VILNIAUS UNIVERSITETAS

MATEMATIKOS IR INFORMATIKOS FAKULTETAS

MATEMATIKOS METODIKOS KATEDRA

Jelena Izmer

TINKLINIO PLANAVIMO UŽDAVINIAI

Magistro darbas

Vadovas

Doc. Antanas Apynis

VILNIUS 2006

TURINYS

ĮVADAS 3
1. Tinklinis planavimas 5
    1.1 Tinklinio planavimo istorija. 5
    1.2. Tinklinio planavimo esmė ir paskirtis 7
2. Pagrindinės tinklinio planavimo sąvokos 9
    2.1. Grafų teorijos sąvokos 9
    2.2. Pagrindiniai tinklinio planavimo elementai 10
        2.2.1. Kritinis kelias ir kritinis laikas 12
        2.2.2. Laiko rezervai 16
        2.2.3. Kritinio kelio nustatymo būdas. Jo pagrindimas 18
    2.3. Tinklinio grafiko vaizdavimas kompiuterio pagalba 20
        2.3.1. Anksčiausios įvykio pabaigos vaizdavimas kompiuterio pagalba 20
        2.3.2. Vėlyviausios įvykio pabaigos vaizdavimas kompiuterio pagalba 21
3. Tinklinio planavimo optimizavimo uždaviniai 31
    3.1 Projekto realizavimo kainos minimizavimas 31
IŠVADOS 38
SUMMARY 39
LITERATŪROS SĄRAŠAS 40

ĮVADAS

Per paskutinius dešimtmečius matematika labai pasikeitė. Sukūrus kompiuterius ir suradus naujas matematikos taikymo sritis (ekonomika, biologija ir kt.), iškilo nauji matematikos klausimai. Kai kurios matematikos šakos, kuriomis anksčiau domėjosi tik siauras specialistų ratas, tapo labai aktualios ir populiarios.

Organizuojant darbų projektus, vis dažniau naudojamas tinklinio planavimo metodas. Jis ypač naudingas, planuojant sudėtingą darbų kompleksą. *Tinklinio planavimo metodas* (Program Evolution and Review Technique) kartais dar vadinamas *kritinio kelio metodu (*Critical Path Method –CRM). Jis šiek tiek skiriasi nuo PERT metodo, kuris taikomas siekiant taupyti ir planuoti laiką. Tuo tarpu kritinio kelio metodas dažniausiai taikomas siekiant racionaliai panaudoti komplekso darbams skirtas lėšas.

Šio magistrinio darbo tikslai tokie:

a) susipažinti su pagrindinėmis tinklinio planavimo sąvokomis;
b) rasti būdą, kuris padėtų analizuoti tinklinį grafiką;
c) sudaryti tinklinio grafiko parametrų skaičiavimo algoritmus;
d) išnagrinėti tinklinius grafikus, naudojant kompiuterinę programą;
e) konkrečių pavyzdžių pagalba parodyti programos efektyvumą, skaičiuojant pagrindines tinklinio grafiko charakteristikas.

Darbo tikslams pasiekti, sprendžiami šie uždaviniai:

- išdėstoma tinklinio planavimo istorija;
- nurodoma tinklinio planavimo esmė bei paskirtis;
- aprašomi pagrindiniai tinklinio planavimo elementai;
- pateikiamas kritinio kelio radimo vienas būdas;
- pasiūlomi būdai, kaip galima vaizduoti grafiką kompiuterio pagalba;
- nagrinėjami tinklinio planavimo uždaviniai.

Darbą sudaro trys skyriai.

Pirmame skyriuje išdėstoma tinklinio planavimo istorija, jo esmė ir paskirtis. Aprašoma, kas sukūrė ir pritaikė tinklinio planavimo metodą, kaip laikui bėgant jis vystėsi.

Antrame skyriuje apibrėžiamos pagrindinės grafų teorijos ir tinklinio planavimo sąvokos: grafas, ciklas, tinklas, kritinis kelias ir t.t. Jos reikalingos tinkliniam grafikui sudaryti bei darbų nuoseklumui parodyti. Toliau pateikiami keli tinklinio grafiko parametrų skaičiavimo algoritmai. Šie parametrai naudojami projekto kritiniam laikui apskaičiuoti. Vienas iš algoritmų skirtas apskaičiuoti anksčiausią įvykių pabaigą, o kitas - vėlyviausią įvykių pabaigą. Aprašomas vienas būdas kritiniam keliui rasti. Naudojantis sudaryta programa, išsprendžiamas kritinio kelio radimo uždavinys.

Trečiame skyriuje nagrinėjami tinklinio planavimo optimizavimo modeliai. Naudojantis kompiuterinėmis programomis, išsprendžiamas viso darbų komplekso kainos minimizavimo uždavinys.

Darbe yra nagrinėjami tinklinio planavimo uždaviniai. Šiuose uždaviniuose aprašomi statybos ir rekonstravimo projektai, bei tokių projektų įgyvendinimo darbai.

1. Tinklinis planavimas

Tinkliniu planavimu vadinamas susijusių darbų komplekso įgyvendinimo tyrimo metodas.

Darbų kompleksu gali būtų vadinamas bet kokio sudėtingo projekto (pastato, lėktuvo, laivo ir panašiai), arba proceso skirto kokiam nors projektui įgyvendinti, planavimas. Darbų kompleksas tai užduotis, kurios įgyvendinimui reikia atlikti pakankamai daug įvairių darbų. Tokių užduočių, kur galima taikyti tinklinio planavimo metodus, gali būti bet kur. Tai gali būti darbai prie kurių dirba tiktai vienas žmogus, ir projektai prie kurių dirba daug žmonių ir organizacijų. Tokių didelių projektų planų sudarymui ir buvo sukurti tinklinio planavimo metodai. Taikant šį metodą, darbų kompleksas vaizduojamas schema, vadinama *tinkliniu grafiku.*

Visuma veiksmų (operacijų), reikalingų pasiekti tam tikrą rezultatą vadinama *projektu.* Šitie veiksmai (operacijos) vykdomi nustatyta tvarka. Tada projektas gali būti pateiktas kaip *tinklas* (kitaip sakant, *tinklinis grafikas* arba *grafas*). Tinklas akivaizdžiai parodo loginį sąryšį tarp darbų.

1.1. Tinklinio planavimo istorija

Tinklinio planavimo metodikos buvo pradėtos nagrinėti JAV. 1956 metais M. Uolkeris (M.R.Walker) iš firmos „E.I.Du Pont de Nemours & Co“ kartu su D. Kellis (J.E.Kelly) iš statybų planavimo grupės firmos „Remington Rand“ ieškojo naujus būdus, kurių pagalba būtų galima supaprastinti firmos „E.I.Du Pont de Nemours & Co“ gamyklų modernizacijos darbų planavimą. Tam tikslui pasiekti, jie panaudojo kompiuterį. Galiausiai buvo sukurtas paprastas ir racionalus *projekto aprašymo* metodas. Iš pradžių jis buvo pavadintas Uolkerio-Kellio (Walker-Kelly) metodu, o vėliau – *kritinio kelio metodu* arba KKM (Critical Path Method – CPM).

Tuo pačiu metu JAV karinėse jūros pajėgose buvo sukurtas programų analizės ir įvertinimo metodas PERT (Program Evaluation and Review Technique). Šį metodą sukūrė korporacija „Lockheed“ ir konsultavimo firma „Bozz-Allen and Hamilton“, įgyvendinant balistinių raketų „Poliaris“ kūrimo programą. Projekto apimtis buvo apie 60 tūkstančių operacijų. Jame bendradarbiavo apie 3800 pagrindinių rangovų. Panaudodami PERT metodą programų vadovai galėjo tiksliai nustatyti, kokie veiksmai turi būti atlikti tam tikru laiko momentu, ir kas turėtų juos atlikti. Taip pat tapo įmanoma sužinoti tikimybę, kada tam tikras veiksmas bus užbaigtas laiku. Programos vykdymas buvo itin sėkmingas, tad projektą pavyko užbaigti dviem metais anksčiau, nei buvo planuota. Tokios sėkmingos pradžios dėka šis valdymo metodas ilgainiui pradėtas naudoti visų JAV karinių pajėgų projektų planavimui.

Netrukus panašias valdymo metodikas naujos produkcijos kūrimui bei gamybos modernizavimui pradėjo taikyti ir didelės pramoninės korporacijos. Darbų planavimo metodas plačiai paplito statybose, realizuojant įvairius projektus. Vienas iš valdymo projekto pavyzdžių yra hidroelektrinės statybos Čerčilio upėje Njufaundlende (Labradoro pusiasalis). Projekto užsakovu buvo korporacija „Churchill Falls Labrador Corp.“, kuri projekto ir statybos įgyvendinimui įdarbino firmą „Acress Canadian Betchel“. Šio projekto vertė apėmė apie 950 mln. dolerių. Hidroelektrinė buvo statoma nuo 1967 iki 1976 metų. Projektas turėjo daugiau nei 100 statybinių sutarčių. Be to, kai kurių sutarčių vertė siekė 76 mln. dolerių. 1974 metais projekto darbų eiga jau lenkė numatytą grafiką 18 mėnesių, neviršijant planuotų sąnaudų įverčių.

Apibendrinus, galimybė sutaupyti projektų įgyvendinimo laiką atsirado todėl, kad, tobulėjant skaičiavimo technikai, didelių ir sudėtingų kompleksinių darbų valdyme buvo pradėti taikyti tikslūs matematiniai metodai. Visgi, pirmieji kompiuteriai buvo labai brangūs ir prieinami tik stambioms organizacijoms.

Iš pradžių stambios kompanijos kurdavo programinį aprūpinimą tik savo projektų įgyvendinimui. Netrukus pirmos projektų valdymo sistemos pasirodė ir programinės įrangos rinkoje. Šios klasės valdymo sistemų pagrindiniai rodikliai buvo jų didelė galia bei galimybė pakankamai tiksliai aprašyti projektus, naudojant sudėtingus tinklinio planavimo metodus. Tokios sistemos buvo orientuotos į kvalifikuotus vadybininkus, vadovaujančius didelių projektų kūrimui, gerai išmanančius tinklinio planavimo algoritmus ir specifinę terminologiją. Paprastai projektus kurdavo ir konsultuodavo dėl iškilusių valdymo klausimų specialios konsultacinės firmos.

Audringas projektų valdymo sistemų vystymosi etapas prasidėjo, atsiradus personaliniams kompiuteriams, tai yra, kai kompiuteris tapo darbo įrankiu plačiam vadovų ratui. Augant valdymo sistemų vartotojų skaičiui, atsirado galimybė sukurti naujo tipo projektų valdymo sistemas. Vienas iš svarbiausių tokių sistemų rodiklių buvo jų naudojimo paprastumas. Naujos kartos projektų valdymo sistemos buvo rengiamos taip, kad būtų suprantamos kiekvienam vadybininkui, nereikalautų specialaus paruošimo. Naujų šios klasės sistemų versijų kūrėjai stengėsi, išlaikant žemas kainas, išsaugoti išorinį sistemų paprastumą ir padidinti jų funkcionalumą bei pajėgumą. Tokiu būdu sistemos tapo prieinamos bet kokio lygio firmoms.

XX amžiaus paskutiniame dešimtmetyje planuojamų projektų pagrindinę dalį sudarė nedidelio mąsto projektai. Pavyzdžiui, Info World savaitraščio apklausos rezultatai parodė, kad 50% JAV vartotojų reikalingos sistemos, kurios leistų įgyvendinti planus, apimančius 500 – 1000 darbų. Tik 28% vartotojų buvo reikalingos sistemos, leidžiančios įgyvendinti daugiau negu 1000 darbų apimties planus. Kalbant apie projektų įgyvendinimo resursus, 38% vartotojų turėjo valdyti 50 – 100 resursų rūšių. Tik 28% vartotojų teko kontroliuoti daugiau nei 100 resursų rūšių. Šio tyrimo metu, atsižvelgiant į darbų ir resursų paskirstymą, buvo išskirtos trys projektų kategorijos:

1. mažieji projektai – iki 81 darbo ir 14 resursų rūšių;
2. vidutiniai projektai – iki 417 darbų ir 47 resursų rūšių;
3. stambieji projektai – iki 1 198 darbų ir daugiau ir 165 resursų rūšių.

