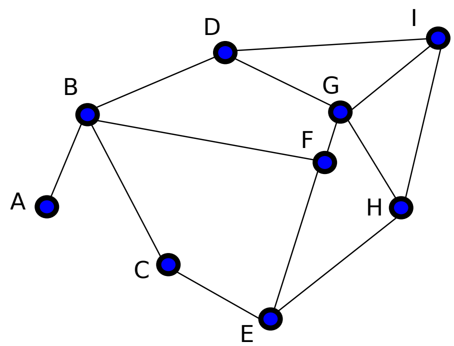
Maximum Matchings for Vertex Cover. In this exercise, we propose to analyze an algorithm for Vertex Cover that also returns a solution which is at most twice the optimal solution in the worst-case. We say that a set of edges M is a matching if no two edges of M share a common vertex, namely for each edges $(u, v), (w, x) \in M$, $u \neq w, x$ and $v \neq w, x$. Moreover, a matching M is said to be maximal if no edge of the graph can be added to M and M remains a matching. We consider the following algorithm:

1. $VC \leftarrow \emptyset$
2. Compute a maximal matching M
3. For each edge (v_i, v_j) in M, add v_i, v_j to VC
4. Return VC

Question 1. Consider the following graph. Is the set $(A, B), (C, E), (D, I), (G, H)$ a matching? Is it maximal?
Question 2. Consider the following graph. Is the set A, B, C, E, D, I, G, H a vertex cover? Is it optimal?



Correctness. Consider the vertices of the solution VC returned by the algorithm. We want to show that they form a correct vertex cover. Recall that a set of vertices VC is a vertex cover if and only if for all edge (u, v) of the graph, either u is in VC or v is in VC or u and v are in VC.
a) Suppose VC is not a correct vertex cover, what does the above sentence imply?
b) If one edge is not covered by the vertices of VC, what does that imply for M? Recall that M is a *maximal* matching.
c) Conclude about the correctness of the algorithm.
Approximation Guarantee. We want to show that the size of any maximal matching is a lower bound for the size of the minimum vertex cover. Suppose M is a maximal matching and S an optimal vertex cover.
a) Consider two edges $(v_i, v_j), (v_k, v_l)$ of M, what is the size of the intersection of $\{v_i, v_j\}$ with $\{v_k, v_l\}$? Namely the size of $\{v_i, v_j\} \cap \{v_k, v_l\}$? Recall that M is a matching.

b) Use the previous question to relate the number of edges in M to the number of vertices in M. Namely, what is the relation between $\sum_{(u,v)\in M} |\{u,v\}|$ and M?
c) Consider an edge (v_i, v_j) of M, what is the minimum size of the intersection of $\{v_i, v_j\}$ with S? Namely, the minimum size of $\{v_i, v_j\} \cap S$? Recall that S is a Vertex Cover solution.
d) What is the size of the solution VC returned by the algorithm compared to M? Compared to an optimal Vertex Cover S in the worst-case (use questions b,c)?
Running time. It is possible to compute a maximal matching in time $O(n+m)$ where n is the number of vertices of the graph and m the number of edges. What is the overall complexity of the algorithm?
Tightness Consider the path with 4 vertices, namely the graph which consists in vertices v_0, v_1, v_2, v_3 and edges $(v_0, v_1), (v_1, v_2), (v_2, v_3)$.
a) What are the 2 possible maximal matchings?
b) What is the maximal matching that will lead the algorithm to return an optimal Vertex Cover?
c) What is the maximal matching that will lead the algorithm to return a factor 2 approximation?

Triangles of a Graph. We consider the following problem: given a graph $G = (V, E)$, we say that a triple of vertices A, B, C forms a *triangle* if the edges $(A, B), (A, C), (B, C)$ are in E. The goal of the exercise is to define an algorithm that, given a graph $G = (V, E)$, returns a set of vertices S such that $G \setminus S$ does not contain any triangle. Moreover we want that the algorithm returns a set as small as possible. Throughout the exercise, we will consider an optimal solution S for the problem.
Question 1. Suppose we are given a variable x_A for each vertex $A \in V$ such that $x_A = 1$ if and only if $A \in S$. Consider a triangle A, B, C, give a tight lower bound for $x_A + x_B + x_C$.
Question 2. Describe an integer program for the problem based on the above observation and on the linear program for Vertex Cover seen during the lectures.
Question 3. We now consider a solution to the linear relaxation of the integer program of the previous question. For any triangle A, B, C, what is the minimum value of $\max(x_A, x_B, x_C)$? Give a tight lower bound for $\max(x_A, x_B, x_C)$.
Rounding. Describe a rounding procedure based on the rounding procedure for vertex cover.
Correctness. Consider the rounding procedure of the previous question and the output S' of the procedure. Suppose that S' is not a solution.
a) Give a lower bound for the number of triangles in $G \setminus S'$.
b) Consider a triangle A, B, C of $G \setminus S'$ and the associated constraint in the linear program, what is the value for $\max(x_A, x_B, x_C)$?
c) Conclude.
Approximation. We now turn to the analysis of the approximation guarantee.

a) Using the above discussion, by which factor can the value of the fractional solution be multiplied in the worst-case?
b) Conclude about the approximation guarantee of the algorithm.
Runtime. Assuming there is an algorithm solving the linear program in time $O(T)$, what is the overall complexity of your algorithm?
Tightness Give an example with 3 vertices that shows that the analysis is tight.