▪ Computable function: f:∑*-> ∑*, some TM on every input w, halts with just f(w) on its tape. ▪ Language A is mapping reducible to language B, $A \leq_m B$ if there is a computable function f:∑*-> ∑* where for every w, w∈A<=>f(w) ∈B▪Function f is called the reduction from A to B –if $A \leq_m B$ and B is decidable, then A is decidable– if $A \leq_m B$ and A is undecidable then B is undecidable – if $A \leq_m B$ and B is Turing-recognizable then A is T-recog – If $A \leq_m B$ and A is not T-recog, then B is not T-recog – EQ_{TM} is neither T-recog nor co-T-recog. ▪ Set A is countable if either it is finite or it has the same size as N ▪ Uncountable: for some infinite sets, no correspondence with N exists, size is too big(real numbers) R is uncount ▪Some languages are not Turing-cognizable. ▪ Characteristic sequence of A: the i^{th} bit of that sequence is a 1 if s_i ∈ A, 0 !∈ A ▪Entry i,j is accept if M_i accepts <M_j>▪Entry I,j is the value of H on input <M_i,<M_j>>▪The diagonalization method: Have sets A and B and function f from A to B+f is one-to-one if it never maps 2 diff elements to the same place, if f(a)≠f(b) whenever a≠b+f is onto if it hits every element of B- for every b ∈B, there is an a∈A such that f(a)=b+A&B are the same size if there is a one-to-one, onto function f:A->B+Correspondence:a function that is both one-to-one and tonto. Paring the elements of A with the elements of B *injective for one-to-one surjective for onto ><bijective for one-to-one and onto ▪A language is Turing-recognizable if & only if +TM+multitaple TM+nondeterminsistic TM+enumerator▪Church-Turing thesis: the connection between the informal notion of algorithm & the precise def. Intuitive notion of algo equals TM algo. ▪The roots of Poly must lie $\pm kc_{max}/c_1$ (k: number of terms in poly, c_{max}: coefficient with the largest absolute value, c_1: coefficient of the highest order term)▪Halting problem ($HALT_{TM}$): the problem of determining whether a TM halts (by accepting or rejecting) on a given input▪M is TM and w is an input string▪Accepting computation history: for M on w is a sequence of configurations $C_1, C_2, \ldots, C_l$ where C_1 is the start conf. of M on w, C_l is an accepting config of M, each C_i legally follows from C_{i-1} according to the rules of M. Rejecting computation history for M on w is defined similarly, except that C_l is a rejecting conf. ▪Linear bounded automation: is a restricted type of TM where in tape head isn't permitted to move off the portion of the tape containing the input. If the machine tries to move its head off either end of the input, head stays where it is in the same way that head will not move off the left hand end of an ordinary TM's tape. ▪Undecidable: $HALT_{TM}$, $REGULAR_{TM}$, E_{LBA}, PCP, E_{TM}, ALL_{CFG}, EQ_{TM}, A_{TM} ▪Decidable: A_{LBA} decider for all A_{DFA}, A_{CFG}, E_{DFA}, E_{CFG} ▪PCP: an undecidable problem concerning simple manipulation of strings. Match: A list of dominos, the string we get by reading off the symbols on the top is the same as the ones on the bottom. ▪Mapping Reducibility: a computable function exists that converts instances of problem A to instances of problem B. A_{TM}={<M,w>| M is a TM and M accepts w} A_{TM} is not undecidable => A_{TM} is not Turing-recognizable▪A language is decidable if it is Turing-recognizable and co-Turing-recognizable ▪Reduction is a way of converting one problem to another problem in such a way that a solution to the second problem can be used to solve the 1st problem. Reducibility: the primary method for proving that problems are computationally unsolvable.