Taigi, projektų valdymo sistemos naudojimas praktikoje gali būti efektyvus įgyvendinant ir labai mažus projektus.

Didėjant projektų valdymo sistemų vartotojų ratui, auga poreikis analizuoti tinklinio planavimo metodus, ir jų taikymo būdus. Vakarų moksliniuose leidiniuose nuolat skelbiami straipsniai apie projektų valdymų sistemas. Šiuose straipsniuose analizuojamas tinklinio planavimo metodo taikymas, sprendžiant įvairių sferų valdymo uždavinius, pateikiami patarimai tokių sistemų vartotojams.

1.2. Tinklinio planavimo esmė bei paskirtis

Kuo sudėtingesnis ir didesnis planuojamas darbas arba projektas, tuo sudėtingesni planavimo, valdymo ir kontrolės uždaviniai. Tokiomis sąlygomis kalendorinio grafiko naudojimas ne visada tinkamas, kadangi sunku pagrįstai ir operatyviai planuoti, parinkti optimalų darbų vykdymo trukmės variantą, naudoti rezervus ir eigoje koreguoti grafiką. Šių tiesinio kalendorinio grafiko trūkumų galima lengvai išvengti, naudojant tinklinio planavimo metodus. Tokiu būdu užtikrinamas kruopščiai apgalvotas darbų organizavimas, sukuriamos sąlygos efektyviam vadovavimui.

Visas procesas vaizduojamas grafiniame modelyje. Šis modelis vadinamas tinkliniu grafiku. Tinkliniame grafike atsižvelgiama į visus darbus, pradedant projektavimu ir baigiant įdiegimu. Taip pat jame nusakomi esminiai darbai, lemiantys projekto užbaigimo terminą.

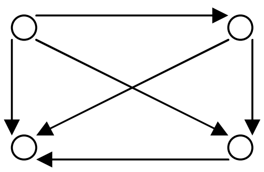
Įgyvendinant projektą atsiranda galimybė koreguoti planą, užtikrinti operatyvinio planavimo tolydumą. Egzistuojantys tinklinio grafiko analizės metodai leidžia įvertinti padarytų pakeitimų įtaką projekto įgyvendinimui, prognozuoti ateities darbus. Tinklinis grafikas tiksliai nurodo darbus, nuo kurių priklauso programos įvykdymo terminas.

2. Pagrindinės tinklinio planavimo sąvokos

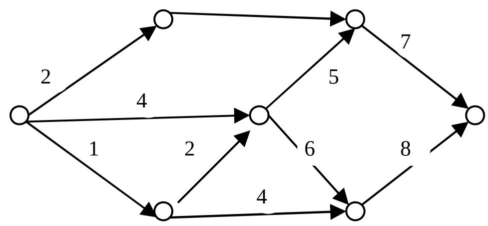
Praktikoje dažnai naudojamasi schemomis, sudarytomis iš taškų ir juos jungiančių linijų. Pavyzdžiui,sudarant statybų projektą, gyvenvietės vaizduojamos taškais, o jas jungiantys keliai – linijomis. Sudarant, susijusių darbų komplekso įgyvendinimo projektą, taip pat patogu naudotis tokia schema. Matematikoje ji vadinama grafu.

2.1. Grafų teorijos sąvokas

- *Grafu* yra vadinama taškų (viršūnių) $P = \{P_0, P_1, ..., P_n\}$ ir orientuotų briaunų $\{(P_i, P_j)\}$ aibė (1 pav.).
- Kreivė, jungianti dvi grafo viršūnes, vadinama *grafo briauna*. Kiekviena grafo briauna yra sutvarkyta viršūnių pora (P_i, P_j); P_i – briaunos pradžia, P_j – briaunos pabaiga.
- Grafo briaunų seka $\{(P_1, P_2), (P_2, P_3), ..., (P_{k-1}, P_k), (P_k, P_{k+1})\}$, kurioje kiekvienos briaunos pabaiga sutampa su kitos briaunos pradžia, yra vadinama *keliu* iš P_1 į P_{k+1}. Jis užrašomas taip: $(P_1, P_2, ..., P_{k+1})$.
- Kelias, kuris prasideda ir baigiasi toje pačioje viršūnėje, vadinamas *ciklu.* Vienos briaunos ciklas vadinamas *kilpa.*
- Grafas, kurio elementams (dažniausiai) priskiriami skaičiai, vadinamas *tinklu* (2 pav.).



1 pav.



2 pav.

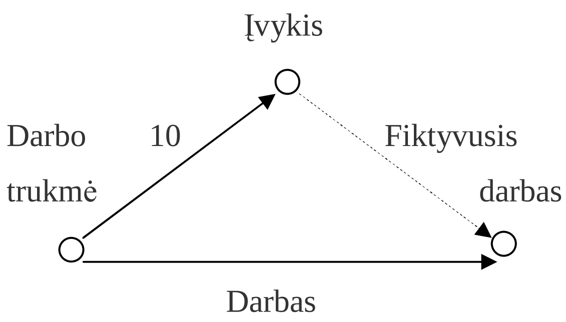
Tinklas sudarytas iš daugybės viršūnių ir briaunų, jungiančių įvairias viršūnių poras. Kiekvienai briaunai paskirta tam tikra kryptis, kryptis nurodoma rodykle. Brėžinyje viršūnės vaizduojamos skritulėliais, briaunos – juos jungiančiomis rodyklėmis. Kiekvienai viršūnei priskiriamas skaičius. Briauna, jungianti viršūnę P_i su viršūnę P_j, žymima simboliu (P_i , P_j).

Tinkliniu grafiku vadinamas modelis, kurio pagrindą sudaro tinklas. Tinklinio grafiko sudarymas yra pirmasis etapas, kuriant projektą. Tai yra labai svarbi dalis, reikalaujanti daug laiko ir pastangų. Žemiau pateikiami esminiai tinklinio grafiko elementai.

2.2. Pagrindiniai tinklinio grafiko elementai

- *Įvykis* – kokio nors darbo ar darbų pabaiga. Įvykiai reprezentuoja tarpinių uždavinių vykdymo rezultatus ir atvaizduoja projekto vykdymo skirtingus etapus.
- *Darbas* – laiko tarpas, reikalingas perėjimui nuo vieno įvykio prie kito.
- *Fiktyvusis darbas* – tai darbus siejanti loginė priklausomybė, kai negalima pradėti vieno darbo, kol nebus baigtas kitas. Fiktyviam darbui nereikia nei laiko, nei resursų.
- *Trukmė* – darbo ilgis. Fiktyviojo darbo trukmė lygi 0.

Tinkliniuose grafikuose įvykiai vaizduojami skritulėliais, o elementarieji darbai – rodyklėmis. Rodyklės pradžia reiškia darbo pradžią, o rodyklės pabaiga – darbo pabaigą (3 pav.). Įvykiai išreiškia baigtinį rezultatą.



3 pav. Tinklinio grafiko fragmentas

Įvykiai kaip ir grafo viršūnės žymimi didžiosiomis lotynų raidėmis P_0, P_1, P_2, ...P_m, ... P_n, kur P_0 – projekto pradžia, o P_n – pabaiga. Darbai žymimi kaip ir grafo briaunos (P_i, P_j), kur P_i – įvykio darbo pradžia, o P_j – įvykio darbo pabaiga. Darbo (P_i, P_j) trukmė t_{ij} užrašoma virš briaunos, vaizduojančios šį darbą. Visi tinkliniame grafike vaizduojami darbai yra elementarūs, tai yra darbai, kuriuos galima pradėti tik tada, kai visiškai užbaigti ankstesni darbai.

Taigi, *tinklinis grafikas* (*rodyklinė diagrama*, *tinklinis modelis* arba *loginis tinklas*) – tai modelis pavaizduotas grafo pavidalu, nurodantis technologinį sąryšį tarp darbų.

Egzistuoja daugybė tinklinio grafiko sudarymo taisyklių. Svarbiausios iš jų:

1. Bet kokiame tinkliniame grafike turi būti pradinis įvykis P_0, iš kurio darbai tik tai išeina, ir pabaigos įvykis P_k – į kurį darbai tik įeina.

2. Tinkliniame grafike negali būti *ciklų*, t.y. kelių, kurie prasideda ir baigiasi toje pačioje viršūnėje.

3. Jeigu įvykį P_i su įvykių P_j jungia daugiau negu vienas darbas, tai tinklinis grafikas papildomas naujais įvykiais, kurie su P_j sujungiami fiktyviais darbais (4 pav.).

4 pav.

4. Jeigu įvykiu P_i baigiasi ir prasideda daugiau kaip vienas darbas, ir ne visiems prasidedantiems įvykiu P_i darbams būtina užbaigti visus darbus, kurie baigiasi tuo įvykiu, tai tinklinis grafikas papildomas naujais įvykiais ir fiktyviais darbais (5 pav.).

5 pav.

5. Jeigu yra kelios projekto pradžios ir pabaigos, tai jos sujungiamos fiktyvais darbais (6 pav.).

6 pav.

6. Tinklinis grafikas turi būti teisingai sunumeruotas, tai yra reikia panaudoti numeravimo algoritmą arba briaunų išbraukimo metodą.

Turint tiktai tinklinio grafiko struktūrą, negalima išspręsti optimizavimo uždavinio. Reikia žinoti dar kelis reikalingus parametrus:

- kritinį kelią ir kritinį laiką;
- anksčiausias ir vėlyviausias įvykio pabaigas, tai yra T_{Aj} ir T_{Vj};
- darbų ir įvykių laiko rezervus.

2.2.1. Kritinis laikas ir kritinis kelias

Nagrinėjamas tinklinis grafikas, kurio briaunos (P_i, P_j) vaizduoja darbus, o viršūnės P_j, $j=0,1,...,n$, — darbų pradžios ar pabaigos įvykius. Kiekvienai briaunai (P_i, P_j) priskirtas skaičius $t_{ij} \geq 0$ — vaizduojamojo darbo trukmė. Skaičius t_{ij} vadinamas *briaunos* (P_i, P_j) *ilgiu*. Tada *kelio ilgiu* (P_l, P_{j_1}), (P_{j_1}, P_{j_2}), ..., (P_{j_k}, P_j) iš P_l į P_j yra vadinama visų į jį įeinančių briaunų ilgių sumą, tai yra skaičius $t_{lj_1} + t_{j_1 j_2} + ... + t_{j_k j}$.

Norint apskaičiuoti viso projekto realizavimo trukmę, reikia nustatyti kiekvieno įvykio anksčiausiąjį laiką. Sutarta, kad projekto pradžia (pradinio įvykio P_0 anksčiausias laikas) T_0 lygu nuliui. Tada anksčiausias įvykio P_j atlikimo laikas T_j yra visų darbų, įeinančių į P_j, laikas. *Anksčiausias įvykio P_j pabaigos laikas lygus maksimaliam kelio ilgiui iš P_0 į P_j.* Šis laikas žymimas T_{Aj}.

1 apibrėžimas. *Kritiniu projekto laiku vadinamas projekto anksčiausias paskutinio įvykio P_n pabaigos laikas T_{An}.* Kitaip tariant, *kritinis laikas – tai minimalus laiko kiekis, būtinas visų darbų komplekso atlikimui*.

Remiantis šiuo apibrėžimu, viso projekto realizavimo trukmė lygi kritiniam laikui $T_{n.}$, tai yra kritinis laikas lygus maksimaliam kelio ilgiui iš P_0 į P_n.

2 apibrėžimas. *Bet kuris kelias iš P_0 į P_n, kurio ilgis lygus T_n , vadinamas kritiniu keliu.*

3 apibrėžimas. *Anksčiausia įvykio pabaiga* vadinamas laiko momentas, kuris sutampa su ilgiausių į šį įvykį įeinančių darbų pabaiga, tai yra, jis negali įvykti anksčiau, nei užsibaigia visi į jį įeinantys darbai. Anksčiausia įvykio pabaiga žymima raide T_{Aj} , čia j – nagrinėjamo įvykio numeris.

