAPTER 1

WHAT IS COMPUTER SCIENCE ?

MISCONCEPTIONS:
- THE STUDY OF COMPUTERS.
- PROGRAMMING AND PROGRAMMING LANGUAGES.
- APPLICATIONS SOFTWARE.

A WIDELY ACCEPTED DEFINITION OF COMPUTER SCIENCE WAS GIVEN BY NORMAN GIBBS AND ALLEN TUCKER:

COMPUTER SCIENCE IS THE STUDY OF ALGORITHMS, ESPECIALLY THEIR
1.) MATHEMATICAL PROPERTIES
    CORRECTNESS, TIME & SPACE EFFICIENCY
2.) HARDWARE REALIZATIONS
    ARCHITECTURE
3.) SOFTWARE REALIZATIONS
    PROGRAMMING & PROGRAMMING METHODOLOGIES
4.) APPLICATIONS
    NUMERICAL ANALYSIS, DATABASE MANAGEMENT...

WHAT IS AN ALGORITHM ?

INFORMAL DEFINITION: A STEP BY STEP PROCEDURE WHICH SOLVES SOME SPECIFIC PROBLEM.

EXAMPLES MIGHT INCLUDE A CAKE RECIPE, INSTRUCTIONS FOR BALANCING YOUR CHECK BOOK, OR FILLING OUT YOUR TAX RETURN.

EXAMPLE

ADD TWO 3-DIGIT NUMBERS:

$$\begin{array}{r} 493 \\ +751 \\ \hline \end{array} \qquad \begin{array}{r} 617 \\ +945 \\ \hline \end{array}$$

HOW CAN WE SPECIFY THIS ALGORITHM?

ALGORITHM

GIVEN: $m \geq 1$ AND TWO m DIGIT INTEGERS $(a_{m-1}\, a_{m-2} \cdots a_1\, a_0)$ AND $(b_{m-1}\, b_{m-2} \cdots b_1\, b_0)$,

FIND: THEIR SUM $(c_m c_{m-1} \cdots c_0) = (a_{m-1} \cdots a_0) + (b_{m-1} \cdots b_0)$.

1.) SET carry TO 0
2.) SET i TO 0
3.) REPEAT 4-10 UNTIL $i > m-1$
4.) SET c_i TO $a_i + b_i$ + carry
5.) if $c_i \geq 10$ DO 6 AND 7
6.) SET c_i TO $c_i - 10$
7.) SET carry TO 1
8.) ELSE DO 9
9.) SET carry TO 0
10.) SET i TO $i + 1$
11.) SET c_m = carry
12.) PRINT ANSWER $(c_m\, c_{m-1} \cdots c_1\, c_0)$
13.) STOP

NOTE THAT IN THIS EXAMPLE THERE ARE THREE KINDS OF OPERATIONS:

(1) SEQUENTIAL OPERATIONS
A SINGLE WELL DEFINED TASK. WHEN COMPLETED, MOVE TO NEXT OPERATION

(2) CONDITIONAL OPERATIONS
ASK A QUESTION, AND ON THE BASIS OF THE ANSWER, SELECT THE NEXT OPERATION.

(3) ITERATIVE OPERATIONS
REPEAT SOME BLOCK OF INSTRUCTIONS UNTIL SOME CONDITION IS MET.

EXAMPLE:
TRACE EXECUTION OF THIS ALGORITHM WITH $m=3$, $a_0=7$, $a_1=1$, $a_2=6$, $b_0=5$, $b_1=4$, $b_2=9$.

WHY SPECIFY AN ALGORITHM IN THIS WAY?

- IF WE CAN SPECIFY AN ALGORITHM TO SOLVE A PROBLEM, WE CAN AUTOMATE ITS SOLUTION.

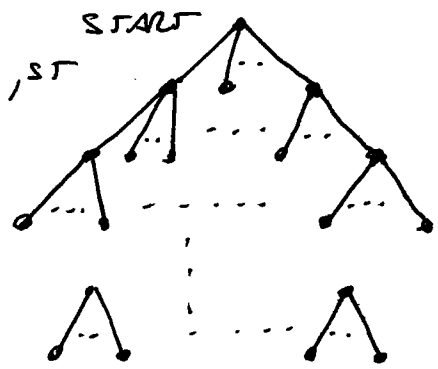
THE ENTITY (MACHINE, ROBOT, PERSON) CARRYING OUT THE STEPS OF AN ALGORITHM IS CALLED A COMPUTING AGENT.

IT MAY SEEM AS IF ANY PROBLEM CAN BE SOLVED BY SOME ALGORITHM. WE'LL SEE IN CHAPTER 10 THAT THERE ARE WELL DEFINED PROBLEMS FOR WHICH NO ALGORITHMIC SOLUTION CAN EXIST (ALAN TURING, KURT GÖDEL)

THERE ARE ALSO PROBLEMS, FOR WHICH WE HAVE ALGORITHMIC SOLUTIONS, BUT WHICH ARE SO INEFFICIENT AS TO BE IMPRACTICAL

EXAMPLE

BRUTE FORCE CHESS ANALYSIS.



40

$40 \cdot 40$

$40 \cdot 40 \cdot 40$

$\vdots$

$40^{30} \approx 10^{48}$

ON AVERAGE $\sim$ 40 LEGAL MOVES FROM ANY POSITION.
$\sim$ 30 MOVES PER GAME.

