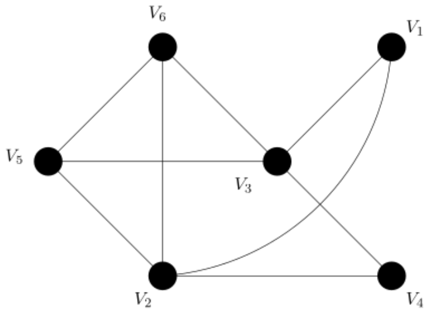
1 Artificial Intelligence (sbh11)

Consider the following CSP. The nodes of the graph represent variables V_i where $i = 1, \dots, 6$, each with domain $\{\downarrow, \uparrow, \oplus, \ominus\}$. Edges (V, V') are binary constraints, each requiring that V can take values in $\{\downarrow, \uparrow\}$ if and only if V' has values in $\{\oplus, \ominus\}$, and V can take values in $\{\oplus, \ominus\}$ if and only if V' has values in $\{\downarrow, \uparrow\}$. Let S be the sequence of assignments $(V_1 = \oplus, V_2 =\uparrow, V_3 =\downarrow, V_4 = \ominus, V_5 = \oplus)$.



(*a*) (*i*) State the components of a *constraint satisfaction problem (CSP)* and describe the requirements for a solution to such a problem. [4 marks]

(*ii*) After performing the assignment S, can a valid assignment be made for V_6? Explain your answer. [1 mark]

(*iii*) Briefly explain how *forward checking* operates when solving a CSP. [2 marks]

(*iv*) Explain in detail how forward checking operates when applying the sequence S. At which assignment in S does the procedure indicate we can not continue? [4 marks]

(*v*) Would it have been possible in Part (*a*)(*iv*) to establish sooner that S could not lead to a solution, by performing a more detailed examination of the constraints? Explain your answer. [3 marks]

(*b*) Now assume that no forward checking or other form of constraint propagation is used.

(*i*) Assume that we are using *graph-based backjumping*, and have applied the sequence S of assignments. We are now attempting to assign V_6. Explain how this leads to the first backjump, and how we establish which variable the first backjump decides to re-assign. [2 marks]

(*ii*) After performing the first backjump we again exhaust the possible assignments for the variable. Explain how graph-based backjumping establishes the second backjump in this example. [4 marks]