Nustatant anksčiausias įvykių pabaigas einama iš kairės į dešinę, tai yra, nuo pradinio projekto įvykio P_0 iki paskutinio P_n. Skaičiuojant anksčiausiąją įvykių pabaigą, naudojamas pagrindinis algoritmas:

1. Paeiliui imami įvykiai P_j ir darbai (P_i, P_j). Įvykiui P_0 priskiriamas skaičius $T_{A0} = 0$, tai yra, sutarta anksčiausiai pradinio įvykio pabaigai priskirti nulį. Kitiems įvykiams P_j , $j = 1, ..., n$, skaičius

$$T_{Aj} = \max \{ T_{Ai} + t_{ij} \}, \quad \textbf{(2.1)}$$

čia T_{Aj} – anksčiausia nagrinėjamo įvykio pabaiga;

j – nagrinėjamo įvykio numeris;

i – prieš tai nagrinėtų įvykių numeriai;

T_{Ai} – anksčiausia prieš tai nagrinėto įvykio pabaiga;

t_{ij} – darbo trukmė, jungianti i - tą prieš tai nagrinėtą įvykį su dabar nagrinėjamu įvykiu.

2. Prie kiekvieno apskaičiuoto T_{Aj} užrašomas to įvykio P_i numeris i_1, i_2, ... i_v, , su kurio skaičius T_{Aj} tampa maksimalus;

3. Skaičius T_{Aj} lygus ilgiausio kelio iš P_0 į P_j ilgiui. Taigi, $T_{A0} = 0, j = 0, 1, ..., n$.

Taigi, skaičius $T_{An} = T_n$ yra projekto kritinis laikas.

4 apibrėžimas. *Vėlyviausia įvykio pabaiga* vadinamas laiko momentas, atitinkantis vėliausią galimą įvykio pabaigą, nepailginančią kritinio kelio trukmės, tai yra, tokią, kad galėtų būti laiku pradėti visi kiti su tuo įvykiu susiję darbai. Vėlyviausia įvykio pabaiga žymima T_{Vj} , čia j – nagrinėjamo įvykio numeris.

Nustatant vėlyviausias įvykių pabaigas einama iš dešinės į kairę, tai yra, nuo paskutiniojo P_n iki pradinio projekto įvykio P_0. Skaičiuojant vėlyviausią įvykių pabaigą naudojamas algoritmas:

1. Atvirkštine tvarka imami įvykiai P_i ir darbai (P_i, P_j). Skaičiuojant vėlyviausią laiką sutarta, kad vėlyviausia užbaigiančio projektą įvykio pabaiga sutampa su anksčiausia to įvykio pabaiga (tai yra $T_{Vn} = T_{An}$).

Įvykiui P_n priskiriamas skaičius $\mu_n = 0$. Kitiems įvykiams P_i , $i = n-1, ..., 0$, skaičius

$$\mu_i = max \{ \mu_j + t_{ij} \}, \quad \textbf{(2.2)}$$

čia μ_i – vėlyviausia nagrinėjamo įvykio pabaiga;

i – nagrinėjamo įvykio numeris;

j – kito nagrinėjamo įvykio numeris;

μ_j – j-to kito nagrinėjamo įvykio vėlyviausia pabaiga;

t_{ij} – darbo trukmė, jungianti j-tą einantį po nagrinėjamo įvykio su dabar nagrinėjamu įvykiu.

2. Prie kiekvieno apskaičiuoto μ_i užrašomas to įvykio P_i numeris $i_1, i_2, ... i_v$, , su kuriuo skaičius μ_i pasiekia maksimumą;

3. Skaičius μ_i lygus ilgiausio kelio iš P_i į P_n ilgiui. Taigi, skaičius $T_n = \mu_0$, $T_{Vj} = T_n - \mu_i$, $i = 0,1, ..., n$.

Remiantis šiais apibrėžimais ir algoritmais, nagrinėjamas praktinis uždavinys.

1 uždavinys. Vieną statybinį projektą (buto remontas) įgyvendina 6 darbininkai ir 1 darbo vadovas. Kiekvienam darbininkui paskirtas tam tikras darbas:

I darbininkas atlieka tris darbus: darbą (P_0, P_1), jo trukmė – 2 dienos; darbą (P_0, P_2), jo trukmė – $t_{02}=1$ ir darbą (P_0, P_3), $t_{03}=4$;

II darbininkas - (P_1, P_5), $t_{15}=4$;

III darbininkas - (P_2, P_3), $t_{23}=2$;

IV darbininkas - (P_3, P_4), $t_{34}=6$ ir (P_3, P_5), $t_{35}=5$;

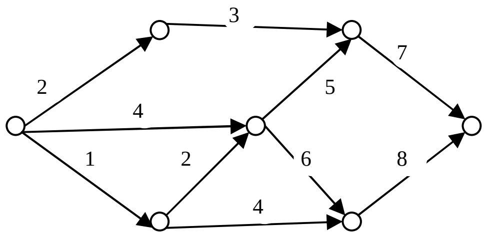
V darbininkas - (P_4, P_6), $t_{46}=8$;

VI darbininkas - (P_5, P_6), $t_{56}=7$.

Darbo vadovas priima visų darbininkų darbus ir analizuoja šitą projektą.

Šio projekto tinklinis grafikas, kuriame pažymėti įvykiai ir nurodytos darbų trukmės, pavaizduotas 7 paveiksle.

Reikia nustatyti, kada anksčiausiai ir vėliausiai kiekvienas darbininkas atliks savo darbus. Kiek dienų užtruks šis projektas?



7 pav. Statybos projekto tinklinis grafikas

Sprendimas

1. Reikia apskaičiuoti anksčiausias įvykių P_J pabaigas. Naudojamasi pirmuoju algoritmu.

Tinklas peržiūrimas, pradedant nuo P_0. Naudojant formulę T_{Aj} = max { T_{Ai} + t_{ij} }, apskaičiuojami visi T_{Aj}. Skliaustuose kartu su T_{Aj} žymimi tų įvykių P_i, su kuriais T_{Aj} pasiekia maksimumą, numeriai:

$T_{A0} = 0$ - (0) – įvykio max

$T_{A1} = T_{A0} + t_{01} = 2$ - (0) – įvykio max

$T_{A2} = T_{A0} + t_{02} = 1$ - (0) – įvykio max

$T_{A3} = max\{ T_{A0} + t_{03}, T_{A2} + t_{23} \}$ = max{4, 1+2} = 4 - (0) – įvykio max

$T_{A4} = max\{ T_{A3} + t_{34}, T_{A2} + t_{24} \}$ = max{4+6, 1+4} = 10 - (3) – įvykio max

$T_{A5} = max\{ T_{A1} + t_{15}, T_{A3} + t_{35} \}$ = max{2+3, 4+5} = 9 - (3) – įvykio max

$T_{A6} = max\{ T_{A5} + t_{56}, T_{A4} + t_{46} \}$ = max{9+7, 10+8} = 18 - (4) – įvykio max

Reiškia, projekto kritinis kelias T_{A6} lygus 18.

2. Tinkliniam grafikui (7 pav.) reikia apskaičiuoti vėlyviausias įvykių P_j pabaigas T_{Vj}, su sąlyga, kad $T_6 = T_{kr}$.
   - Daroma prielaida, kad $\mu_6 = 0$, ir apskaičiuojama μ_j, pradedant nuo tinklo galo $P_6, P_5, ..., P_0$:
   - $\mu_6 = 0$,

     $\mu_5 = \mu_6 + t_{56} = 7$,

     $\mu_4 = \mu_6 + t_{46} = 8$,

     $\mu_3 = max\{ \mu_4 + t_{34}, \mu_5 + t_{35} \} = max\{8+6, 7+5\} = 14$,

     $\mu_2 = max\{ \mu_3 + t_{23}, \mu_4 + t_{24} \} = max\{14+2, 8+4\} = 16$,

     $\mu_1 = \mu_5 + t_{15} = 7+3=10$,

$\mu_0 = max\{\mu_1 + t_{01}, \mu_2 + t_{02}, \mu_3 + t_{03}\} = max\{10+2, 16+1, 14+4\} = 18$

- Maksimalūs laikai T_{Vj} apskaičiuojami naudojant formulę: $T_{Vj} = T_n - \mu_i$

$T_{V6} = T_6 = 18$,

$T_{V5} = 18 – 7 = 11$,

$T_{V4} = 18 – 8 = 10$,

$T_{V3} = 18 – 14 = 4$,

$T_{V2} = 18 – 16 = 2$,

$T_{V1} = 18 – 10 = 8$,

$T_{V0} = 18 – 18 = 0$.

3. Taigi, galima atsakyti į klausimą, kada vadovas priims projektą: po 18 dienų, nes $T_{A6}=T_{V6} = T_6 = 18$.

2.2.2. Laiko rezervai

Žinant anksčiausias ir vėlyviausias įvykio pabaigas, galima apskaičiuoti įvairius rezervus.

Laiko tarpas $[T_{Aj}, T_{Vj}]$ vadinamas įvykio P_j *laisvės intervalu*, arba *rezerviniu intervalu*. *Įvykio rezervu* (arba atsarga*) vadinamas skirtumas tarp vėlyviausios ir anksčiausios įvykių pabaigų*. Jis žymimas $r_j = T_{Vj} - T_{Aj}$. Įvykio laiko rezervai parodo, kiek galima atidėti įvykio vykdymą, lyginant su j-ju anksčiausiuoju vykdymo laiku, kai viso projekto vykdymo laikas nepakito.

5 apibrėžimas. *Darbo* (P_i, P_j) *pilnuoju laiko rezervu vadinamas skaičius*

$$p_{ij} = T_{Vj} - T_{Ai} - t_{ij}\,. \tag{2.3}$$

Jis nusako, kiek darbą (P_i, P_j) galima užtęsti (vėliau pradėti), nedidinant kritinio laiko. Išnaudojus pilnąjį darbo (P_i, P_j) laiko rezervą, vėlesni darbai (P_j, P_k) pradedami tik vėliausiu vykdymo laiku.

6 apibrėžimas. *Darbo* (P_i, P_j) *laisvuoju laiko rezervu vadinamas skaičius*

$$l_{ij} = T_{Aj} - T_{Ai} - t_{ij}\,. \tag{2.4}$$

Jis nurodo, kiek darbą (P_i, P_j) galima užtęsti (vėliau pradėti), kad vėlesnių darbų pradžios laikas T_{Aj} nesikeistų. Naudojantis darbo (P_i, P_j) laisvuoju laiku rezervu, gali sumažėti ankstesniųjų darbų (P_k, P_i) pilnieji laiko rezervai, tačiau vėlesniųjų darbų pilnieji laiko rezervai nesikeičia.

7 apibrėžimas. *Darbo (P_i , P_j) nepriklausomuoju laiko rezervu vadinamas skaičius*

$$n_{ij} = max\{ 0, T_{Aj} - T_{Vi} - t_{ij} \}. \quad \textbf{(2.5)}$$

Išnaudojus šį rezervą, įvykis P_i įvyksta vėliausiuoju laiku T_{Vi} , o P_j – anksčiausiuoju laiku T_{Aj} . Jis nepriklauso nuo kitų darbų laiko rezervų panaudojimo.

2 uždavinys. Reikia apskaičiuoti 1 uždavinio statybos projekto (buto remonto) pilnus, laisvus ir nepriklausomus laiko rezervus.

