

LOGISTIKADA INTEGRATSIYALASHGAN AXBOROT OQIMLARI ALMASHINUVI

Bahrom Yangibayevich Ishmetov

Muhammad al-Xorazmiy nomidagi Toshkent axborot texnologiyalari universiteti
Urganch filiali

ishmetovbahrom2123@gmail.com

ANNOTATSIYA

Logistika kompaniyasining mijozlar bilan axborot aloqasini optimallashtirish taqdim etilayotgan xizmatlar sifatini sezilarli darajada oshirishi mumkin. Bugungi kunda bunday o'zaro ta'sir qanday yo'llar bilan amalga oshirilmoqda? Ularning afzalliklari va kamchiliklari qanday? Integratsiyalashgan onlayn xizmatlar O'rtacha 2PL yoki 3PL logistika operatori o'z ma'lumotlar bazasida kamida 250 mijozga ega. O'zingizning CRM tizimingizni mijozlar bazasiga integratsiya qilishdagi asosiy qiyinchilik kompaniyaning asosiy, asosiy mijozlarini aniqlashdir. Faqat logistika kompaniyasidagi barcha jarayonlarni tavsiflash va standartlashtirish prizmasi orqali tushunish mumkin ma'lumotlar oqimining barqarorligini qanday ta'minlash va axborot bo'shliqlarini oldini olish, shuningdek, mijoz uchun yangi texnologiyalar va xizmatlarni joriy etish. Menga ishonib, qayta-qayta chalg'itib, dastlab noto'g'ri bo'lgan axborot texnologiyalaridagi kamchiliklarni tuzatishga pul sarflashdan ko'ra, jarayonlarni tavsiflash va standartlashtirishga mablag' va resurslarni bir marta sarflash samaraliroqdir.

Kalit so'zlar: Integratsiya, Logistika, Oqim, Protsedura, Algoritm, Elektron, Statistika, Dinamik, Effekt, Graf.

EXCHANGE OF INTEGRATED INFORMATION FLOWS IN LOGISTICS

ABSTRACT

Optimizing the information communication of a logistics company with customers can significantly increase the quality of the provided services. In what ways is such interaction taking place today? What are their advantages and disadvantages? Integrated Online Services The average 2PL or 3PL logistics operator has at least 250 customers in its database. The main challenge in integrating your CRM system into your customer base is identifying your company's key, core customers. Only through the prism of description and standardization of all processes in a logistics



company can it be understood how to ensure the stability of the flow of information and prevent information gaps, as well as the introduction of new technologies and services for the client. Believe me, it is more effective to spend money and resources once on the description and standardization of processes than to spend money on correcting the shortcomings of information technology that were wrong in the first place, over and over again.

Keywords: Integration, Logistics, Flow, Procedure, Algorithm, Electronic, Statistical, Dynamic, Effect, Graph.

KIRISH

Axborotlarni yig'ishda va qayta ishlashda bevosita tizim va undagi dasturiy vositalarini foydalaniladi. Bunda, turli toifadagi foydalanuvchilar turli shakldagi axborotlarni tizimdan olishlari va yangi turdagi axborotlarni uzatishlari mumkin. Natijada elektron shakldagi axborotga bo'lgan talab ortib boradi. Elektron shakldagi axborotlarni tizimdan olib foydalanish oson hamda axborotlarni tezkor olish va jo'natish imkoniyatlari mavjud. Bunday qulaylik axborotlar almashish jarayonida tizimda axborot oqimlarini vujudga keltiradi. Tizimdan foydalanuvchilarni soni ko'payib ketganda axborot oqimlarini ko'payib ketishi kuzatiladi va bu tizimning turg'unligini buzilishiga olib keladi. Natijada axborot saqlovchi va qayta ishlovchi vositalar tirbandligi va ba'zi axborotlar yo'qolib ketishi kuzatiladi. Bunday jarayonlarni sodir bo'lishi tashkilotlarni ish faoliyatiga salbiy ta'sir ko'rsatib bir qator muammolarni vujudga keltirishi mumkin. Muammoni bartaraf qilish uchun axborot saqlovchi vositalarni texnik imkoniyatlarini oshirish bilan yechib bo'lmaydi. Balki tizimlarda ishlatiladigan dasturlarga yangi algoritm kiritish yoki mavjudlarini mukammallashtirish orqali erishiladi.

Ammo axborot oqimlarini optimallashtirish zarur bo'lgan sohalar yoki biznes jarayonlarini qanday aniqlash mumkin? Kompaniya xizmatlarining ajralmas qismi sifatida axborot logistikasini tashkil qilishni qanday boshlash kerak? Agar ma'lumotlar harakatini optimallashtirish jiddiy moliyaviy va vaqt xarajatlarini talab qilishini va tartibsiz va turli xil ma'lumotlarga asoslangan tizimni qurishning o'zi mumkin emasligini hisobga olsak, bu savollarga javoblar yanada muhimroq bo'ladi. Biroq, zamonaviy bozor qaror hech qachon yagona to'g'ri va shubhasiz qaror emasligini o'rgatadi. Bu yuqoridagi savollarga ham tegishli. Shu bilan birga, o'z-o'zini rivojlantirish tizimlarini qurish bo'yicha o'z tajribamga asoslanib, optimallashtirishning markaziy, majburiy bosqichini ta'kidlamoqchiman. Faqat logistika kompaniyasidagi barcha jarayonlarni tavsiflash va standartlashtirish prizmasi orqali

