

Zadanie 4

Úloha 1

Pre tuto ulohu som implementoval riedy:

- Room ktora popisuje miestnosti

- Vertex ktora popisuje dvere(hrany grafu)

- Flow ktora obsahuje cely graf

trieda Flow obsahuje metodu `edmons_karp` ktora vypocita touto metodom vsetky parametre pre optimalny tok nasledne to v subore `evacuation.py` vypisujem

Úloha 2

V druhej ulohy som pouzil algoritmus ktory vyuziva dekompoziciu grafu. Vďaka triedam `Person` a `Graph` som si jednoducho odstraňoval a pridával vrcholy grafu. Zoradil som ho od najväčšieho stupňa uzla a uberal postupne nody podľa toho či nemali s odobratými nejaké prepojenie. ak neexistovalo prepojenie s odobratými uzlami dal som ich do rovnakej skupiny

Následne ak mi vznikla skupina o veľkosti 1 zobral som náhodne zo skupiny ktorej bola väčšia ako 2 členov a skúsil pridať do takto vzniknutých 1členných členov ak nemali prepojenie s už existujúcim. následne som pridaného člena aj odstránil z pôvodnej skupiny

Úloha 3

Tretiu ulohu som naprogramoval "brute force" metódou kde si najdem všetky možné kombinácie mapovania ohodnotím ich podľa počtu hran od budovy k parkovisku a následne najdem minimum v strukture. Táto metóda je veľmi pomalá a zaberie veľmi veľa pamäte !!!