Sprendimas

Kad būtų galima apskaičiuoti visus laiko rezervus, būtina pasinaudoti 1 uždaviniu. Duomenys iš 1 uždavinio:

T_{Ai} : 0, 0, 0, 2, 1, 1, 4, 4, 10, 9;

T_{Vi}: 0, 0, 0, 8, 2, 2, 4, 4, 10, 11;

T_{Aj} : 2, 1, 4, 9, 4, 10, 10, 9, 18, 18

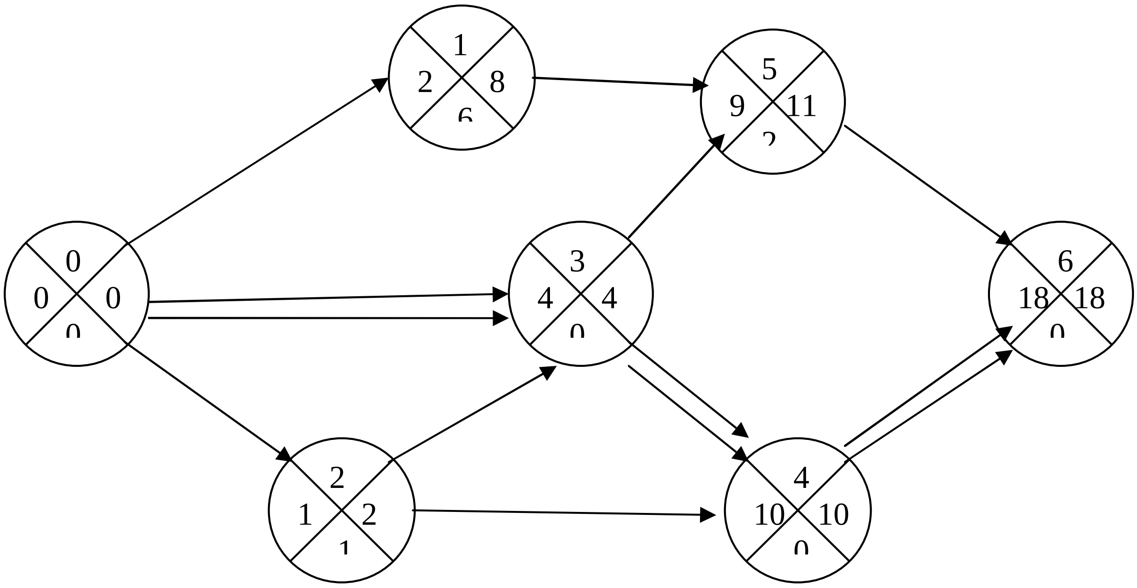
T_{Vj} : 8, 2, 4, 11, 4, 10, 10, 11, 18, 18.

Projekto laiko rezervai surašyti 1 lentelėje.

1 lentelė. Statybos projekto laiko rezervai

Darbas		Trukmė t_{ij}	Rezervas	Rezervas	Rezervas
Pradinis įvykis P_i	Galutinis įvykis P_j		Laisvas $T_{Aj} - T_{Ai} - t_{ij}$	Nepriklausomas $max\{0, T_{Aj} - T_{Vi} - t_{ij}\}$	Pilnas $T_{Vj} - T_{Ai} - t_{ij}$
P_0	P_1	2	0	0	6
P_0	P_2	1	0	0	1
P_0	P_3	4	0	0	0
P_1	P_5	3	4	0	6
P_2	P_3	2	1	0	1
P_2	P_4	4	5	4	5
P_3	P_4	6	0	0	0
P_3	P_5	5	0	0	2
P_4	P_6	8	0	0	0
P_5	P_6	7	2	0	2

Norint išvengti daugybės užrašų grafike, tikslinga kiekvieną įvykį vaizduojantį skrituliuką padalinti į keturias dalis. Skrituliuko viršutinėje dalyje užrašomas įvykio numeris, kairėje – anksčiausia įvykio pabaiga (T_{Aj}), dešinėje – vėlyviausia įvykio pabaiga (T_{Vj}), apačioje – įvykio rezervas (r_j). Žinant tokius parametrus, tinklinis grafikas įgauna kitą pavidalą (8 pav.).



8 pav. Tinklinio grafiko pavyzdys

2.2.3. Kritinio kelio nustatymo būdas. Jo pagrindimas

Tinkliniuose grafikuose kritinio kelio nustatymo būdas pagrįstas pilnais darbo rezervais. Pilnieji darbo rezervai parodo, kiek galima užtęsti darbą, arba padidinti darbo trukmę, nedidinant kritinio laiko.

Iš pilnųjų darbo rezervų sąvokos seka vienas teisingas teiginys.

Teiginys. *Pilnasis darbo rezervas gali atsirasti tiktai tada, kai egzistuoja ilgesnis kelias, nei kelias, į kurį įeina nagrinėjamas darbas.*

Šis teiginys įgauna prasmę, atsakant į klausimą, kodėl kai kuriems darbams gali būti atidėta darbo pradžia, arba galima padidinti jų trukmę, bet viso projekto vykdymo laikas nepasikeičia? Taip yra tiktai dėlto, kad šis projekto vykdymo laikas priklauso nuo kito ilgiausio kelio.

Pirmiausia, derėtų prisiminti kelias pagalbines teoremas.

1 teorema. *Tinklinio grafiko kelias yra kritinis tada ir tik tada, kai visų į jį įeinančių darbų pilnieji laiko rezervai lygus nuliui.*

Įrodymas

Būtinumas: jeigu kelias yra kritinis, tai visų į jį įeinančių darbų pilnieji laiko rezervai lygūs nuliui (tai yra $p_{ij=0}$).

Įrodymas pagrįstas prieštaros metodu.

Prielaida. Tegul, nagrinėjamas kelias yra kritinis, ir šiam keliui priklauso vienas darbas, turintis nenulinį laiko rezervą. Tai reiškia, kad egzistuoja kitas kelias, kurio trukmė yra didesnė negu nagrinėjamojo kelio. Todėl ir atsiranda šitas nenulinis laiko rezervas. Bet, jeigu egzistuoja kelias kurio trukmė didesnė, tai nagrinėjamas kelias negali būti kritinis, kadangi kritinis kelias yra ilgiausias kelias. Atsiranda prieštara, kad nagrinėjamas kritinis kelias, turintis darbą, kurio pilnasis laiko rezervas yra nelygus nuliui, nėra kritinis kelias.

Taigi, prielaida buvo klaidinga. Galima padaryti išvadą, jog egzistuoja kelias, kurio kiekvienas darbas turi nulinius pilnus laiko rezervus.

Pakankamumas: jeigu visi kelio darbai turi nulinius pilnus laiko rezervus, tai šitas kelias būtinai bus kritinis.

Jeigu kelias turi tik darbus, kurių pilnieji laiko rezervai lygus nuliui, tai reiškia, kad nei vieno šio kelio darbo negalima užtęsti (pradėti vėliau), nes užsitęstų ir visi likusieji darbai, tai yra paskutinis įvykis būtų atliktas nenumatytu laiku. Užtęsti darbą galima tada, kai nagrinėjamojo kelio visų darbų trukmių suma lygi paskutiniojo įvykio anksčiausiajam laikui (žiūrėkite 1, 2 apibrėžimus), tai yra, kritinio kelio ilgiui. Tada nagrinėjamas kelias ir bus kritinis, nes jis bus lygus kritinio kelio ilgiui.

Teorema įrodyta.

Kiekvienas tinklinis grafikas turi kritinį kelią. Tai reiškia, kad yra kelias su didžiausia trukme (arba pats ilgiausias kelias). Remiantis 1 teoremos įrodymu, galima padaryti išvadą: *kiekviename tinkliniame grafike galima rasti kelią, kurio darbai turės tiktai nulinius pilnus laiko rezervus.*

2 teorema. *Jeigu į tinklinio grafiko įvykį įeina darbas, turintis nulinį pilnąjį rezervą, tai tarp visų išeinančių iš šio įvykio darbų atsiras nors vienas darbas, turintis nulinį laiko rezervą. Tai yra, darbai turintys nulinius laiko rezervus, eina tolygiai vienas paskui kitą.*

Šita teorema įrodinėjama prieštaros metodu.

Išanalizavus šitas teoremas, galima užrašyti kritinio kelio nustatymo vieną būdą.

Kritinio kelio nustatymo būdas

- tinklinis grafikas peržiūrimas nuo pradinio įvykio iki paskutinio;
- nagrinėjant tinklinio grafiko pradinį įvykį, darbu esančiu kritiniame kelyje, išrenkamas tas darbas, kuris turi nulinį pilnajį laiko rezervą. Remiantis 1 teorema, toks darbas turi egzistuoti;
- nagrinėjant darbus, išeinančius iš įvykio, prie kurio atvedė darbas su nuliniu pilnuoju laiko rezervu, išrenkamas darbas, turintis nulinį pilnąjį laiko rezervą. Remiantis 2 teorema, toks darbas turi egzistuoti;
- jeigu yra keli darbai, turintys nulinius pilnuosius laiko rezervus, išeinantys iš tam tikro įvykio, tai pasirenkamas bet koks darbas. Remiantis 2 teorema, kritinio kelio sudarymo procesas negali atsidurti aklavietėje, ir anksčiau ar vėliau bus pasiektas paskutinis tinklinio grafiko įvykis.

Laikantis šių taisyklių galima atrasti kelią, turintį tik tai nulinius pilnuosius laiko rezervus. Tada, remiantis 1 teorema, tas kelias ir bus kritinis.

2.3. Tinklinio grafiko vaizdavimas kompiuterio pagalba

Dirbant kompiuteriu, būtina visas abstrakčias sąvokas ir reiškinius, kurie yra aiškūs kiekvienam žmogui, pertvarkyti kompiuteriui suprantama forma. Tinklinis grafikas, kaip nuoseklus skrituliukų ir rodyklių grafinis vaizdavimas, kompiuteriui nieko nereiškia. Tam, kad kompiuteris galėtų suprasti tinklinio grafiko struktūrą, ir svarbiausia, jį apdoroti, būtina pavaizduoti šią struktūrą kompiuteriui pažįstama kalba.

2.3.1. Anksčiausios įvykio pabaigos vaizdavimas kompiuterio pagalba

Išnagrinėjus vieną teoremą, galima parašyti algoritmą. Jo pagalba galima apskaičiuoti anksčiausias įvykių pabaigas.

3 teorema. *Jeigu tinklinio grafiko įvykiai sunumeruoti taip, kad bet koks jo darbas išeina iš įvykio su mažesniu numeriu ir įeina į įvykį su didesniu numeriu, tai anksčiausios* *įvykių pabaigos (T_{Aj}) skaičiuojamos didėjimo tvarka (0-nio įvykio, 1-ojo įvykio, ir taip toliau) iki paskutiniojo įvykio. Tokiu atveju atsidurti aklavietėje neįmanoma, su sąlyga, kad skaičiuojant anksčiausią nagrinėjamo įvykio pabaigą, iš karto skaičiuojamos ir visų kitų iš šio įvykio išeinančių darbų ankstyvųjų pabaigų pradžios (T_{Ai}).*

Šią teoremą galima įrodyti matematinės indukcijos metodu.

Pasinaudojus šia teorema ir anksčiausio įvykio pabaigos algoritmu (iš 2.2.2.skyrelio) galima sudaryti vieno iš tinklinio grafiko parametrų skaičiavimo algoritmą. Šis algoritmas parašytas bet kuriai programavimo kalbai.

Algoritmas

```
Begin
T_A0= 0;
for n ⊂ [1; |P|-1] // |P| - viršūnių skaičius
        T_An= 0;
        for k ⊂ [0; |D|-1] // |D| - darbų skaičius
                i:= D_k pradžios indeksas
                j:= D_k pabaigos indeksas
                t:= D_k vertė
                if n=j
                        T_An = max(t + T_Aj, T_An)
End.
```

2.3.2. Vėlyviausios įvykio pabaigos vaizdavimas kompiuterio pagalba

Toliau pateikta teorema leidžia parašyti algoritmą, su kuriuo įmanoma apskaičiuoti vėliausias įvykių pabaigas.

4 teorema. *Jeigu tinklinio grafiko įvykiai sunumeruoti taip, kad bet koks jo darbas išeina iš įvykio su mažesniu numeriu ir įeina į įvykį su didesniu numeriu, tai vėlyviausių įvykių pabaigos skaičiuojamos įvykių numerių mažėjimo tvarka (paskutinio įvykio, prieš paskutinio įvykio, ...) iki pradinio įvykio. Tokiu būdu atsidurti aklavietėje neįmanoma, su sąlyga, kad* *skaičiuojant vėlyviausią nagrinėjamo įvykio pabaigą, iš karto skaičiuojamos ir visų kitų į šį įvykį įeinančių darbų vėlyviausios pradžios.*

Šią teoremą galima įrodyti matematinės indukcijos metodu.

Naudojantis šia teorema ir vėlyviausio įvykio pabaigos algoritmu (iš 2.2.2 skyrelio), sudaromas tinklinio grafiko vėlyviausios įvykio pabaigos skaičiavimo algoritmą.