tushunish mumkin ma'lumotlar oqimining barqarorligini qanday ta'minlash va axborot bo'shliqlarini oldini olish, shuningdek, mijoz uchun yangi texnologiyalar va xizmatlarni joriy etish. Menga ishonib, qayta-qayta chalg'itib, dastlab noto'g'ri bo'lgan axborot texnologiyalaridagi kamchiliklarni tuzatishga pul sarflashdan ko'ra, jarayonlarni tavsiflash va standartlashtirishga mablag' va resurslarni bir marta sarflash samaraliroqdir. Shunday qilib, har bir kompaniya ISO sifat standartlariga asoslanib, ichki jarayonlarni (asosiy va yordamchi) tavsiflashi va ularning har biri uchun ajralib turadigan xususiyatlarni ajratib ko'rsatishi mumkin. Keyinchalik, ushbu xususiyatlarni yig'ish protseduralari mijoz tomonidan jalb qilingan xizmatlar uchun ham, har bir jarayon uchun ham yozilishi va amalga oshirilishi kerak. Olingan statistik ma'lumotlar kompaniyaning amalga oshirilayotgan jarayonlaridagi zaif tomonlarni aniqlashga, axborot tizimining konturlari va asoslarini belgilashga va tegishli dasturiy mahsulotni tanlashga imkon beradi. Bu qilinadigan asosiy narsa. Masalan, STS Logistics kompaniyasi jarayonlar monitoringi natijalariga ko'ra o'zining IT bo'limini tashkil etishga qaror qildi. Ushbu bo'limning asosiy vazifasi ichki va tashqi axborot oqimlarini tizimlashtirgan dasturiy mahsulotni yaratish va takomillashtirishdan iborat. Bundan tashqari, u mijozlardan ma'lumotlarni yig'ish va bizning axborot tizimimizni mijozlar dasturiy mahsulotlariga integratsiya qilish uchun javobgar edi. Rejaga ko'ra, IT-mahsulotlar funkcionalligini yanada rivojlantirish va ma'lumotlarni kiritish va qayta ishlash uchun vaqt xarajatlarini qisqartirish kompaniyani 3PL- va hatto 4PL-logistika xizmatlarini provayderlari darajasiga olib chiqishga yordam berishi kerak. Kiruvchi yuklar haqida logistika uchun eng oddiy ma'lumotlarni tizimlashtirish axborot tuzilmalari bilan ishlashni boshlashi mumkin.

ADABIYOTLAR TAHLILI VA METODOLOGIYA

Axborotlarni shakllantirish va ularni oqimlarni boshqarishda ma'lumotlarni intellektual tahlili (MIT) masalalarini yechish, katta hajmdagi ma'lumotlarni tahlil qilish va boshqarish bo'yicha tadqiqotchilar AQSH, Buyuk Britaniya, Rossiya, Xitoy, Germaniya, Avstriya, Shvesiya, Ispaniya, Finlyandiya, Kanada, Janubiy Koreya va boshqa mamlakatlar olimlari tomonidan shiddat bilan olib borilmoqda va ma'lum natijalarga erishilmoqda.

Hozirda ko'pgina xorij universitetlarida "Information flows" axborot oqimi, "Traffic flows" transport harakat oqimlari, "Water flows" suv oqimlar va "Data flows in network" tarmoqda ma'lumotlar oqimi masalalarni ma'lumotlarni intellektual tahlil qilish instrumentlarida va optimallashtirish algoritmlardan foydalanib yechish aktual mavzularga



aylangan. Bundan tashqari bu yo'nalish bo'yicha horijiy olimlardan Munakata, T. & D. J. Hashier, Inagaki, J., M. Haseyama & H. Kitajima, Gen, M., R. Cheng & S.S. Oren, Ford-Falkerson, Denist, Gen, M., R. Cheng & S.S. Oren, Edmonds – Kapra, Karzanov, Malxotra – Kumar – Maxasherava, Golberg – Roalar, Zhou, G. & M. Gen, H. Ishibuchi and T. Nakashima, Xorn, D.A.Forsayt, U.Prett, I.S.Gruzman, T.Pavlids, R.S.Gonsales, R.YE.Vuds, D.Rodjers, Dj.Adams kabilarni sanab o'tish mumkin, O'zbekistonda esa bu sohada V.Qobulov, M.Komilov, SH.Tulyaganov, T.F.Bekmurodov, SH.X.Fazilov, S.S.Sodiqov, A.X.Nishonov, SH.Nazirov N.Mirzayev, M.N.Malikov, I.R.Samandarov va boshqalar ilmiy izlanishlar olib bormoqdalar.

