(1) Consider the algorithm that takes as input, \( f \), a description of a Boolean function, \( f = f(x_1, \ldots, x_n) \), ignores this input, and as output writes down a description of the graph, \( G \), with two vertices and one edge; note that \( G \) is 2-colourable. If \( f \in \text{SAT} \), then the resulting output, \( \langle G \rangle \), lies in \( \text{2COLOR} \). Does this show that \( \text{SAT} \leq_p \text{2COLOR} \)?

(2) (a) Give a reduction from \( 3\text{SAT} \) to \( 3\text{COLOR} \), using the subgraphs in the hint for Problem 7.29 in [Sip].
(b) Show that \( 3\text{COLOR} \) is \( \text{NP} \)-complete.

[Hint: To show that a language, \( L \), is \( \text{NP} \)-complete, you must show that (1) \( L \) lies in \( \text{NP} \), and (2) any language in \( \text{NP} \) can be reduced in polynomial time to \( L \). In class we have already described why \( 3\text{COLOR} \) is in \( \text{NP} \), but you should state this and justify this in a sentence or two. You should also, in a sentence or two, explain why \( 3\text{SAT} \leq_p 3\text{COLOR} \) implies that any language in \( \text{NP} \) can be reduced to \( 3\text{COLOR} \).]
(3) Exercise 7.22 in [Sip] (show that DOUBLE-SAT, defined there, is NP-complete); do this by reducing SAT (or 3SAT) to DOUBLE-SAT.

[Hint: The reduction is easy, much easier than to SUBSET-SUM, 3COLOR, or VERTEX-EXPANSION. Don’t forget to: (1) show that DOUBLE-SAT lies in NP, and (2) explain why any language in NP can be reduced to DOUBLE-SAT once you have reduced SAT (or 3SAT) to DOUBLE-SAT.]

(4) In class we described a way to reduce 3SAT to VERTEX-EXPANSION, where the latter is given by

\[ \{ \langle G, a, b \rangle \mid G = (V, E) \text{ is a graph such for some } A \subset V, |A| = a \text{ and } |\Gamma(A)| \geq b \} \]

We did this by taking a 3CNF Boolean formula, \( f = f(x_1, \ldots, x_n) \), with \( n \) variables and \( m \) clauses, and building a graph \( G = (V, E) \) by

(a) introducing, for each Boolean variable, \( x_i \),
   (i) two “true/false” vertices, representing setting the variable \( x_i \) to either true or false, and
   (ii) \( B \) “dummy vertices” connected to the two true/false vertices for some integer \( B \) that we will specify below (we think of \( B \) as being “very large,” but hopefully no larger than some polynomial in \( n + m \)), and

(b) one additional “clause enforcing” vertex for each clause, connected to the three vertices representing the true/false variable settings in the clause.

(see notes for 11-13).

Based on this construction taking \( f \) to a graph \( G = (V, E) \), answer the following questions:

(a) What is \( |V| \) as a function of \( n, m, B \)?

(b) Show that if \( A \subset V \) is any subset that for some \( i = 1, \ldots, n \) does not contain either “true/false” vertex associated to \( x_i \), then

\[ |\Gamma(A)| \leq |V| - B \]

(whether or not \( f \) is satisfiable).

(c) Show that if \( A \subset V \) is any subset that for each \( i = 1, \ldots, n \) contains at least “true/false” vertex associated to \( x_i \), then

\[ |\Gamma(A)| \geq nB \]

(whether or not \( f \) is satisfiable).

(d) Use the above three parts to show that if \( B > 2n + m \), then the largest possible value of \( |\Gamma(A)| \) subject to \( A \subset V \) with \( |A| = n \) is attained for some \( A \) that has exactly one true/false vertex associated to each variable \( x_1, \ldots, x_n \) (whether or not \( f \) is satisfiable).

(e) Conclude that for \( B = 2n + m + 1 \), \( f \) is satisfiable iff for some \( A \subset V \) with \( |A| = n \) we have

\[ |\Gamma(A)| = nB + m. \]
(f) Use this last statement to explain how this gives a polynomial time reduction from 3SAT to VERTEX-EXPANSION; make sure to include a sentence or two explaining why this reduction is polynomial time.