Algoritmas

Begin

$\mu_{|P|-1} = 0;$

for $n \subset [|P|-2; 0]$ // $|P|$ - viršūnių skaičius

$\mu_{V_n} = 0;$

for $k \subset [0; |D|-1]$ // $|D|$ - darbų skaičius

$i := D_k$ pradžios indeksas

$j := D_k$ pabaigos indeksas

$t := D_k$ vertė

if $n=i$

$\mu_n = \max(t + \mu_i, \mu_n)$

$T_{V_n} = T_{A|P|-1} - \mu_n$

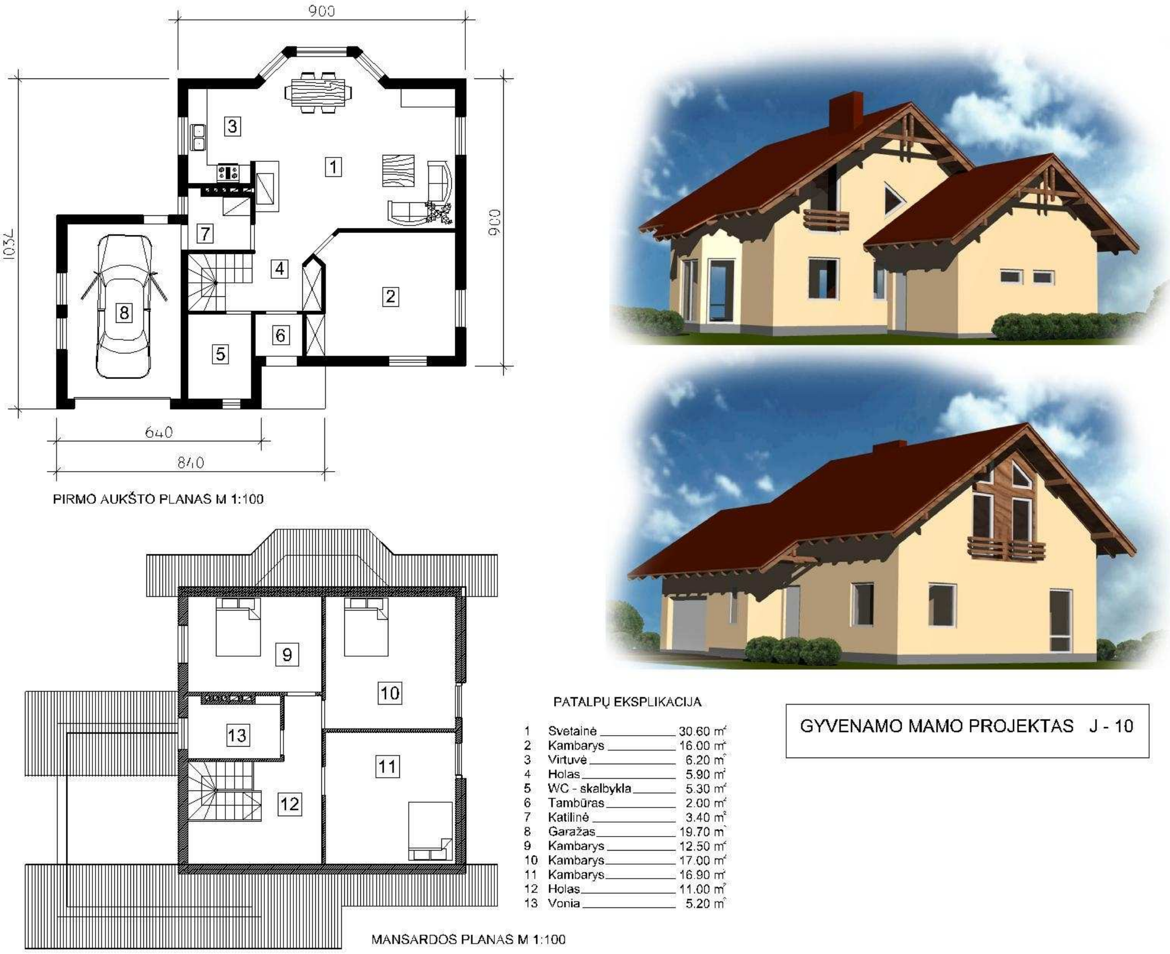
$T_{V|P|-1} = T_{A|P|-1}$

End.

Naudojantis darbe pateiktais algoritmais, sudaryta tinklinio grafiko sudarymo programa. Ji apžvelgiama 3 uždavinyje.

3 uždavinys. Viena sutuoktinių pora nusprendė likusį gyvenimą praleisti ne mieste, o kur nors užmiestyje. Paskaičiavus savo santaupas ir pasikonsultavus su nekilnojamojo turto agentūromis, sutuoktiniai nusprendė, kad jie gali sau leisti pastatyti nedidelį namą užmiestyje. Jie peržiūrėjo skelbimus, surado statybos firmą ir kreipėsi į ją dėl namo statybos plano sudarymo. Ši pora turėjo vieną prašymą – kad namas nebūtų labai didelis, būtų pastatytas kaip įmanoma greičiau.

Atsižvelgdamas į klientų pageidavimus, firmos projektuotojas sudarė gyvenamojo namo projektą (9 pav.), statybos darbų atlikimo planą su sąmatomis.



9 pav. Namo projektas

Sutuoktinių pora peržiūrėjo visus planus su nurodytais skaičiavimais, ir sutiko pasamdyti šitą firmą.

Statybos firma perdavė šitą užsakymą vienam iš projekto vadovų (vadovas žymimas V), kad jis suplanuotų darbo eigą ir pradėtų vykdyti. Šio namo statybai buvo atrinktos 11 darbuotojų brigadų (jos žymimos B1 – B11), ir 5 pagalbiniai darbuotojai (jie žymimi P). Pagalbiniai darbuotojai padėdavo atlikti kai kuriuos statybos darbus. Darbų paskyrimas:

- pirmoji brigada (B1) turėjo atlikti visus darbus, susijusius su namo pamatų ir namo perdangų statymu;
- antroji brigada (B2) – sienų mūrijimo darbus;
- trečioji brigada (B3) – stogo konstrukcijos darbus;
- ketvirtoji brigada (B4) – viso namo apšiltinimo ir namo apdailos darbus;

- penktoji brigada (B5) – santechnikos darbus;
- šeštoji brigada (B6) – su elektros instaliacijos darbus;
- septintoji brigada (B7) – langų montavimo darbus;
- aštuntoji brigada (B8) – durų montavimo darbus;
- devintoji brigada (B9) – namo vidinės apdailos darbus;
- dešimtoji brigada (B10) – namo tvarkymo darbus;
- pagalbiniai darbuotojai naktimis saugojo statybas, tai yra, atlikdavo apsaugos darbus.

Šito projekto darbų atlikimo planas pavaizduotas 2 lentelėje. Šioje lentelėje yra surašyti visi reikalingi duomenys: darbų pavadinimai, vykdytojų grupės, darbų atlikimo tvarka ir trukmė.

2 lentelė. Namo statybos darbų sąrašas

Darbo turinys	**Simbolis**	**Priklausomybė**	**Trukmė (val.)**	**Vykdytojai**
Namo statybų plano sudarymas	D_1	–	10	V
Juodžemio nuėmimas	D_2	D_1	5	P
Aikštelės paruošimas pamatams	D_3	D_2	40	P
Armatūrinio karkaso rišimas	D_4	D_3	20	B1
Pamatų užpylimas betonu	D_5	D_4	10	B1
Džiovinimas	D_6	D_5	240	P
1 aukšto sienų mūrijimas	D_7	D_6	240	B2
Gelžbetonių perdangų montavimas	D_8	D_7	20	B1
2 aukšto sienų mūrijimas	D_9	D_8	100	B2
Stogo konstrukcijos montavimas	D_{10}	D_9	100	B3
Stogo dangos montavimas	D_{11}	D_{10}	100	B3
Stogo apšiltinimas	D_{12}	D_{11}	30	B4
Langų montavimas	D_{13}	D_{11}	10	B7
Durų montavimas	D_{14}	D_{11}	10	B8
Namo apšildymas (iš išorės)	D_{15}	D_{13}	100	B4
Namo armavimas su tinkleliu (iš išorės)	D_{16}	D_{15}	140	B4
Namo glaistymas (iš išorės)	D_{17}	D_{16}	100	B4

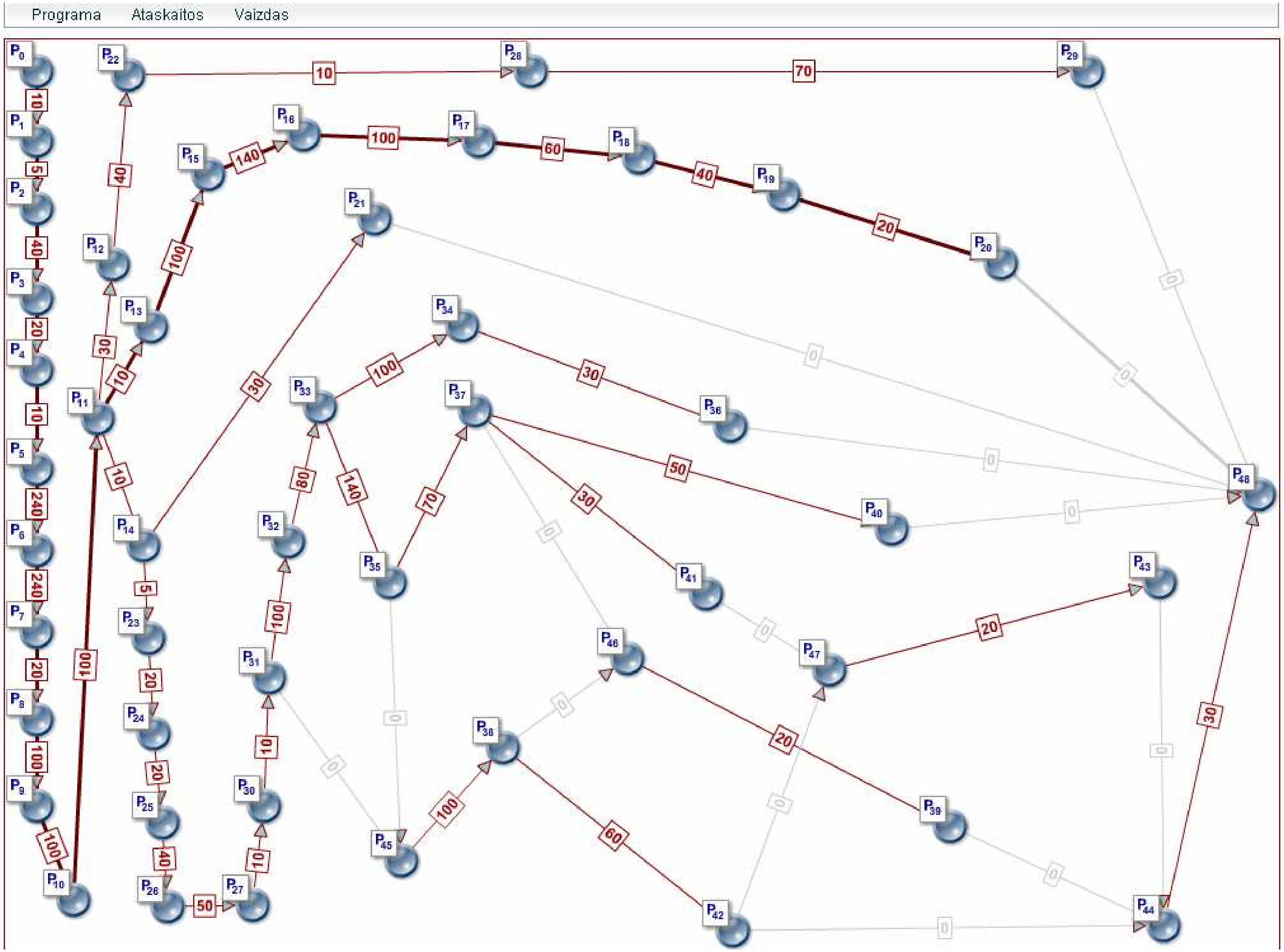
Darbo turinys	**Simbolis**	**Priklausomybė**	**Trukmė (val.)**	**Vykdytojai**
Namo sienų dekoratyvinis tinkavimas (iš išorės)	D_{18}	D_{17}	60	B4
Namo sienų dažymas (iš išorės)	D_{19}	D_{18}	40	B4
Lietvamzdžių pritvirtinimas	D_{20}	D_{19}	20	P
Kanalizacijos išvedžiojimas	D_{21}	D_{14}	30	B5
Ventiliacijos ortakių išvedžiojimas	D_{22}	D_{12}	40	B5
Vandentiekio įvedimas į namą	D_{23}	D_{14}	5	B5
Vandentiekio išvedžiojimas	D_{24}	D_{23}	20	B5
Grunto lyginimas, paruošimas grindims	D_{25}	D_{24}	20	P
Grindų apšildymas (su polestirolių)	D_{26}	D_{25}	40	P
Šildymo išvedžiojimas (šildomų grindų ir radiatorių, vamzdžių išvedžiojimas)	D_{27}	D_{26}	50	B5
Elektros įvedimas į namą	D_{28}	D_{22}	10	B6
Elektros laidų išvedžiojimas	D_{29}	D_{28}	70	B6
Grindų betonavimas	D_{30}	D_{27}	10	B1
Džiovinimas	D_{31}	D_{30}	10	P
Sienų tinkavimas	D_{32}	D_{31}	100	B9
Gipso kartoniniai darbai	D_{33}	D_{32}	80	B9
Lubų glaistymas	D_{34}	D_{33}	100	P
Sienų glaistymas	D_{35}	D_{33}	140	B9
Lubų dažymas	D_{36}	D_{34}	30	B9
Sienų dažymas	D_{37}	D_{35}	70	B9
Plytelių klijavimas	D_{38}	D_{31} , D_{35}	100	B9
Durų montavimas	D_{39}	D_{37} , D_{38}	20	B8
Elektros skydinės montavimas	D_{40}	D_{37} ,	50	B6
Katelinės montavimas	D_{41}	D_{37}	30	B5
Parketo (laminato) klojimas	D_{42}	D_{38}	60	B9
Radiatorių montavimas	D_{43}	D_{41}, D_{42}	20	B5
Namo tvarkymas	D_{44}	D_{39}, D_{42}, D_{43}	30	B10