NATIJALAR

Jadvalga muvofiq harakatni nazorat qilish. Yuk tashishning dastlabki jadvalini ishlab chiqish va unga aniq rioya qilish mijozni yutib olish va saqlab qolish yo'lidagi birinchi qadamdir. O'zingizning axborot ilovalaringizni mijoz tizimiga integratsiyalashni boshlashingiz mumkin bo'lgan qo'shimcha xizmat rovida, bu holda mijozni tovarlarni etkazib berish vaqti haqida xabardor qilish uchun yaxshi ishlaydigan tizim ishlaydi. Shu bilan birga, mijozlarning qabul qilingan tasdiqlashni kodlash va qayta ishlash usullari bo'yicha istaklarini hisobga olish kerak. Joriy jarayonni xizmatni amalga oshirishning keyingi bosqichi bilan muvofiqlashtirish. Ushbu bosqichda ma'lumotning asosiy qiymati xizmatni amalga oshirish yoki ishlab chiqarishdan oldin ham rejadan yoki belgilangan parametrlardan har qanday og'ishlarni aniqlashdir. Xizmatlarni ishlab chiqarishning barcha bosqichlarini har doim uch bosqichga bo'lish mumkin:

- xizmatni sotish, uni mijozga ko'rsatish rejasini shakllantirish, sifat mezonlarini ishlab chiqish yoki umumiy qabul qilingan sifat standartlariga rioya qilish;
- xizmatlarning haqiqiy taqdim etilishi;
- tranzaksiyani moliyaviy yakunlash, xizmat ko'rsatish sifati bo'yicha mijozlarning fikr-mulohazalarini olish va bitimning hujjatli bajarilishi.

Har bir bosqichni eng kichik detallarigacha tafsilotlash mumkin. Tafsilotlar darajasi tizimni ishlab chiquvchilarga bog'liq. Biroq, har qanday holatda, xizmatni amalga oshirish tugagunga qadar noto'g'ri amalga oshirilishining oldini olish uchun ma'lumot kerak. Siz uni xizmat ko'rsatish bosqichlarining kesishgan joylaridagi nazorat punktlaridan ko'rsatkichlarni olish orqali olishingiz mumkin. Ma'lumotlar bir vaqtning o'zida kompaniyaning sifat bo'limiga va mijozga jarayonni nazorat qilishingizga ishontirish



uchun uzatiladi. Axir, ishonch mijozning nutq operatoriga bo'lgan ishonchining kalitidir. Ideal holda, bu ma'lumotlar yagona ma'lumotlar bazasida saqlanishi va bitta dasturiy ta'minot manbasidan ichki va tashqi iste'molchilarga etkazilishi kerak. Joriy inventar iste'molini sozlash. Shubhasiz, Har bir bosqichda texnologik yoki ishlab chiqarish ko'rsatkichlariga qo'shimcha ravishda (ISO sifat nazorati tizimida - jarayon mezonlari) kompaniyaning moliyaviy oqimlarini rejalashtirish va kerak bo'lganda tuzatish muhim ahamiyatga ega (pul oqimi). Shu sababli, bitimning har bir bosqichi oldindan ishlab chiqilishi kerak va yagona ma'lumot maydoni orqali o'tkazilgan mablag'larning muddatlari va miqdoridagi og'ishlar to'g'risidagi ma'lumotlar jarayonning barcha ishtirokchilari uchun mavjud bo'lishi kerak. Tajriba shuni ko'rsatadiki, moliyaviy va texnologik mezonlarni hisobga olish uchun axborot mahsulotini bloklarga bo'lish shart emas, chunki bu ko'rsatkichlar o'zaro bog'liqdir. Saqlash uchun resurslarni tayyorlash va yuklash va tushirish operatsiyalarini tashkil etish (PRR). Yukning harakati to'g'risidagi ma'lumotlar tovarlarni omborga joylashtirishda log operatori uchun ham, mijozning o'zi uchun ham tegishli; chunki u PRRni saqlash va tashkil etish uchun ombor maydonidan foydalanishni rejalashtirish, xarajatlarni minimallashtirish imkonini beradi. Sotishni rejalashtirish. Yuk tashish jadvallari haqida o'z vaqtida ma'lumotni qulay tarzda olgan mijoz ham, kelajakdagi va allaqachon tugallangan tranzaksiyalar haqida ma'lumotga ega bo'lgan logistika kompaniyasi ham o'z savdolarini ishonchli rejalashtirishi mumkin. Va bu holda, tranzaksiyalarni kuzatish sxemalarini mijoz tizimiga integratsiyalashuvi kelajak uchun muhim zaxiraga aylanishi mumkin. Axir, mijozning savdo bo'limlari turli transport xizmatlaridan ma'lumot yig'ishdan ko'ra, barcha yuklar bo'yicha ma'lumotlar yig'iladigan yagona axborot maydonida ishlashi ancha oson va tezroq. Transport xavfsizligini nazorat qilish. Albatta, yuk haqida ma'lumot o'z vaqtida olingan taqdirdagina tashishning har bir bosqichi xavfsizligini ta'minlash mumkin.

Axborot oqimlarini boshqarishda marshrutizatsiya algoritmlarini qo'llanilishi bu graf tugunlariaro marshrutlarni aniqlash telekommunikatsiya yo'nalishi, transport, rejalashtirish va boshqa xabar almashish bilan bog'liq bo'lgan muammolarni yechishda dolzarb hisoblanadi. Odatda, bir parametrga bog'liq holda masalani yechish algoritmlari juda ko'p uchraydi. Ko'p parametrliligi, graf tugunlari orasidagi bog'lanish, ko'p parametrga bog'liq holda masalani yechish yetarlicha o'rganilmagan. Ko'p parametrliligi marshrutizatsiya masalasini yechish murakkab va hozirgi kunning dolzarb masalalaridan hisoblanadi.

