

Standardní zadání semestrální práce pro PT 2020/2021

Zadání je určeno pro **dva** studenty. Práce zahrnuje dvě dílčí části – vytvoření funkčního programu diskrétní simulace a napsání strukturované dokumentace.

Zadání:

Společnost „Dřevák a syn“ vlastní po celém světě síť D dřevozpracujících továren. Snaha společnosti je minimalizovat cenu přepravy výrobků z továren do supermarketů. Starý Dřevák nikdy nešel příliš s dobou, co se kam má přepravit (samozřejmě aby to vyšlo nejlevněji), vždy počítal na papír. Nyní ale převzal společnost mladý Luboš Dřevák, který si na to chce napsat software a místo celovečerního počítání raději sledovat TokTik nebo hrát GallFuys přes Diskord s přáteli.

Všechny dřevozpracující továrny dokáží produkovat Z druhů různého zboží, od skříní přes dekorace až po záchodová prkénka. Své výrobky rozváží do S supermarketů, nejčastěji do supermarketů BOBI a PATu, ale ta prkénka chtějí i v CUUPu. Vedení každého supermarketu s je však náročné, zvláště pak v BOBI na Borech (který mají interně zařazený jako pátý v pořadí, tedy $s = 5$, kde to vypadá, jako by management generoval poptávku náhodně) a každý den t hodlá daný supermarket prodat jiný počet zboží z (hodnotu poptávaného počtu zboží značíme $r_{s,z,t}$) svým zákazníkům. S továrnami to také snadné není, protože každá továrna d dokáže vyprodukovat pouze omezený počet produktů daného typu $p_{d,z,t}$ v závislosti na tom, kolik jim dodavatelé daný den zrovna (například kvůli koronaviru) dodají materiálu a také jak se zrovna chce zaměstnancům do práce. V továrně $d = 2$ v Mostu se navíc poslední dobou často stává, že cena převozu zboží z továrny do supermarketu je dána pouze továrnou a supermarketem (jeden kus libovolného zboží lze převézt mezi dvěma uzly s konstantní cenou $c_{s,d}$, tedy například převoz jednoho záchodového prkénka, dvou matrijóšek a dvou skříní z továrny v Mostu do supermarketu BOBI na Borech stojí $5c_{5,2}$).

Mladík však převzal společnost po otci náhle (otec mu řekl, že to nedokáže lépe, protože už to dělá léta, ať se tedy ukáže), a v supermarketech tedy zbyly zásoby zboží $q_{s,z}$. Pomozme Lubošovi naplánovat přepravu zboží pro T dní dopředu. Luboš zná:

D – počet továren,

S – počet supermarketů,

Z – počet druhů zboží,

T – počet dní,

$c_{s,d}$ – cenu převozu jednoho kusu libovolného zboží z továrny d do supermarketu s ,

$q_{z,s}$ – počáteční zásoby výrobku z v supermarketu s .

$p_{d,z,t}$ – produkci továrnou d druhu zboží z v den t ,

$r_{s,z,t}$ – poptávku zákazníků po zboží z v supermarketu s dne t ,

a chce znát $k_{s,d,z,t}$ pro $\forall s \in \{1, \dots, S\}, d \in \{1, \dots, D\}, z \in \{1, \dots, Z\}$ a $t \in \{1, \dots, T\}$, tedy počty kusů všech druhů zboží z rozvezených ze všech továren d do všech supermarketů s každý den t . Pokud by se stalo, že továrny nedokáží zásobit supermarket, Luboš potřebuje znát, kdy to bude (den t), aby doobjednal zboží z Číny.

Supermarkety mají povoleno prediktivní plánování s neomezenými zásobami (někteří tomu mohou říkat „křečkování“) – lze tedy předzásobit supermarket dle dat poptávky z „budoucnosti“. Jestli toho využijete či nikoliv je na Vás, je ale třeba toto uvést do dokumentace (viz dále).

Vytvoření funkčního programu:

- Seznamte se se strukturou vstupních dat (počty, ceny cest, objem poptávek...) a načtěte je do svého programu. Formát souborů je popsán přímo v záhlaví vstupních souborů. **(5b.)**.
- Navrhněte vhodné datové struktury pro reprezentaci vstupních dat, zvažujte časovou a paměťovou náročnost algoritmů pracujících s danými strukturami **(10b.)**.
- Proveďte základní simulaci pro první den, vypište celkovou cenu přepravy prvního dne pomocí výrazu $\sum_{s=1}^S \sum_{d=1}^D \left(c_{s,d} \sum_{z=1}^Z k_{s,d,z,1} \right)$ **(10b.)**.

Výše popsaná část bude váš minimální výstup při kontrolním cvičení cca v polovině semestru.

- Vytvořte prostředí pro snadnou obsluhu programu (menu, ošetření vstupů) – nemusí být grafické, umožněte manuální zadání požadavku **(5b.)**,
- umožněte sledování (za běhu simulace) aktuálního stavu přepravy různých druhů zboží a dopravního prostředku – (dopravní prostředky jsou libovolné, např. nákladní vůz, vlak, loď, lamy, saně apod.) **(5b.)**,
- proveďte simulaci pro zadaný počet dnů a vygenerujte do souborů následující statistiky (uložte je do vhodných souborů **10b.**):
 - přehled jednotlivých továren – rozpis, co kam a kdy produkovaly, kolik zboží vyprodukovaly zbytečně
 - přehled jednotlivých supermarketů – každodenní rozpis skladových zásob
 - celková cena přepravy $\sum_{s=1}^S \sum_{d=1}^D \left(c_{s,d} \sum_{z=1}^Z \sum_{t=1}^T k_{s,d,z,t} \right)$
 - celková doba běhu simulace
 - nemá-li úloha řešení, vypište, který den již nebylo možné supermarkety zásobit a kolik zboží kde chybělo.
- Vytvořte generátor vlastních dat. Generátor bude generovat vstupní data pomocí rovnoměrného, normálního nebo extrémního rozdělení (zvolte jedno z uvedených rozdělení, podrobnosti poskytnou cvičící na cvičeních) s vhodnými parametry. Data budou generována do souboru (nebudou přímo použita programem) o stejném formátu jako již dodané vstupní soubory. Při odevzdání přiložte jeden dataset s řešitelnou úlohou a jeden dataset, kdy se nepodaří supermarkety zásobit **(5b.)**.
- Vytvořte dokumentační komentáře ve zdrojovém textu programu a vygenerujte programovou dokumentaci (Javadoc) **(10b.)**,
- vytvořte kvalitní dále rozšiřitelný kód – pro kontrolu použijte softwarový nástroj PMD (více na <http://www.kiv.zcu.cz/~herout/pruzkumy/pmd/pmd.html>), soubor s pravidly `pmdrules.xml` najdete na portálu v podmenu Samostatná práce **(10b.)**
 - mínus 1 bod za vážnější chybu, při 6 a více chybách nutno opravit,
 - mínus 2 body za 10 a více drobných chyb.

V rámci dokumentace:

- připojte zadání **(1b.)**,
- popište analýzu problému **(6b.)**,
- popište návrh programu (např. jednoduchý UML diagram) **(6b.)**,
- vytvořte uživatelskou dokumentaci **(5b.)**,
- zhodnoťte celou práci, vytvořte závěr **(2b.)**.