Turint šią projekto darbų lentelę, tinklinį grafiką galima sudaryti pagal tokią schemą:

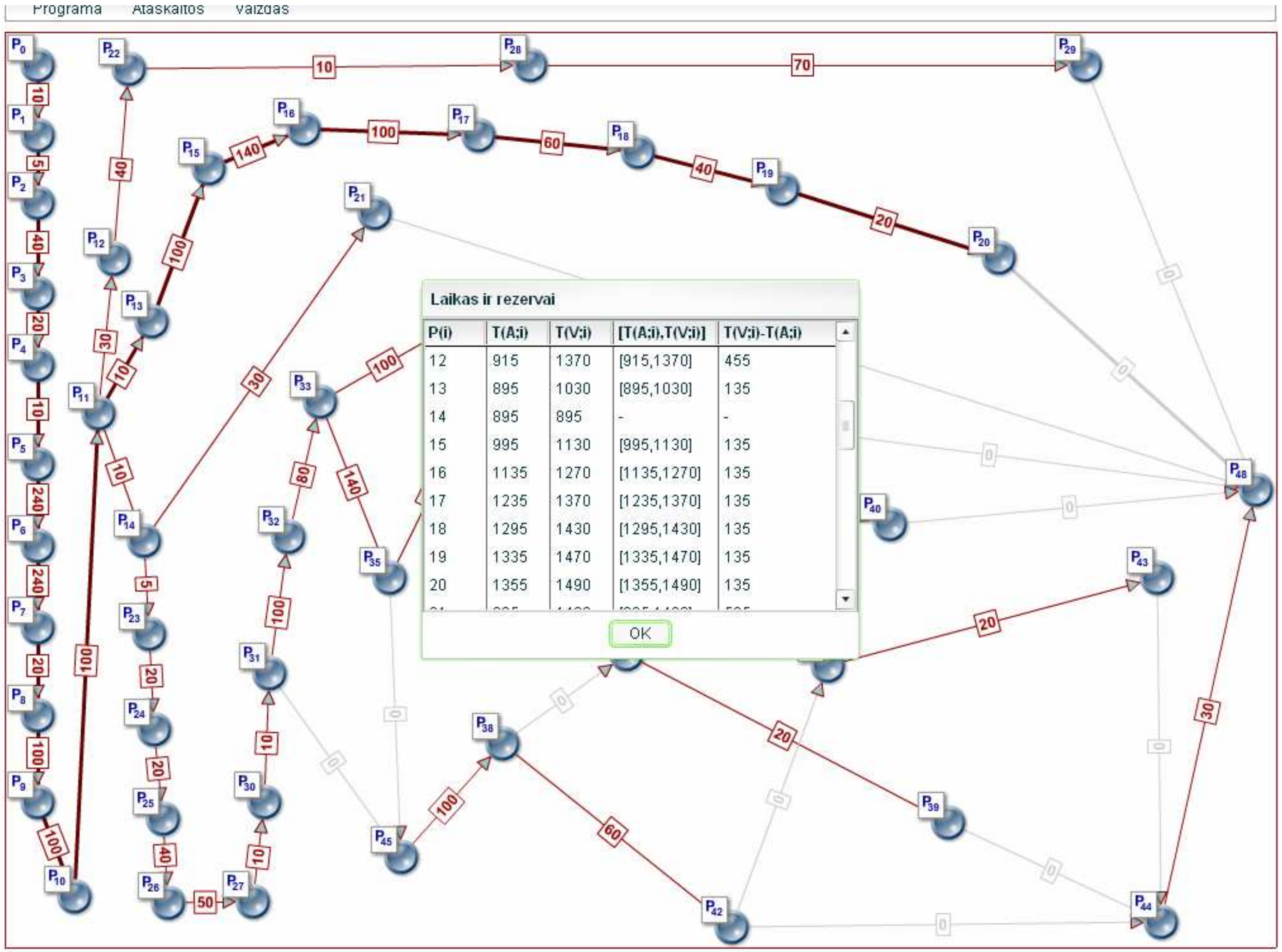
1. Kiekvienam darbui D_j priskiriamas pabaigos įvykis P_j ;
2. Jeigu prieš kurį nors darbą nėra darbų, kurie turi būti užbaigti, tai jo pradžios įvykis žymimas P_0 ir vadinamas projekto pradžia;
3. Jeigu prieš darbą D_j reikia atlikti tik vieną darbą D_i, tai darbo D_j pradžios įvykiu laikomas darbo D_i pabaigos įvykis;
4. Jeigu prieš darbą D_j reikia atlikti darbus D_{i1}, D_{i2}, ..., D_{ik} , kai $k>1$, tai tinklinis grafikas papildomas nauju įvykiu P_{j1}, kuris bus laikomas darbo D_j pradžia. Darbų D_{i1}, D_{i2}, ..., D_{ik} pabaigos įvykiai su įvykiu P_{j1} sujungiami fiktyviais nulinės trukmės darbais. Fiktyvūs darbai grafiškai vaizduojami punktyrinėmis linijomis (programėlėje fiktyvūs darbai bus vaizduojami pilkos spalvos ištisa linija);
5. Darbo, po kurio nėra kitų darbų, pabaigos įvykis vadinamas projekto pabaiga. Jeigu tinkliniame grafike yra kelios projekto pabaigos, tai jas sujungus fiktyviais darbais, liks viena pabaiga.

Remiantis šia schema ir 1 lentele, yra sudarytas tinklinis grafikas, kuris yra pavaizduotas 10 paveiksle.

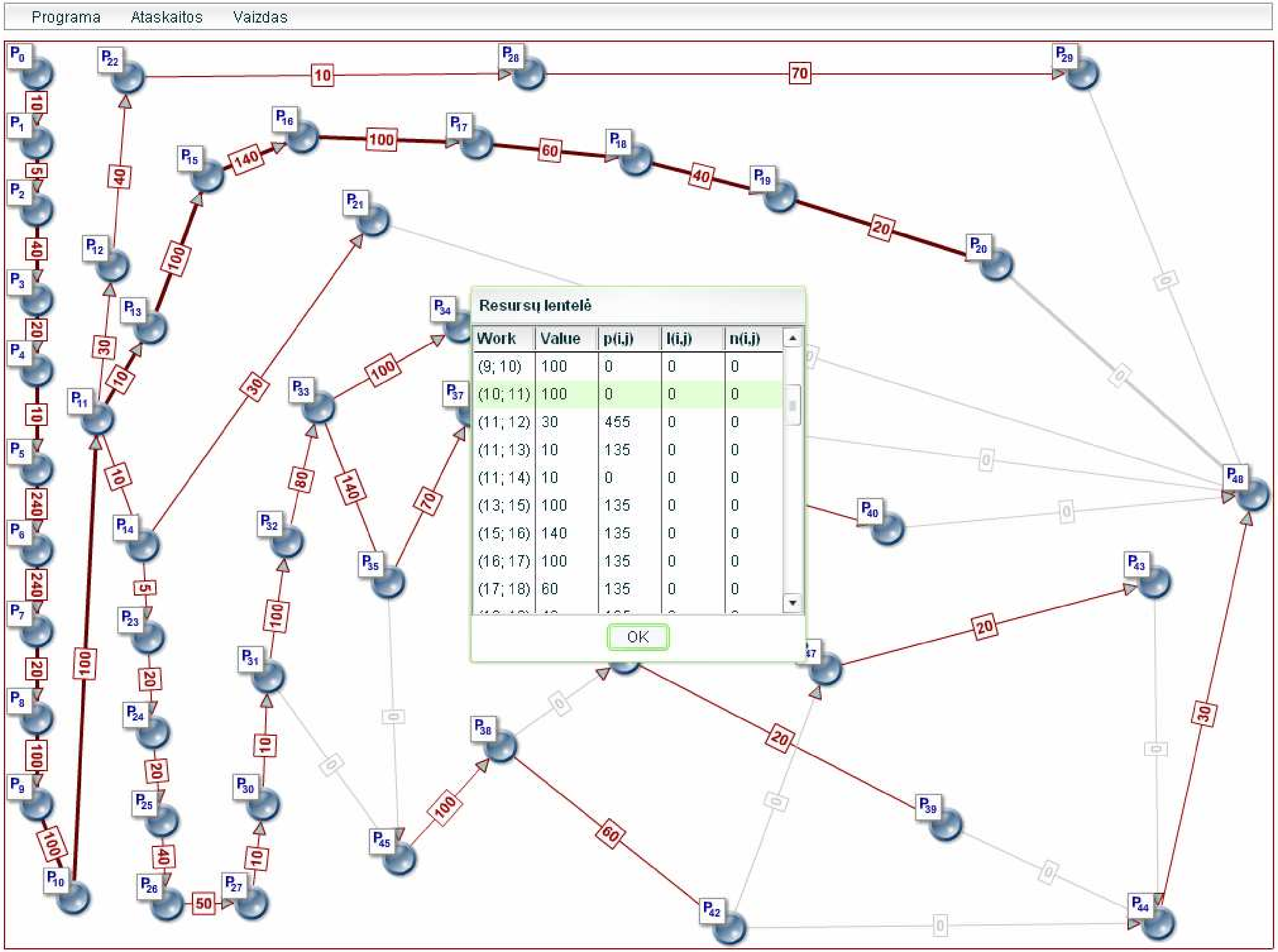
Naudojant programą, buvo apskaičiuoti visi reikalingi parametrai – anksčiausios ir vėlyviausios įvykių pabaigos, įvykių rezervai, darbų laiko rezervai, nustatytas projekto kritinis kelias. Šių parametrų skaičiavimas aprašytas 2.2 skyriuje. Programos skaičiavimo fragmentai pateikti 11 paveiksle ir 12 paveiksle.



10 pav. Namo statybos tinklinis grafikas



11 pav. Projekto T_{Aj}, T_{Vj} ir $r_{j.}$



12 pav. namo statybos darbų laiko rezervai.

Naudojant programą, nustatytas kritinis kelias { ($P_{0,} P_1$), (P_1, P_2), ($P_{2,} P_3$), (P_3, P_4), ($P_{4,}$ P_5), (P_5, P_6 ,) ($P_{6,}$ P_7), (P_7, P_8), (P_8, P_9), ($P_{9,}$ P_{10}), (P_{10}, P_{11}), ($P_{11,}$ P_{13}), (P_{13}, P_{15}), ($P_{15,}$ P_{16}), (P_{16}, P_{17}), (P_{17}, P_{18}), (P_{18}, P_{19}), (P_{19}, P_{20}), (P_{20}, P_{48}) }ir projekto kritinis laikas – 1490 valandų. Taigi, namą galima pastatyti per 149 dienų.