Qaralayotgan parametrlarni shartli ravishda statik va dinamik guruhga ajratamiz:

Statik guruh parametrlari – graf tugunlarini o‘zaro o‘zgarimas parametrlar bilan doimiy bog‘lashnini ta‘minlaydi.

Dinamik guruh parametrlari – graf tugunlarini o‘zaro o‘zgaruvchan parametrlar bilan vaqt oralig‘idagi bog‘lanishini ta‘minlaydi.

Graf ko‘rinishida berilgan masalalarda statik va dinamik parametrlarni birgalikda hisobga olgan holda marshrutizatsiya, mablag‘ bilan bog‘liq resurslar taqsimoti kabi masalalarni yechish axborot texnologiya sohasining dolzarb muammolaridan hisoblanadi.

Ushbu maqolada resurslar: grafning statistik va dinamik parametrlari; vaqt bo‘yicha cheklanganlik; graf tugunlari aro bandlik hisobga olingan holatlarda, dinamikada marshrutizatsiya masalasini yechishning effektiv algoritmi ishlab chiqilgan.

Masalani yechishda Dekstra algoritmini modifikatsiya qilish orqali mukammal yechimlar topilishi asoslangan. Mukammal yechimni aniqlashda dinamik tuzilmalardan foydalanilgan.

Faraz qilaylik, ko‘p, aralash (statik va dinamik) parametrli graf berilgan bo‘lsin. Grafning statik va dinamik bog‘lanishni ta‘minlaydigan parametrlarini inobatga olgan holda graf bir tugunidan boshqa tuguniga eng kam sarf harajat bilan o‘tish, tarmoq marshrutizatsiya masalasining asosi hisoblanadi. Demak, asosiy gipoteza tarmoqda shunday marshrut topilishi zarurki, bu marshrutdan maksimum ma‘lumotlar uzatilib minimum sarf xarajat qilinsin.

Parametrlarning maqsad funksionalini maximumga intiltiruvchi qismini, ya‘ni qo‘yilgan maqsad bilan bog‘liq qismini musbat va minimumga intiltiruvchilarini, ya‘ni sarf-xarajatlari bilan bog‘liq qismini esa manfiy son ko‘rinishida ifodalaylik.

Masala. Dinamik vaqt oralig‘ida ko‘p, aralash parametrli graf va uning tugunlararo bandlik, vaqt hamda resurs bo‘yicha cheklanganligini inobatga olgan holda tugunlararo eng qisqa L ta zaxira yo‘llar marshrutini aniqlashning effektiv algoritmi ishlab chiqilsin.

Ko‘p parametrli grafda resurs cheklangan holatlarda marshrutizatsiya masalasini yechish algoritmi

Quyidagicha belgilashlar kiritaylik. Graf $G=(N,C,D)$ ko‘rinishda berilgan bo‘lsin. Bu yerda N - grafdagi tugunlar soni, S -statik va D -esa dinamik parametrlar majmuasi. Statik va dinamik parametrlar uch o‘lchovli matritsa ko‘rinishida berilgan bo‘lsin.

Uch o‘lchovli $\{C_{uij}\}$ - ($u,i,j : u \in U, i \in N, j \in N$) – statik parametrlar majmuasi. Bu yerda U - statik parametrlar soni, $N \times N$ – matritsa o‘lchovi. $\{C_{uij}\}$ – uch o‘lchovli matritsa i - tugundan

j - tugungacha bo'lgan bog'lanishni u – statik parametr orqali ta'minlaydi.

Uch o'lchovli $\{D_{hij}\} - (h, i, j : h \in H, i \in N, j \in N)$ – dinamik parametrlar majmuasi. – i - tugundan j - tugungacha bo'lgan bog'lanishni h – dinamik parametr orqali ta'minlaydi. $\{D_{1ij}, D_{2ij}, D_{3ij}, \dots, D_{Hij}\}$ – dinamik parametrlar soni H ga teng.

Dinamik parametrli matritsa elementlari vaqtga bog'liq diskret ifodalanadi. Masalan, vaqtda matritsaning D_{hij} parametrlari quyidagicha ifodalanishi mumkin:

$$D_{hij}(t) = D_{hij}(t-1) + \dots \quad (1)$$

Bu yerda uch o'lchovli matritsaning D_{hij} parametri, elementi qiymati diskret vaqt oralig'ida Δt ga o'zgarishini bildiradi.

Bundan ko'rinib turibdiki vaqtga bog'liq holda uch o'lchovli dinamik matritsa elementlari doimo o'zgarib turadi ya'ni bir holatdan boshqa holatga o'tganda matritsasi elementlari Δt ga farq qiladi.

Demak, qo'yilgan masalani yechishda dinamik parametrlarni o'zgarib turishi umumiy yechimni olishda xatolikka olib kelishi mumkin. Shuning uchun masalani yechish ikki bosqichdan iborat. Birinchi bosqichda statik parametrlardan foydalanilgan holda masala yechiladi, ikkinchi bosqichda esa topilgan yechimlar dinamik parametrlar asosida qayta ishlanadi va umumiy yechim topiladi.

Birinchi bosqichda statik parametrlar umumiy koeffitsentga olib kelinadi va L ta eng qisqa marshrut aniqlanadi. Topilgan marshrutlar asosida dinamik parametrlar qo'llanilib L ta eng yaxshi marshrut taklif qilinadi. Bu yerda L .

