

Zhrnutie práce: Letný semester

V letnom semestri som sa snažil priblížiť implementáciu algoritmu k algoritmu zo študijnej práce, urobiť ho čitateľnejším a štruktúrovanejším.

Hlavné vylepšenia a zmeny:

- **Optimalizácia pre veľké grafy:** Zatiaľ čo predchádzajúca verzia algoritmu zlyhávala a cyklila sa pri spracovaní väčších grafov, ale teraz algoritmus funguje bezchybne a rýchlo aj na rozsiahlych a komplexných grafoch.
- **Univerzálny vstup a spracovanie:** Zmenil som vstup funkcie tak, aby mohla spracovať ľubovoľný kubický graf vo forme vektora susedov vrcholov, ktoré program následne transformuje do správnej vnútornej grafovej štruktúry. Graf teraz dokáže akceptovať a správne spracovať akýkoľvek typ kubického grafu, bez ohľadu na to, či obsahuje slučky, násobné (dvojité) hrany alebo viacero komponentov. Výstupom algoritmu je pravdivostná hodnota (`true/false`) indikujúca existenciu cyklického rezu hranou, pričom samotný rez sa vracia ako vektor párov vrcholov (kde každý pár reprezentuje hranu určenú na prerezanie).
- **Štruktúra a čitateľnosť:** Funkcie som prepísal takmer identickým spôsobom, ako boli formulované v pôvodnom článku, čo zlepšilo čitateľnosť a nadväznosť kódu na teóriu.

Návrh toku (Ford-Fulkersonov model) a dôkaz správnosti Pre riešenie problematiky toku som sa samostatne rozhodol adaptovať myšlienku Ford-Fulkersonovho (FF) modelu. Tento krok nielenže uľahčil pochopenie logiky toku v sieti, ale aj viditeľne zlepšil čas behu (runtime) algoritmu.

(Nie úplný) dôkaz správnosti algoritmu bude v súbore README.md v repozitári GitHub.

Algoritmus hľadania cyklického rezu Logika a algoritmus pre samotné hľadanie rezu boli pôvodne navrhnuté v konzultácií s Gemini AI, a som tento návrh následne detailne upravil, implementoval do neho správny backtracking a optimalizoval ho.

(Nie úplný) dôkaz správnosti algoritmu bude v súbore README.md v repozitári GitHub.