Cake Protection Agency

Mouseland has multiple elite police forces with different missions, one of the (arguably) most important ones is the CPA, the cake protection agency. They are tasked with protecting all the important cakes in the land from vicious attacks.

The CPA just got the word of a very dangerous cake terrorist, mouse Gehr, plotting an attack. Apparently he plans to leave his home, walk to the cake and throw it to the ground. The only way for the CPA agents to stop him is to either arrive at the cake earlier than terrorist Gehr (and move the cake to a more secure location), or to intercept and detain him at an intersection before reaching the cake.

Your task will be to help the noble agents of the CPA trying to stop cake terrorist Gehr.

Formal Description Assume mouseland is given as a *weighted*, directed graph $G = (V, E)$ with $n + 3$ vertices $V = \{v_{\text{cpa}}, v_{\text{gehr}}, v_{\text{cake}}, v_1, \dots v_n\}$, the starting base of the CPA agents, the initial location of mouse Gehr, the location of the cake and the remaining intersections in mouseland. From each vertex you can reach the cake (it's not a lie). An edge $e = (u, v, t) \in E$ connects $u \in V$ to $v \in V$ with integer travel time $t \geq 0$. Note that all roads in mouseland are one way, so the edge (u, v, t) can only be travelled from u to v and not v to u (there might be an additional edge (v, u, t') in the graph to travel in the opposite direction). The CPA agents can stop mouse Gehr if one arrives at the cake strictly earlier than mouse Gehr or if they occupy the same node $v \neq v_{\text{cake}}$ at the same time.

Subtask 1: Latest Starting Time (15 points)

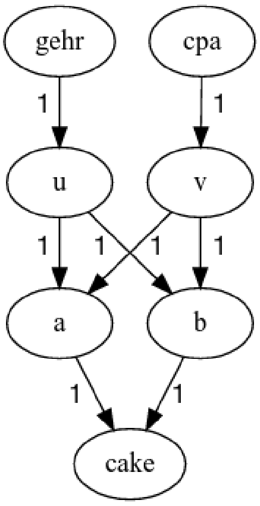
Mouse Gehr was not careful enough to protect his secret plan and his exact route $p = (v_{\text{gehr}}, \dots, v_{\text{cake}})$ to the cake as well as his (integer) start time T got leaked. Design an algorithm which finds the latest integer time T' at which a CPA agent has to leave the CPA base v_{cpa} to stop mouse Gehr by either arriving earlier at the cake or intercepting him along his route. Argue why your algorithm is correct, write down pseudocode, analyze its asymptotic runtime and its memory usage.

Subtask 2: Plenty of Agents (35 points)

Mouse Gehr was more careful and his route didn't leak. Assume he leaves his home at time T and the CPA can send out arbitrarily many agents. Find the latest integer time T', when those agents have to be dispatched in order to stop mouse Gehr. Argue why your algorithm is correct and analyse its runtime complexity.

Also note that mouse Gehr knows where all deployed CPA agents are at any time and vice versa. Moreever, whenever a CPA agent and Mouse Gehr have to make a decision (which outgoing edge to follow) at the exact same time, then mouse Gehr decides first and the CPA agent *will know* which one he chose.

This is illustrated in the following graph. Assume all edge weights are 1, in that case *one* CPA agent leaving at the same time as mouse Gehr will always catch him, because the optimal strategy for both is going to be u (for mouse Gehr) and v (for the CPA agent) in the first step. After that, mouse Gehr has to decide first between a and b, and the CPA agent can just mirror his choice to meet and detain him at the respective node.



Subtask 3: One Agent (50 points)

Assume the same setup as in subtask B, however, the CPA only has one agent available. Again determine the latest integer time T' when he needs to leave the CPA base v_{cpa} in order to stop mouse Gehr. Argue why your algorithm is correct, write down pseudocode and analyse its runtime complexity and its memory usage.