Statik parametrlar umumiy koeffitsentga quyidagicha olib kelinadi ya'ni uch o'lchovli matritsadan ikkita ikki o'lchovli matritsaga o'tkaziladi va yangi matritsalar koeffitsiyentlari maqsadga ko'ra quyidagicha hisoblanadi.

Bu yerda, $-i$ - tugun bilan j - tugun orasidagi qiymatlari manfiy bo'lgan statik parametrlarining modulli ko'paytmasi. $-i$ - tugun bilan j - tugun orasidagi qiymatlari musbat bo'lgan statik parametrlar ko'paytmasi.

Statik parametrlarning umumiy koeffitsentlarini ifodalovchi matritsani Q – orqali, uning elementlarini esa a_{ij} orqali belgilaylik. U holda

$$(4)$$

Q - matritsa elementlari grafning biror bir i - tugunidan j - qo'shni tuguniga o'tishning bir o'lchamli koeffitsentini beradi. Umumiy koeffitsentni topishda quyidagi ifoda o'rinli. Demak, berilgan grafda a_{ij} umumiy koeffitsent bo'yicha eng qisqa marshrutni topsak, berilgan statik parametrlarni inobatga olib masalani yechgan bo'lamiz.

Grafda bir tugundan boshqasiga boradigan eng qisqa bitta yo'lni Deykstra algoritmi orqali aniqlanadi. Deykstra algoritmiga



o'zgartirish kiritib, grafda bir tugundan boshqasiga boradigan eng qisqa Lta yo'l aniqlanadi.

Grafdagi har bir tugunga tuzilma kiritib olinib, bu tuzilmalarda har bir tugunga keluvchi eng yaxshi yo'llarni L tasini saqlab qo'yiladi. Natijada, tuzilmadan bir tugundan boshqasiga boradigan eng qisqa L ta yo'l aniqlanadi. Statik parametrlar asosida topilgan eng qisqa yo'llar fiksirlanib, masala dinamik parametrlari bo'yicha yechiladi va ikkala parametrlar majmuasini qanoatlantiradigan, foydalanishga qulay bo'lgan marshrut ajratib olinadi.

Ajratib olingan marshrut qancha vaqt foydalanilishini bilish uchun dinamik matritsaga kiritib qo'yiladi, chunki masala yechilayotganda yo'lning bandligi ham inobatga olingan bo'ladi. Bundan asosiy maqsad:

boshqa tugunlarga o'tishlarni amalga oshirishda undan foydalanmaslikni ta'minlash;

marshrutni qancha vaqt band bo'lib turishini aniqlash;

tarmoqni qanchalik yuklanish bilan ishlayotganini aniqlash.

Yuqorida ifoda etilgan ma'lumotlar asosida ko'p parametrli grafda resurs cheklangan holatlarda marshrutizatsiya masalasini yechish algoritmini quyidagi psevdakodlar orqali ifodalaymiz.

main() – bu asosiy psevdakod bo'lib, quyidagilarni o'z ichiga oladi:

Input() – boshlang'ich ma'lumotlarni kiritish uchun ishlatiladi. Bu yerda graf bog'lanishlar navbat tuzilmalari orqali shakllantiriladi.

FindPaths() – statik parametrlar bo'yicha barcha tugunlardan barchasiga boradigan eng yaxshi L ta zaxira marshrutlar aniqlanib result tuzilmasiga saqlab boradi. Ushbu algoritm o'z navbatida FindPath(i) psevdokodiga murojaat qiladi.

FindPath(i) – result tuzilmasini shakllantirish uchun ishlatiladi. Bu yerda i – tugundan boshqa barcha tugunlargacha bo'lgan mukammal marshrutni aniqlashda foydalaniladi.

LoopLife() – result tuzilmasidan foydalangan holda marshrutlarni dinamik parametrlari bo'yicha bandligini tekshiradi va dinamik matritsa o'zgarishlarini yangilab boradi.

Main qism dasturi

Main qism dasturi barcha glabal o'zgaruvchi, o'zgarmaslar va psevdakodlarning ketma - ketligini ifodalash uchun qo'llaniladi.

1: N, L – butun va glabal o'zgaruvchilar

2: ages[] – list tuzilma va glabal o'zgaruvchi

3: correcting = 10000 – o'zgarimac

4: result[][][] – 3 o'lchamli list tuzilma


```

5: void main() {
6:   Input();
7:   FindPaths();
8:   LoopLife();
   }

```

Dasturning 1,2,3,4 – qatorlari global o‘zgarmas va o‘zgaruvchilarni tavsiflash uchun qo‘llanildi, 6, 7, 8 – qatorlari esa psevdakodlarga ketma – ket murojaatlarni amalga oshiradi.

Input orqali boshlang‘ich qiymatlar: S matritsa; NxN - matritsa o‘lchami kiritiladi. Ma’lumki, S matritsa elementlarining barchasi ham noldan farqli bo‘lmaydi, chunki u graf ko‘rinishda berilgan bo‘lib, ushbu graf tugunlari aro barcha bog‘lanishlar mavjud emas. Shuning uchun, faqatgina matritsaning noldan farqli elementlari kiritiladi.