$\therefore \sim 10^{48}$ BOARD POSITIONS TO ANALYZE.

SUPPOSE WE CAN EVALUATE 1 QUADRILLION $= 10^{15}$ POSITIONS PER SECOND (FAR MORE THAN PRESENTLY POSSIBLE.) THEN

$$\text{TIME} = \frac{10^{48}}{10^{15}} = 10^{33} \text{ SECONDS} \approx 10^{25} \text{ YEARS.}$$

PRESENT AGE OF THE UNIVERSE $\approx$ 10 BILLION $= 10^{10}$ YRS.

THERE ARE ALSO PROBLEMS WHICH MAY HAVE AN ALGORITHMIC SOLUTION, BUT FOR WHICH NONE IS KNOWN. THESE ARE PROBLEMS WHICH INVOLVE SOME KIND OF "INTELLIGENCE".

FORMAL DEFINITION:

AN ALGORITHM IS A WELL ORDERED COLLECTION OF UNAMBIGUOUS AND EFFECTIVELY COMPUTABLE OPERATIONS THAT, WHEN EXECUTED, PRODUCES A RESULT, AND HALTS IN A FINITE AMOUNT OF TIME.

LETS EXAMINE THIS DEFINITION.

WELL ORDERED MEANS THAT IT IS CLEARLY SPECIFIED WHICH OPERATION TO PERFORM FIRST, AND WHEN ANY OPERATION IS COMPLETED, WHICH OPERATION TO PERFORM NEXT.

EX. 1.) DO SOMETHING
2.) " "
3.) " "
4.) REPEAT

REPEAT WHICH STEPS ?

IS M-DIGIT ADDITION WELL ORDERED ?

AN UNAMBIGUOUS OPERATION IS ONE THAT CAN BE CARRIED OUT DIRECTLY BY THE COMPUTING AGENT WITH NO OUTSIDE ASSISTANCE OR FURTHER SIMPLIFICATION.

SUCH AN OPERATION IS ALSO CALLED A PRIMITIVE OPERATION OF THE COMPUTING AGENT

NOTE THAT WHAT IS PRIMITIVE FOR ONE COMPUTING AGENT MAY NOT BE PRIMITIVE FOR ANOTHER.

OUR M-DIGIT ADDITION ASSUMES THAT ADDING TWO (OR THREE) 1-DIGIT NUMBERS IS A PRIMITIVE OPERATION.

IF 1-DIGIT ADDITION IS NOT PRIMITIVE, HOW CAN WE SPECIFY AN ALGORITHM FOR IT?

SUPPOSE OUR COMPUTING AGENT HAS ACCESS TO A TABLE WITH THE ANSWERS TO ALL 1-DIGIT ADDITION PROBLEMS (WITH THREE TERMS), AND SUPPOSE LOOKING UP ANSWERS IN A TABLE IS A PRIMITIVE OPERATION.

GIVEN: 1-DIGIT NUMBERS a, b, c
FIND: THEIR SUM a+b+c
ALGORITHM:
1.) SET ANSWER TO RESULT OF LOOKUP a, b, c
2.) PRINT ANSWER.
3.) STOP.

IF LOOKUP WERE NOT A PRIMITIVE OPERATION WE WOULD HAVE TO SPECIFY MORE DETAIL.

- THE LEVEL OF DETAIL/ABSTRACTION NECESSARY TO SPECIFY AN ALGORITHM DEPENDS ON THE CAPABILITIES OF THE COMPUTING AGENT.

IS ADDING TWO 10-DIGIT NUMBERS A PRIMITIVE OPERATION FOR YOUR POCKET CALCULATOR ?

AN OPERATION IS <u>EFFECTIVELY COMPUTABLE</u> IF IT IS WELL DEFINED. ULTIMATELY THIS MEANS IT IS DOABLE BY SOME COMPUTING AGENT.

<u>EX</u> SET c TO $\frac{a}{b}$

THERE ARE VALUES OF a & b FOR WHICH NO COMPUTING AGENT CAN PERFORM THIS OPERATION.

<u>EX.</u> SET a TO $\sqrt{b}$ (ASSUMING REAL NUMBERS)

<u>EX.</u> list ALL PRIMES $P_1, P_2, P_3 \ldots$.

WE SAY AN ALGORITHM <u>PRODUCES A RESULT</u> RATHER THAN AN ANSWER BECAUSE SOMETIMES (i.e. FOR SOME INPUT) THERE IS NO ANSWER, IN SUCH A CASE THE RESULT WOULD BE AN ERROR MESSAGE.

HALT IN FINITE TIME MEANS ONLY FINITELY MANY OPERATIONS WILL BE EXECUTED

EX. 1.) DO SOMETHING
2.) " "
3.) GO TO 1.

} THIS IS CALLED AN INFINITE LOOP

THIS IS WHY WE ALWAYS INCLUDE THE OPERATION STOP IN OUR ALGORITHMS. IT SHOULD ALWAYS BE CLEAR HOW AND UNDER WHAT CIRCUMSTANCES EXECUTION OF AN ALGORITHM HALTS.

WE CAN SAY THAT COMPUTER SCIENCE IS THE STUDY OF ALGORITHMIC PROBLEM SOLVING.

HW1: DUE IN 1 WEEK { 01 - MON 10/4
02 - TUE 10/5

READ CHAPTERS 1 & 2
DO (P.23) # 3, 4, 5, 7ab, 8, 9