

#: 1A

Vertici: Ordini di uno Store

Archi (condizione): Distanza temporale $\leq K$ giorni

Peso arco: Somma quantity dei due ordini

Input utente: Store, K

Possibili domande:

- Crea grafo

- Cammino più lungo

- Chi ha grado uscente max

#: 2A

Tema: Ordini di uno store (continuazione di 1A)

Vincoli ricorsivi: Peso archi decrescente • ordini non ripetuti

Obiettivo da massimizzare: Somma pesi archi

#: 1B

Vertici: Products disponibili in uno store

Archi (condizione): Sono stati venduti nello stesso ordine

Peso arco: # ordini in comune

Input utente: Store, soglia min vendite

Possibili domande:

- Trova community (componenti connesse)

- Prodotto più "centrale" (betweenness)

#: 2B

Tema: Catena di prodotti

Vincoli ricorsivi: Ogni nuovo prodotto deve appartenere a categoria diversa

- peso = list_price

Obiettivo da massimizzare: Prezzo totale della catena

#: 1C

Vertici: Clienti di uno store

Archi (condizione): Hanno acquistato lo stesso prodotto

Peso arco: # prodotti condivisi

Input utente: Store, prodotto facoltativo

Possibili domande:

- Cluster di clienti simili

- Cliente col maggior clustering-coefficient

#: 2C

Tema: Percorso tra store

Vincoli ricorsivi: Devono essere collegati da archi con #clienti crescente

Obiettivo da massimizzare: Lunghezza del percorso (n store)

#: 1D

Vertici: Stores

Archi (condizione): Condividono almeno M clienti

Peso arco: # clienti in comune

Input utente: M

Possibili domande:

- Trova store “ponte” (highest eigenvector centrality)

#: 2D

Tema: Selezione di clienti

Vincoli ricorsivi:

Nessun cliente condivide più di X prodotti con un altro già scelto

Obiettivo da massimizzare: Fatturato totale generato

#: 1E

Vertici: Staff

Archi (condizione): Hanno gestito ordini dello stesso cliente

Peso arco: # clienti condivisi

Input utente: Store opzionale

Possibili domande:

- Staff più “collaborativo” (grado)
- Cammino più lungo tra due membri

#: 2E

Tema: Turno ideale di staff

Vincoli ricorsivi: Staff non può appartenere allo stesso store già in turno

- peso = ordini gestiti

Obiettivo da massimizzare: Numero di ordini coperti

#: 1F

Vertici: Ordini

Archi (condizione): Sequenza cronologica, ma solo se quantità totale $\geq Q$

Peso arco: Giorni di distanza

Input utente: Q

Possibili domande:

- Path più breve in termini di tempo
- Individua eventuali cicli (dovrebbe essere DAG)

#: 2F

Tema: Catena di ordini con sconto

Vincoli ricorsivi: Ogni arco deve avere discount maggiore del precedente

- max 20 ordini

Obiettivo da massimizzare: Sconto medio del percorso