N o‘lchovli kvadrat matritsa xotiradan katta joyni egallaydi va buni inobatga olib, S matritsani ages tuzilma bilan almashtirsak xotira xajmini yetarlicha tejaymiz.

```

1: void Input()
2: {
3:   N = nextInt();
4:   m = nextInt();
5:   k = nextInt();
6:   L = nextInt();
7:   ages = new ArrayList[N+1];
8:   for i = 1 to N {
9:     ages[i] = new
ArrayList<Point>();
10:  }
11:  for i = 1 to m {
12:    u = nextInt();
13:    v = nextInt();
14:    s = 1, p=1;
15:    for j = 0 to k {
16:      pr = nextInt();
17:      if (pr>0) s=s*pr; else
p=p*Math.abs(pr);
18:    }
19:    KK = correcting*p/s;
20:    ages[u]->qo'sh(newPoint(v, KK));

```

```
21: ages[v]->qo'sh(newPoint(u, KK));
```

```
22: }
```

```
23: }
```

Yuqorida keltirilgan psevdakodlarning 3,4,5,6 – qatorlari o‘zgaruvchilar qiymatlarini kiritib olish uchun ishlatiladi. Bu yerda N – grafdagi tugunlar soni, m esa tugunlararo bog‘lamlar soni, k – tugunlararo statik parametrlar soni, L – tugunlararo zaxira marshrutlar sonini bildiradi.

Psevdakodning 7, 8, 9 – qatorlari ages tuzilmasiga joy ajratib beradi, ages tuzilmasining har bir elementi 2 ta qiymatni saqlaydi.

Psevdakodning 11 – 23 – qatorlari oralig‘ida u tugundan v tugungacha statik parametrlar kiritiladi, umumiy koeffitsent topiladi va ages tuzilmasiga nuqta ko‘rinishida saqlab boriladi.

FindPaths qism dasturi

FindPaths qism dasturi ages tuzilmasidan, boshlang‘ich ma‘lumot sifatida foydalangan holda tugunlararo zaxira yo‘llarni topadi va result tuzilmasini hosil qiladi hamda uni to‘ldiradi. Bu tuzilmada barcha tugunlardan barchasiga boradigan eng yaxshi marshrutlar saqlanadi.

```
1: void FindPaths()
```

```
2: {
```

```
3: result = new ArrayList[N+1][N+1][L+1];
```

```
4: for i = 1 to N {
```

```
5:     FindPath(i);
```

```
6: }
```

```
7: }
```

Psevdakodning 3 – qatori result tuzilmasiga joy ajratadi va FindPaths usulidan FindPath(i) usuliga murojatni amalga oshiradi. FindPath usuli i – tugundan boshqalariga boradigan barcha eng yaxshi marshrutlarni result tuzilmasiga joylashtiradi.

FindPath qism dasturi

FindPath qism dasturi $v1$ tugundan boshqalariga boradigan L ta zaxira marshrutlarni aniqlaydi. Zaxiralarni aniqlash modifikatsiya qilingan Dekstra algoritmi yordamida amalga oshiriladi. Ma‘lumki, Dekstra algoritmi bir tugundan boshqalariga boradigan eng qisqa bitta masofani topishda ishlatiladi. Demak, protsedurani bajarish davomida eng qisqa yo‘lga yaqin bo‘lgan boshqa yo‘llar eslab qolinsa, bir tugundan boshqalariga boradigan barcha marshrutlar ham aniqlanadi.

```
1: void FindPath(v1) {
```

```

2:d - ∞
3:d[v1][1] = 0;
4:used = new boolean[N+1][L+1];
5>List = new ArrayList[N+1][L+1];
6: for i = 1 to N {
7:   for j = 1 to L {
8:     List[i][j] yaratish;
9:   }
10: }
11>List[v1][1]->qo'sh(v1);
12:for i = 1 to N*L {
13: v = 0, k = 0;
14: for j = 1 to N {
15:   for j2 = 1 to L {
16:     if (!used[j][j2]&&
(v==0||d[j][j2]<d[v][k])) {
17:       v = j;
18:       k = j2;
19:     }
20:   }
21: }
22: used[v][k] = true;
23: if (d[v][k]==INF)
24: break;
25: for (Point to : ages[v]) {
26:   for j = 1 to L {
27:     if(d[v][k]+to.y<d[to.x][j]
va not to.x in List[v][k]){
28:       for j2 = L downto j {
29:         d[to.x][j2]=d[to.x][j2-1];
30:         List[to.x][j2->tozalaymiz;
31:         List[to.x][j2->
qo'sh(List[to.x][j2-1]);
32:       }
33:       d[to.x][j] = d[v][k]+to.y;
34:       List[to.x][j]->tozalaymiz;
35:       List[to.x][j]->

```



```

add(List[v][k]);
36: List[to.x][j]->add(to.x);
37: break;
38: }
39: }
40: }
41: }
42: for i = 1 to N {
43: for j = 1 to L {
44: result[v1][i][j] = new
ArrayList<Integer>();
45: result[v1][i][j]->
qo'sh(List[i][j]);
46: }
47: }
48:}

```

Ushbu psevdakod sun'iy d matritsa, used mantiqiy tuzilma, List tuzilmalari yordamida v1 tugundan boshqa tugunlarga boradigan L zaxira yo'llarni aniqlaydi va result tuzilmasiga saqlaydi.

