Fakulta riadenia a informatiky

Informatika

SEMESTRÁLNA PRÁCA I. – ALGORITMY A ÚDAJOVÉ ŠTRUKTÚRY I.

# AoE

Ing. Michal Varga, PhD. – PIA 12:00

2018/2019 Dominik Bullo 5ZY029

## Návrh aplikácie

Aplikáciu som si rozvrhol do nasledujúcich tried : Biofarmar, Vozidlo, Zakaznik, Objednavka, Firma, Datum, Konstanty, Uicko a ešte samostatná trieda pre main.

Trieda Biofarmar zastrešuje jednotlivých biofarmárov a prácu s nimi, obsahuje informácie čo farmár vyrába, množstvo surovín, ktoré má a udržuje si info o nákupe spoločnosti od daného biofarmára za posledných 30 dni.

Trieda Vozidlo podobne ako Biofarmar udržuje si informácie o jednotlivých vozidlách a nastavuje unikátku ŠPZ každému autu.

Trieda Zakaznik uchováva len meno a adresu zákazníka plus výpis do konzoly.

Trieda Objednavka pamätá Zákazníka, ktorý danú objednávku vytvoril a pracuje sa tu hlavne s výpismi a zmenou stavu objednávky.

Trieda Firma zastrešuje jednotlivé funkcie a spájanie ostatných tried, obsahuje niekoľko podtried (použitých na obalenie niektorých objektov) uchováva zoznamy na jednotlivé objekty, rieši výrobu produktov, doručovanie a nákupy od Biofarmárov taktiež zahrňuje jednotlivé výpisy podľa kritérií stanovených v jednotlivých bodoch.

Trieda Datum rieši prácu s dátumom, poskytuje nám len jednu inštanciu, obsahuje statické metódy na prekonvertovanie dátumu na int a naopak a taktiež posúvanie dní.

Trieda Konstanty obsahuje ako vyplýva z názvu konštanty a enumy a taktiež ju používam pri prepočítaní surovín na výrobu produktov (Recept)

Trieda Uicko má „nekonečnú slučku“ a rieši jednotlivú obsluhu triedy firma.

Trieda main spúšťa UI.

## Popis implemetácie údajových štruktúr

V Biofarmárovi používam LinkedList na ukladanie 30dňovej histórie nákupov spoločnosti ktorý následne používam na prepočítanie priemeru za jednotlivé tovary. LinkedList pretože je jednoduché mazať prvý prvok a pridávať posledný každý deň a teda nemusím posúvať celé pole.

V triede Firma ukladám Biofarmárov do LinkListu nepristupujem k nim cez [ ] a pri ukladaní insertujem pri čom je lepšie použiť LinkList ak sa prvky priraďujú blízko začiatku listu. Vozidla ukladám do ArrayListu, nepredpokladám že sa bude pole často meniť a teda ušetrím miesto keď nemusím ukladať pointre ako v LinkListe. Oboje objednávky aj vozidlá ukladám v LinkListe nepotrebujem prístup k jednotlivým prvkom cez index len prechádzam listy cez foreach plus predpokladám že informácie o Objednávkach a Zákazníkoch sa vo firme môžu často meniť resp. odstaňovať prvky zo zoznamu a podobne. Uchovávam si informáciu o nakúpenom tovare od Biofarmarov tieto Nákupy ukladám do fronty implementovanej explicitne nikdy prvkov nie je veľa priebežne (každý deň sa tento list vyprázdňuje). Taktiež používam prioritný front implementovaný ako halda v niektorých metódach firmy kde potrebujem prvky zoradiť podľa nejakého údaju a následne s nimi pracovať napr. pri výpise alebo nákupe tovaru. Pri prídavaní nákladov jednotlivým vozidlám si triedim zoznam vozidiel do dvoch lokálnych ArrayListov podľa typu vozidla, v tomto prípade pristupujem k prvkom cez index preto ArrayList.

## Zložitosti jednotlivých operácii

1. Pridanie Biofarmára do linkListu, rovno ich triedim čiže hľadám správnu pozíciu => N operácií a volám insert, čo je ďalších N operácii
2. Vypísanie zoznamu Biofarmárov foreach => O(N)
3. Vloženie vozidla do LinkListu O(1).
4. Vypísanie vozidiel podľa dátumu evidencie, pri vložení sa vkladajú s aktuálnym dátumom čiže už sú zoradené. Zložitosť O(N).
5. Pridanie nového zákazníka. Utriedenie podľa abecedy ako biofarmárov, zložitosť O(N).
6. Kontrola či existuje Zákazník ktorý si objednal tovar N operácií, pridanie objednávky ak spĺňa požadované vlastnosti 1 operácia.
7. Prebehne všetky objednávky a spočíta počet tovarov O(N), prepočíta koľko potrebujeme polotovarov 1 operácia pre každý polotovar, dokúpenie surovín => napĺňam haldu všetkými biofarmarmi čiźe pridanie do haldy zložitosť O(logN) a prejdenie foreachom cez zoznam Biofarmarov O(N) , prebehnutie cez haldu kým nenakúpim alebo nedôjdu Biofarmári N- krát pop, ktorý má zložitosť logN, pridanie nakúpeného tovaru do zoznamu nakúpenyTovar O(1). Naskladnenie má zložitosť O(N).
8. Výpis objednávok na zajtra zložitosť O(N).
9. a  10. vyrobenie produktov => odčítanie surovín zo skladov jednotková zložitosť, naplnenie prioritného frontu implemetovaného haldou pripravenými objednávkami O(N), vytvorenie podzoznamov pre vozidlá, podľa typu vozidla O(N), Rozloženie objednávok podľa regiónu do áut zmena nákladov áut O(N) v halde mám objednávky už zoradené podľa regiónu.
10. Pri každom zákazníkovi pozerám zoznam objednávok teda O(N^2).
11. Vloženie cez cyklus N operácií do Haldy O(logN) potom opäť N operácií pri výpise.
12. Objednávky sú v zozname podľa dátumu evidencie takže sa len vypíšu podľa podmienky ktoré potrebujem O(N).
13. Vkladanie do haldy O(logN) a cez všetkých biofarmárov foreach O(N) a v každom sa spočítava počet dní kedy kúpil (priorita) O(N). Výpis O(1).
14. Neimplementované

## Diagram Tried



## Používateľská príručka

Program obsahuje jednoduché konzolové užívateľské rozhranie. Jednotlivé pokyny sú číslované stačí sa riadiť upresňujúcim textom ktorý vyzýva užívateľa k akcií.