3. Tinklinio planavimo optimizavimo uždaviniai

Praktinėje žmonių veikloje (ekonomikoje, valdyme, projektavime ir kt.) dažnai tenka rinkti vieną sprendinį iš kelių galimų. Atsiranda uždavinys – rasti geriausią sprendinį (gamybos arba projekto). Uždavinys, kai reikia rasti geriausią sprendinį vadinamas *optimizavimo uždaviniu*. Tokie uždaviniai sėkmingai sprendžiami matematiniais metodais.

3.1. Projekto realizavimo kainos minimizavimas

Paprastai, įgyvendinant projektą, darbų kainos (išlaidos, atliekant darbus) priklauso nuo šių darbų trukmės. Dažnai didėjant trukmei, kaina mažėja. Tačiau būna atvejų, kai didėjant trukmei, didėja ir išlaidos.

Nagrinėjamas projektas, kai nustatyta kiekvieno darbo (P_i, P_j) minimali trukmė d_{ij}. Darbo kaina c_{ij} apskaičiuojama pagal formulę

$$c_{ij} = -a_{ij}t_{ij} + b_{ij} \,,$$
$$a_{ij} \geq 0,\ b_{ij} > 0; \qquad \textbf{(3.1)}$$

čia t_{ij} – darbo (P_i, P_j) trukmė.

Reikia rasti projekto realizavimo planą, pagal kurį darbų kompleksas būtų atliktas per trumpiausią laiką su mažiausiomis sąnaudomis.

Remiantis (3.1) formule, galima padaryti išvadą, kad, didinant darbo trukmę, jo kaina mažėja. Vadinasi, kiekvieno darbo trukmė turi būti kiek galima didesnė. Kita vertus, trumpiausias laikas lygus kritiniam laikui, imant $t_{ij}= d_{ij}$. Vadinasi galima užtęsti tik nekritinius darbus, kol jie netapo kritiniais.

Tarkime, kad apskaičiuoti parametrai T_{Aj} ir T_{Vj}, $j = 0,1, ..., n$, ir rastas kritinis kelias bei kritinis laikas. Nekritinių darbų aibė pažymima – R, o įvykių, priklausančių kritiniam keliui, aibė – K. Sakykime, kad T_j –planuojamas įvykio P_j laikas, $j = 0,1, ..., n$. Kad darbų kompleksas būtų atliktas per kritinį laiką, dydžiai T_j turi tenkinti šias sąlygas:

$$T_{Aj} = T_{Vj} = T_j, \text{ kai } P_j \in K;$$
$$T_j - T_i = t_{ij} \geq d_{ij}, \text{ kai } (P_i, P_j) \in R \qquad \textbf{(3.2)}$$

Tada projekto realizavimo kaina yra

$$z = \sum_{(P_i,P_j)\in R} \left[-a_{ij}(T_j - T_i) + b_{ij}\right]. \qquad \textbf{(3.3)}$$

Dabar galima pateikti nagrinėjamo uždavinio matematinį modelį.

Reikia rasti T_j, $j = 0,1, ..., n$ reikšmes, tenkinančias (3.2) sąlygas, su kuriomis (3.3) funkcija įgyja mažiausią reikšmę. Taigi, nagrinėjamo uždavinio matematinis modelis yra šitoks tiesinio programavimo uždavinys:

$$min \sum_{(P_i,P_j)\in R}\left[-a_{ij}(T_j - T_i) + b_{ij}\right], \quad \text{kai} \qquad \textbf{(3.4)}$$

$$T_j - T_i \geq d_{ij}\text{ , } jei\ (P_i, P_j) \in R;$$
$$T_i = T_{Ai},\ jei\ P_i \in K. \qquad \textbf{(3.5)}$$

Jeigu reikia sudaryti projekto realizavimo planą, pagal kurį darbų kompleksas būtų atliktas su mažiausiomis sąnaudomis per tam tikrą laiką T, didesnį už kritinį laiką. Matematinis modelis bus šitoks:

$$min \sum_{(P_i,P_j)\in R}\left[-a_{ij}(T_j - T_i) + b_{ij}\right], \quad \text{kai} \qquad \textbf{(3.6)}$$

$$T_j - T_i \geq d_{ij}\text{ , visiems } (P_i, P_j);$$
$$T_0 = 0,\ T_n \leq T. \qquad \textbf{(3.7)}$$

Pastaba. Čia T_n ne kritinis, o planuojamas įvykio P_n laikas.

Toliau, remiantis šiuo modeliu, nagrinėjamas praktinis uždavinys.

4 uždavinys. Vyksta buto rekonstravimo darbai. Prie šių darbų komplekso dirba 6 darbininkai ir 1 darbo vadovas. Kiekvienam darbininkui paskirtas tam tikras darbas:

I darbininkas atlieka du darbus: darbą (P_0, P_1), jo trukmė 3 dienos; darbą (P_0, P_2), jo trukmė $t_{02}=4$;

II darbininkas - (P_1, P_2), $t_{12}=2$, $(P_1, P3)$, $t_{13}=7$,(P_1, P_4), $t_{14}=3$ *ir* (P_1, P_5), $t_{15}=5$;

III darbininkas - (P_2, P_3), $t_{23}=6$ *ir* (P_2, P_4), $t_{24}=1$;

IV darbininkas - (P_3, P_6), $t_{36}=8$;

V darbininkas - (P_4, P_5), $t_{45}=8$;

VI darbininkas - (P_5, P_6), $t_{56}=2$.

Darbo vadovas priima visų darbininkų darbus.

Šių darbų komplekso tinklinis grafikas, kuriame pažymėti įvykiai ir nurodytos darbų trukmės, pavaizduotas 13 paveiksle. Skaičiai surašyti prie grafiko briaunų, reiškia darbų (P_i, P_j) minimalią trukmę d_{ij}. Darbų kainos c_{ij} apskaičiuojamos pagal formules:

$c_{01} = -5\ t_{01} + 100;$ $c_{23} = -4\ t_{23} + 80;$

$c_{02} = -3\ t_{02} + 100;$ $c_{24} = -5\ t_{24} + 120;$

$c_{12} = -10\ t_{12} + 50;$ $c_{36} = -3\ t_{36} + 100;$

$c_{13} = -8\ t_{13} + 150;$ $c_{45} = -10\ t_{45} + 250;$

$c_{14} = -2\ t_{14} + 60;$ $c_{56} = -4\ t_{56} + 40.$

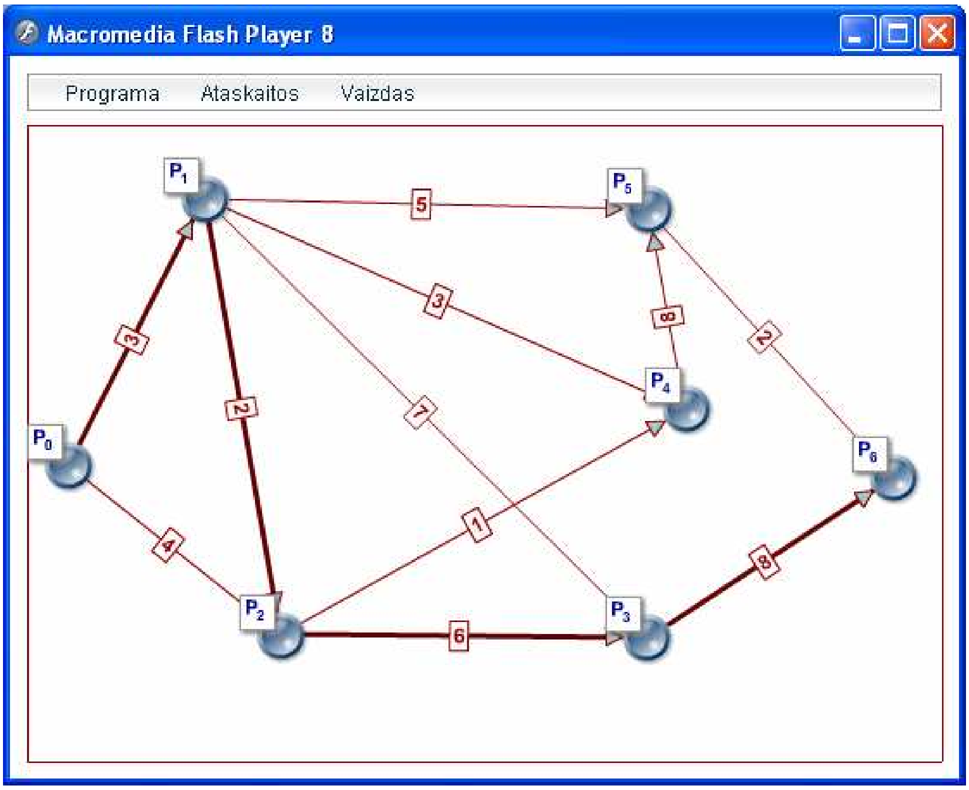
$c_{15} = -10\ t_{15} + 100;$

Reikia rasti tokį buto rekonstravimo planą, pagal kurį visų darbų kompleksas būtų atliktas per trumpiausią laiką su mažiausiomis sąnaudomis.

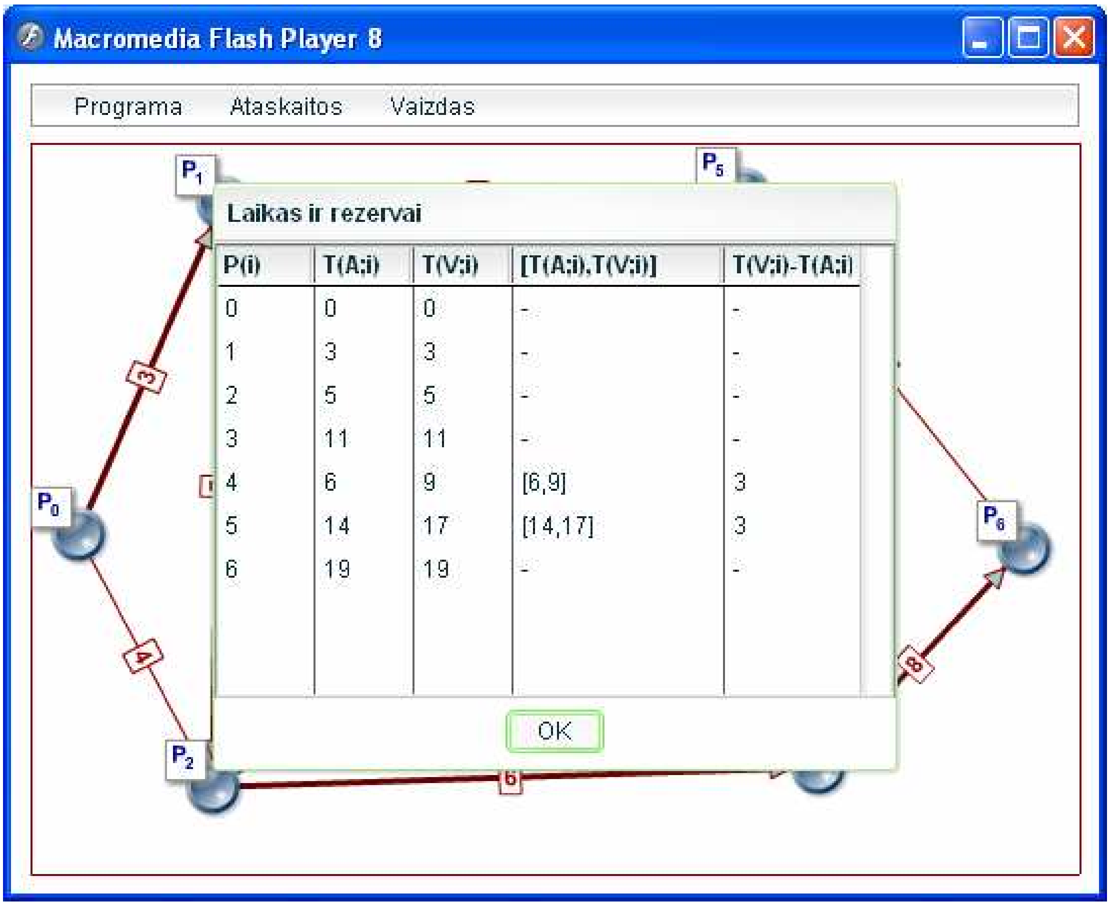
Sprendimas

Šio darbų komplekso tinklinis grafikas pavaizduotas 13 paveiksle.