LoopLife qism dasturi

LoopLife qism dasturi - result tuzilmasi ma'lumotlari asosida, tugunlararo tarmoq bandligi va foydalanuvchi ehtiyojlarini inobatga olgan holda graf tugunlarida marshrutlarni mukammal tanlash imkonini beradi. Agar, eng yaxshi marshrut yoki uning biror qismi band bo'lsa, u holda masalani yechishda taklif qilinayotgan algoritm zaxira marshrutlarni inobatga oladi. Vaqt o'tishi bilan band yo'llar ochilsa, u holda algoritm shu vaqtdan boshlab, ushbu marshrutlarni hisobga olgan holda ishlaydi.

```

1:void LoopLife()
2:{
3:d = new long[N+1][N+1];
4: while (true)
5: {
6: u = nextInt();
7: v = nextInt();
8: t = nextLong();
9: println(u+"dan"+v+"ga boradigan"
+L+" ta qisqa yo'l : ");

```

```

10: indexPath = 0;
11: do
12: {
13: for i = 1 to L {
14: if (result[u][v][i]=null yoki
bosh { 15: break; 16: }
17: path = new ArrayList<Integer>();
18: path->qo'sh(result[u][v][i]);
19: pathTime->d[path->(0)][path->(1)];
20: for j -> 1 to path->size()-1 {
21: if(pathTime <
d[path->(j)][path->(j+1)])
22: pathTime=d[path->
(j)][path->(j+1)];
23: }
24: if (pathTime<
System.currentTimeMillis()){
25: indexPath = i;
26: break;
27: }
29: }
30: }while (indexPath=0);
31: path = new ArrayList<Integer>();
32: path->
qo'sh(result[u][v][indexPath]);
33: t=1000*t+
System.currentTimeMillis();
34: for j = 0 to path->size()-1 {
35: print(path->(j)+" ");
36: d[path->(j)][path->(j+1)]=t;
37: }
38: println(path->(path->size()-1));
39: }
40:}

```

Algoritmi testlash

Misol uchun tugunlari 7 ta, tugunlararo bogʻlamlar 12 ta va tugunlararo statik parametrlar soni 4 ta boʻlgan graf berilgan boʻlsin. Tugunlararo zaxira marshrutlar son 6.

$$N=7, m=12, k=4, L=6$$

Statik oʻzgaruvchilar

№ u v C1 C2 C3 C4

1 1 2 50 2 -100 -2

2 1 3 110 4 -110 2

3 2 3 80 4 -90 -6

4 2 4 40 2 -100 -2

5 2 5 150 6 -150 -1

6 3 5 150 6 -150 -1

7 3 6 120 6 -160 -2

8 4 5 100 2 -20 -3

9 4 7 80 2 -90 -3

10 5 6 90 4 -30 -1

11 5 7 100 4 -80 -1

12 6 7 100 6 -90 -2

Algoritmga kiritilgan parametrlarga nisbatan natijalarni kuyidagicha koʻrish mumkin: u – tugundan v – tugunga boradigan marshrutni topish talab etiladi. Bu yerda t topilgan marshrut qancha vaqt band boʻlishini bildiradi. Bu ifoda dinamik matritsani hosil qilishda muhim hisoblanadi.

1 7 20

1 dan 7 ga boradigan 6 ta qisqa yoʻl : 1 3 5 7

1 7 15

1 dan 7 ga boradigan 6 ta qisqa yoʻl : 1 2 5 6 7

1 7 22

1 dan 7 ga boradigan 6 ta qisqa yoʻl : 1 3 2 4 7

2 6 20

2 dan 6 ga boradigan 6 ta qisqa yoʻl : 2 5 6

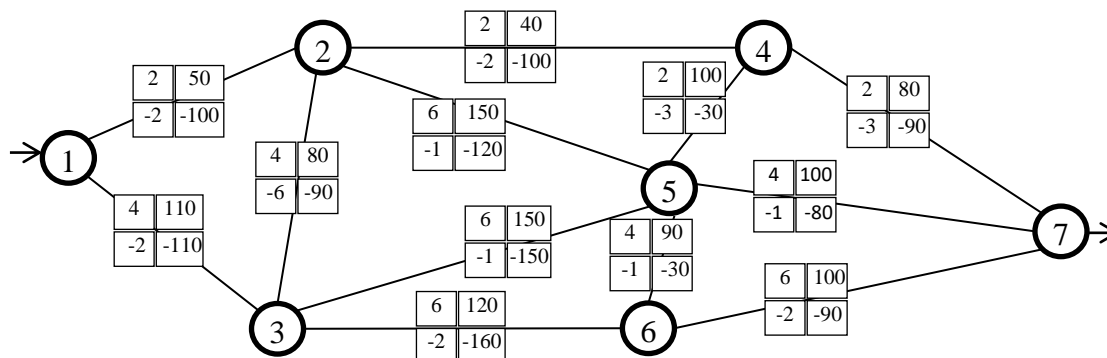
3 6 7

3 dan 6 ga boradigan 6 ta qisqa yoʻl : 3 6

Natijada: 1 – tugundan 7 – tugunga boradigan eng yaxshi marshrut aniqlanadi. Marshrut aniqlanishida qism yoʻllar bandligi inobatga olinadi. 1 – tugundan 7 – tugunga boradigan marshrutlar bir nechta marta takroriy soʻralsa L ta zaxira marshrutlar jadvalidan ham foydalanadi.



