1. (10 points) Given the positive integers $x, y, m$, compute the quantity $z = (x^y \mod m)$ in polynomial time. Here $0 \leq z \leq m - 1$. Your algorithm should be direct, no recursive calls to itself. Give your solution in pseudocode. Use as few arithmetic operations as possible. Name the method used. Assuming each of $x, y, m$ have $n$ digits, estimate the number of multiplications/divisions of $O(n)$-digit integers required by your algorithm.

2. (2+6+7 points) (a) Define the CLIQUE language. (This language corresponds to the decision version of the “maximum clique” problem.) (b) Give a Karp-reduction from CLIQUE to HALTING. (c) Prove that there is no Karp-reduction from HALTING to CLIQUE.
3. (5+10 points) When asked to give a formal definition of NP, Chuck gave this answer: “A language \( L \subseteq \Sigma^* \) belongs to NP if and only if there exists a finite alphabet \( \Sigma_1 \) and a language \( L_1 \subseteq \Sigma_1^* \) such that \( L_1 \in P \) and \( (\exists c)(\forall x \in \Sigma^*)(x \in L \Rightarrow (\exists y \in \Sigma_1^*)(|y| \leq |x|^c \text{ AND } (x, y) \in L_1)) \).

(a) Find the error in this definition; make the small change needed to correct it. (There is only one small error.)
(b) Determine, exactly which languages \( L \) satisfy Chuck’s definition. Prove your answer.

4. (10 points; lose 4 points for each mistake) Consider the following three statements: (A) 3-colorability of graphs can be decided in polynomial time. (B) RSA can be broken in polynomial time. (C) Integers can be factored into their prime factors in polynomial time. – Which of the six implications is known (circle all that apply): (A) \( \Rightarrow \) (B); (B) \( \Rightarrow \) (A); (A) \( \Rightarrow \) (C); (C) \( \Rightarrow \) (A); (B) \( \Rightarrow \) (C); (C) \( \Rightarrow \) (B). Do not prove.

5. (10 points) (MAX-3-SAT) Let \( C_1, \ldots, C_m \) be disjunctive 3-clauses (expressions of the form \( z_1 \lor z_2 \lor z_3 \) where each \( z_i \) is a literal) over \( n \) Boolean variables \( x_1, \ldots, x_n \). Prove that at least \( 7m/8 \) of the clauses are simultaneously satisfiable. Define your random variables!

6. (BONUS, 8B points) The “weighted interval scheduling” problem takes as input a list of \( n \) intervals \( (s(i), t(i)) \) and corresponding weights \( w_i > 0 \) and asks to find a set of disjoint intervals among these of maximum total weight. Solve this problem in \( O(n) \) plus sorting whatever needs to be sorted. Hint: dynamic programming. Half the credit goes for a clear definition of the array of problems to be solved (the “brain” of the algorithm).

7. (BONUS, 6B points) Let \( K \) be the set of those 3-colorable graphs which have fewer edges than vertices. Assuming 3-COL is NP-complete, prove that \( K \) is NP-complete,