Pagrindiniai parametrai apskaičiuoti programos pagalba pateikti 14 paveiksle.



13 pav. darbų komplekso tinklinis grafikas



P(i)	T(A;i)	T(V;i)	[T(A;i),T(V;i)]	T(V;i)-T(A;i)
0	0	0	-	-
1	3	3	-	-
2	5	5	-	-
3	11	11	-	-
4	6	9	[6,9]	3
5	14	17	[14,17]	3
6	19	19	-	-

14 pav. T_{Aj} ir T_{Vj} skaičiavimų lentelė.

Sudaromas matematinis modelis.

Šito darbų komplekso kritinis kelias yra (žr. 13 pav.) {$(P_0, P_1), (P_1,P_2), (P_2,P_3), (P_3,P_6)$}. Kritinis trumpiausias laikas – T_6 = 19. Nekritinių darbų aibė: R = { $(P_0, P_2), (P_1,P_3), (P_1,P_4), (P_1,P_5), (P_2, P_4), (P_4,P_5), (P_5,P_6)$ }.Įvykių, priklausančių kritiniam keliui, aibė $K = \{P_0, P_1, P_2, P_3, P_6\}$

Pagal (3.4) ir (3.5) formules užrašomas tiesinio programavimo uždavinys:

$$\min (8 T_0 +25T_1 - 4T_2 - 9T_3+3T_4 - 16T_5 - 7T_6 + 1150), \qquad \text{kai}$$

$$T_2 - T_0 \geq 4; \qquad T_4 - T_2 \geq 1;$$

$$T_3 - T_1 \geq 7; \qquad T_5 - T_4 \geq 8;$$

$$T_4 - T_1 \geq 3; \qquad T_6 - T_5 \geq 2;$$

$$T_5 - T_1 \geq 5;$$

$$T_0 = 0,\ T_1 = 3,\ T_2 = 5,\ T_3 = 11,\ T_6 = 19.$$

Eliminavus dydžius T_J, turinčius tik vieną reikšmę, gautas šitoks uždavinys:

$$\min(3T_4 - 16T_5 + 973), \qquad \text{kai}$$

$$T_5 - T_4 \geq 8;$$

$$T_4 \geq 6,\ T_5 \geq 8,\ T_5 \leq 17.$$

Šitas uždavinys sprendžiamas simplekso metodu. Tam reikia standartinį uždavinį suvesti į kanoninį pavidalą.

Daromas keitinys: $R_1 := T_4$ ir $R_2 := T_5$. Įvedami papildomi kintamieji: R_3, R_4, R_5, R_6. Simplekso metodo sprendimas:

UŽDAVINIO SĄLYGA

$$\min(3R_1 - 16R_2 + 0R_3 + 0R_4 + 0R_5 + 0R_6)$$

$$\text{kai} \begin{cases} 1R_1 + 1R_2 - 1R_3 + 0R_4 + 0R_5 + 0R_6 = 8 \\ 1R_1 + 0R_2 + 0R_3 - 1R_4 + 0R_5 + 0R_6 = 6 \\ 0R_1 + 1R_2 + 0R_3 + 0R_4 - 1R_5 + 0R_6 = 8 \end{cases}$$

$$\mathbf{C} = \begin{pmatrix} 3 \\ -16 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} 8 \\ 6 \\ 8 \\ 17 \end{pmatrix} \quad \mathbf{A} = \begin{pmatrix} -1 & 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$R_3 = 0$	$R_4 = 0$

$1R_1 + 1R_2 + 0R_5 + 0R_6 = 8$

$1R_1 + 0R_2 + 0R_5 + 0R_6 = 6$

$0R_1 + 1R_2 - 1R_5 + 0R_6 = 8$

$0R_1 + 1R_2 + 0R_5 + 1R_6 = 17$

$R_1 = 6$
$R_2 = 14$
$R_5 = 6$
$R_6 = 3$

$$\mathbf{B} = (a^1; a^2; a^5; a^6) = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} \quad |\mathbf{B}| = 1 \quad \mathbf{D} = (a^3; a^4) = \begin{pmatrix} -1 & 0 \\ 0 & -1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

$$\mathbf{B^{-1}} = 1 \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & -1 & 0 \\ -1 & -1 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & -1 & 0 \\ -1 & -1 & 0 & 1 \end{pmatrix}$$

$$\mathbf{B^{-1}D} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & -1 & 0 \\ -1 & -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} -1 & 0 \\ 0 & -1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ -1 & -1 \\ -1 & -1 \\ 1 & 1 \end{pmatrix}$$

		C_D	0	0
B	C_B	X_B	A_3	A_4
a^1	3	6	0	-1
a^2	-16	14	-1	-1
a^5	0	6	-1	-1
a^6	0	3	1	1
	Z	-206	16	13
		Δ_D	-16	-13

$R_3 = 0$	$R_6 = 0$

$1R_1 + 1R_2 + 0R_4 + 0R_5 = 8$

$1R_1 + 0R_2 - 1R_4 + 0R_5 = 6$

$0R_1 + 1R_2 + 0R_4 - 1R_5 = 8$

$0R_1 + 1R_2 + 0R_4 + 0R_5 = 17$

$R_1 = 9$
$R_2 = 17$
$R_4 = 3$
$R_5 = 9$

$$\mathbf{B}=(a^1;a^2;a^4;a^5)=\begin{pmatrix} -1 & 1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad \mathbf{|B|}=-1 \quad \mathbf{D}=(a^3;a^6)=\begin{pmatrix} -1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\mathbf{B^{-1}}=-1\begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 \\ 1 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{pmatrix}=\begin{pmatrix} -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ -1 & -1 & 0 & 1 \\ 0 & 0 & -1 & 1 \end{pmatrix}$$

$$\mathbf{B^{-1}D}=\begin{pmatrix} -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ -1 & -1 & 0 & 1 \\ 0 & 0 & -1 & 1 \end{pmatrix}\begin{pmatrix} -1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}=\begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 1 \\ 0 & 1 \end{pmatrix}$$

		C_D	0	0
B	C_B	X_B	A_3	A_6
a^1	3	9	1	1
a^2	-16	17	0	1
a^4	0	3	1	1
a^5	0	9	0	1
	Z	-245	3	-13
		Δ_D	-3	13

$R_4 = 0$	$R_6 = 0$

$1R_1 + 1R_2 - 1R_3 + 0R_5 = 8$

$1R_1 + 0R_2 + 0R_3 + 0R_5 = 6$

$0R_1 + 1R_2 + 0R_3 - 1R_5 = 8$

$0R_1 + 1R_2 + 0R_3 + 0R_5 = 17$

$R_1 = 6$
$R_2 = 17$
$R_3 = 3$
$R_5 = 9$

$$\mathbf{B}=(a^1;a^2;a^3;a^5)=\begin{pmatrix} -1 & 1 & -1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad \mathbf{|B|}=-1 \quad \mathbf{D}=(a^4;a^6)=\begin{pmatrix} 0 & 0 \\ -1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\mathbf{B^{-1}}=-1\begin{pmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{pmatrix}=\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & -1 & 0 & 1 \\ 0 & 0 & -1 & 1 \end{pmatrix}$$

$$\mathbf{B^{-1}D}=\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & -1 & 0 & 1 \\ 0 & 0 & -1 & 1 \end{pmatrix}\begin{pmatrix} 0 & 0 \\ -1 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}=\begin{pmatrix} -1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 0 & 1 \end{pmatrix}$$

		C_D	0	0
B	C_B	X_B	A_4	A_6
a^1	3	6	-1	0
a^2	-16	17	0	1
a^3	0	3	1	1
a^5	0	9	0	1
	Z	-254	-3	-16
		Δ_D	3	16

Po trijų nuoseklių žingsnių gauta, kad $R_1 = T_4^* = 6$ *ir* $R_2 = T_5^* = 17$.

Vadinasi įvykiams P_i^*, $i = 0, 1, ..., 6$, reikia planuoti šitokį laiką:

$T_0^* = 0;$	$T_4^* = 6;$
$T_1^* = 3;$	$T_5^* = 17;$
$T_2^* = 5;$	$T_6^* = 19.$
$T_3^* = 11;$	

Optimali darbo trukmė yra $t_{ij}^* = T_j^* - T_i^*$. Taigi,

$t_{01}^* = 3;$	$t_{23}^* = 6;$
$t_{02}^* = 5;$	$t_{24}^* = 1;$
$t_{12}^* = 2;$	$t_{36}^* = 8;$
$t_{13}^* = 8;$	$t_{45}^* = 11;$
$t_{15}^* = 14;$	$t_{56}^* = 2.$

Dabar galima apskaičiuoti, kad buto rekonstravimo kaina yra $z_{min} = 719$.

IŠVADOS

Šiame magistriniame darbe buvo siekiama išnagrinėti tinklinio planavimo uždavinius.

Darbe pateiktas vienas kritinio kelio nustatymo būdas. Išnagrinėti bei sudaryti tinklinio grafiko parametrų skaičiavimo algoritmai. Taip pat, yra sudaryta kompiuterinė programa kritiniam tinklinio grafiko keliui rasti. Naudojant šią programą yra išspręsti du uždaviniai.

Naudojant tinklinio planavimo metodus galima efektyviai išnaudoti laiką ir kitus resursus, optimaliai organizuoti ir realizuoti sudėtingus projektus.

Tinklinio planavimo metodai gali sudominti ne tik matematikus, bet ir vadybininkus, inžinierius, kurie užsiima sudėtingų sistemų planavimo ir valdymo klausymais.

SUMMARY

Problems of net planning

The goal of this master's degree thesis is to analyze problems of net planning.

The thesis defines one way of determining the critical path. There are also algorithms of net graph parameters calculation analyzed and constructed. In addition, the thesis contains a computer program which determines critical path of net graph. Two problems are solved using this program.

Implementation of net planning methods can help use time and other resources more effectively, organize and accomplish complex projects.

Methods of net planning can gain interest not only of mathematicians, but also of managers, engineers, and those responsible for planning and managing complex systems.

LITERATŪROS SĄRAŠAS

1. М. О. Асанов. Дискретная оптимизация, УралНАУКА, Екатеринбург, 1998.
2. С. Зуховицкий, И. Радчик. Математические методы сетевого планирования, Москва: издательство Наука, 1965.
3. V. Čiočys, R Jasilionis. Matematinis programavimas, Vilnius: leidykla Mokslas, 1990.
4. Л. С. Костевичю Математическое программирование, Минск, Гриф РБ, 2003.
5. Н. И. Холод. Экономико – математические методы и модели, Минск, БГЭУ, 2000.
6. В. Н. Захаров. Алгоритмические методы решения задач оптимального планирования и управления, ВАД. 1986.
7. Е. Г. Давыдов. Исследование операций, Москва, 1990.
8. A. Žilinskas. Naujieji projektavimo metodai, Vilnius: leidykla Mintis, 1990.
9. Ore O., Grafai ir jų pritaikymai, Vilnius: leidykla ,Mintis, 1973
10. Е. Г. Давыдов., Георгиевич, Игры, графы, ресурсы.
11. Papi Ž., Modernioji matematika, I d., Paris, 1963.
12. Ore O., Theory of Graphs (Ore O., Grafų teorija), Providence (JAV), 1962
13. Доморяд А., Математические игры и развлечения, М., Физматгиз, 1961
14. Ch. Lemchenas, J. Macaitis. Rusų – Lietuvių kalbos žodynas, Vilnius: Mokslo ir enciklopedijų leidybos institutas, 1997.
15. J Kubilius. Matematikos terminų žodynas, Vilnius, Mokslo ir enciklopedijų leidykla, 1994.
16. Г. Верников. Введение в проектный менеджмент, http://www.vernikov.ru/material80.html
17. PERT/ CPM for Project Scheluding and Management, http://www.interventions.org/pertcpm.html
18. http://www.google.lt/