Graf bog‘lanish chizig‘ining ustki qismida musbat va pastki qismida manfiy parametrlar ifodalanadi (1 - rasm).



1 – rasm. Algoritmni testlash uchun graf tuzilmasi

MUHOKAMA

Axborot oqimi turli ko‘rinishdagi tashuvchilarni birlashtirishi mumkin. Masalan, qog‘oz va elektron ko‘rinishdagi tashuvchilar. Bu tashuvchilar bir-birlarini takrorlashi yoki bir-birlarini to‘ldirishlari mumkin. Axborot oqimi tuzilmasi uning bir jinsli yoki bir jinsli emasligini aniqlaydi. Bir jinsli axborot oqimi tashuvchini turi va vazifasini yagonaligi orqali tavsiflanadi. Axborot oqimi davriyligi bo‘yicha ikkitaga bo‘linadi.

Axborotlarni yig‘ishda va qayta ishlashda bevosita tizim va undagi dasturiy vositalarini foydalaniladi. Bunda, turli toifadagi foydalanuvchilar turli shakldagi axborotlarni tizimdan olishlari va yangi turdagi axborotlarni uzatishlari mumkin. Natijada elektron shakldagi axborotga bo‘lgan talab ortib boradi. Elektron shakldagi axborotlarni tizimdan olib foydalanish oson hamda axborotlarni tezkor olish va jo‘natish imkoniyatlari mavjud. Bunday qulaylik axborotlar almashish jarayonida tizimda axborot oqimlarini vujudga keltiradi. Tizimdan foydalanuvchilarni soni ko‘payib ketganda axborot oqimlarini ko‘payib ketishi kuzatiladi va bu tizimning turg‘unligini buzilishiga olib keladi. Natijada axborot saqlovchi va qayta ishlovchi vositalar tirbandligi va ba‘zi axborotlar yo‘qolib ketishi kuzatiladi. Bunday jarayonlarni sodir bo‘lishi tashkilotlarni ish faoliyatiga salbiy ta‘sir ko‘rsatib bir qator muammolarni vujudga keltirishi mumkin. Muammoni bartaraf qilish uchun axborot saqlovchi vositalarni texnik imkoniyatlarini oshirish bilan yechib bo‘lmaydi. Balki tizimlarda ishlatiladigan dasturlarga yangi algoritm kiritish yoki mavjudlarini mukammallashtirish orqali erishiladi.

XULOSA

Axborot oqimlarini ma'lumotlarni intellektual tahlil algoritmlaridan foydalanish ko'p parametrliligi grafda resurs cheklangan holatlarda marshrutizatsiya masalasini yechishning effektiv algoritmi va axborot oqimlarini boshqarish masalasini yechishning bir qancha samarali algoritmlari, modellari va ular asosida yaratilgan dasturiy ta'minotlar orqali ifodalandi. Maqolada yaratilgan algoritmlar, grafning bog'lanish parametrlari ko'p va ularning ba'zilari vaqt davomida o'zgarishini hisobga olgan holda marshrutizatsiya masalasini yechishning mukammal yechim berishi asoslandi. Mukammal yechimni aniqlashda dinamik tuzilmalardan foydalanildi. Bunda, turli toifadagi foydalanuvchilar turli shakldagi axborotlarni tizimdan olishlari va yangi turdagi axborotlarni uzatishlari mumkin. Natijada elektron shakldagi axborotga bo'lgan talab ortib boradi. Elektron shakldagi axborotlarni tizimdan olib foydalanish oson hamda axborotlarni tezkor olish va jo'natish imkoniyatlari mavjud. Bunday qulaylik axborotlar almashish jarayonida tizimda axborot oqimlarini vujudga keltiradi. Tizimdan foydalanuvchilarni soni ko'payib ketganda axborot oqimlarini ko'payib ketishi kuzatiladi va bu tizimning turg'unligini buzilishiga olib keladi. Natijada axborot saqlovchi va qayta ishlovchi vositalar tirbandligi va ba'zi axborotlar yo'qolib ketishi kuzatiladi.

REFERENCES

1. Karimov.I.A. "Zamonaviy axborot-kommunikatsiya texnologiyalarni yanada joriy etish va rivojlantirish bo'yicha chora-tadbirlari to'g'risida"gi PQ-1730-son , Toshkent, 2012y.
2. Kudryashov, S. V. Optimal routing of data flows in wireless sensor networks, 2008.
3. Кузовков, Н. Т. Инерциальная навигация и оптимальная фильтрация / Н. Т. Кузовков, О. С. Салычев. М. : Машиностроение, 1982.
4. GPS NavStar. Global Positioning System Standard Positioning Service Signal Specification. 1995.
5. Khru' S.A., Son'kin D.M. Adaptive algorithm processing flow of navigation data on basis of diagnostic filtering, 2012.
6. NMEA data. Gpsinformation.org: website. Available at:
7. <https://habrahabr.ru/> - Axborot texnologiyalariga oid maqolalar berib boriluvchi axborot tizimi.
8. <https://stackoverflow.com/> - Eng kata dasturlashni o'rganish uchun onlayn bilimlar jamlanmasi.
9. <https://www.w3schools.com/> - Web dasturlashni o'rgatuvchi tizim.