EXERCISE 1.24

Theory:

- CHURCH'S THESIS: TURING MACHINES ARE ENOUGH POWERFUL TO DESCRIBE ANY KIND OF ALGORITHM

1. TURING MACHINES
   - 1.1 COMPONENTS (BASIC)
     - 1.1.1 FINITE CONTROL PART
     - 1.1.2 TAPE USED TO STORE DATA
   - 1.2 VARIATIONS W.R.T. BASIC
     - 1.2.1 MULTIPLE TAPES
     - 1.2.2 1-WAY TAPE
     - 1.2.3 2-WAY TAPE
   - 1.3 ALL THOSE VARIATIONS ARE NOT REALLY RELEVANT IN ORDER TO DISTINGUISH COMPUTABILITY
     - 1.3.1 DETERMINISTIC → EACH STEPS MOVE TO A UNIQUE SUCCESSOR CONFIGURATION
     - 1.3.2 NON DETERMINISTIC → ONE OF THE POSSIBLE SUCCESSOR CONFIGURATIONS.
   - 1.4 EFFICENCY
     - 1.4.1 ~~HALTS WITHIN $P(|x|)$ STEPS WRT INPUT x - POLYNOMIAL~~ TURING MACHINE
     - 1.4.2

2. EFFICENCY
   - 2.1 POLYNOMIAL TIME TURING MACHINE, IF IT HALTS IN $P(|x|)$
     - 2.1.1 $P(\ )$ IS A POLYNOMIAL
     - 2.1.2 x IS THE INPUT STRING
     - 2.1.3 $|x|$ DENOTES THE LENGTH OF THE INPUT STRING
   - 2.2 COMPLEXITY CLASS P:
     - 2.2.2 SET OF ALL PROBLEMS WHICH CAN BE SOLVED BY A DETERMINISTIC POLYNOMIAL
       - a) P IS USEFUL TO DEFINE IS FEASIBLE
       - b) PROBLEM $A \in P \Rightarrow$ IS FEASIBLE

3 RANDOM TURING MACHINES
   - 3.1 "WORST-CASE" COMPLEXITY TO CHARACTERIZE COMPLEXITY W/ STANDARD TM
   - 3.2 IN CRYPTO IS MUCH MORE USEFUL "AVERAGE CASE"
   - 3.3 PROBABILISTIC TURING MACHINES
     - 3.3.1 SIMILAR TO NON-DETERMINISTIC TURING MACHINES
     - 3.3.2 ~~∀STEP~~ WE CHOSE NEXT UNIFORMLY AT RANDOM
     - 3.3.3 RANDOM TAPE: TURING TAPE W/ RANDOM BITS
   - 3.4 PROBLEM IS FEASIBLE IF PPt (PROBABILISTIC POLYNOMIAL TIME)

4 CRYPTOGRAPHY AND PROBABILISTIC TM

4.1 SCHEMES ADOPT PPt ALGORITHMS

4.1.1 PRIMALITY TESTING (PROBABILISTIC)

4.1.2 PRIME NUMBER GENERATION (PROBABILISTIC)

4.2 ADVERSARIES ARE P.P.t

4.2.1 NOT ENOUGH (SECURE) TO STATE that CRSchp is NOT in P

4.2.2 We Need SRSchp ∈ BPP (BOUNDED-ERROR PPt)

4.3 EASY PROBLEM – SOLVED BY A PPt IN WITH High PROBABILITY

4.4 HARD PROBLEM – SOLVED BY A PPt with Negligible probability

4.5 P=NP DO NOT SAY US ANYTHING ABOUT AVERAGE CASE)

II ASSUMPTIONS

1 HARDNESS ASSUMPTION: (ODH ⇒ DH ⇏ DL)

1.1 DISCRETE LOGARITHM (DL – STANDARD CONDITION)

1.1.1 FOR GROUP $\langle g \rangle$ IS HARD TO COMPUTE x given A Random group element g^x

1.2 DIFFIE-HELLMAN (DH)

1.2.1 FOR GROUP $\langle g \rangle$ IS HARD TO COMPUTE g^{xy} given Random Group Elements g^x, g^y

1.3 DECISIONAL DIFFIE-HELLMAN (DDH –

1.3.1 DDL GROUP $\langle g \rangle$ IS HARD TO Distinguish g^{xy} from Rge g^z given g, g^x, g^y (now we have defined this) II 2.5/2.6

2 INDISTINGUISHABILITY ASSUMPTIONS

2.1 NEGLIGIBLE FUNCTIONS

2.1.1 NEGLIGIBLE: $\frac{1}{2^k}$; $\frac{1}{k^{\log k}}$

2.1.2 NON-NEGLIGIBLE: $1/k$; $\frac{1}{k^{100}}$

2.1.3 $1/k$ IS THE SMALLEST NON-NEGLIGIBLE FUNCTION AND $\lim_{k \to \infty} f(k) = 0$

2.2 PERFECTLY INDISTINGUISHABLE: IF $\Delta(X_1, X_1) = 0$, IDENTICALLY DISTRIBUTED

2.3 Statistically indistinguishable. $\Delta(X_1, X_1)$ IS NEGLIGIBLE AS FUNCTION OF k

2.4 Computationally indistinguishable: $\Delta(X_1, X_1)$ IS NEG AS A FUNCTION of k FOR EVERY PPt. ALGORITHM D W/ OUTPUT $\{0,1\}$

2.4.1 D is a boolean Distinguisher

2.5* BOOLEAN DISTINGUISHER = AN PPt. ALGORITHM IS ABLE TO TELL YOU IF TWO DISTRIBUTION ARE THE SAME OR NOT

2.6* ADVANTAGE HOW GOOD ARE the PERFORMANCE of A DISTINGUISHER

2.6.1 $ADV_D(X_1, Y_1) = |Pr[D(x_1) = 1] - Pr[D(Y_1) = 1]| = |Pr[D(x_1) = 0] - Pr[D(Y_1) = 0]|$