**ՀԱՅԱՍՏԱՆԻ ԱԶԳԱՅԻՆ ՊՈԼԻՏԵԽՆԻԿԱԿԱՆ ՀԱՄԱԼՍԱՐԱՆ**

Կուրսային Աշխատանք

(Դիսկրետ Մաթեմատիկա)

Առաջադրանքը տրվեց` 01.01.2022

Նախագծի պաշտպանությունը` 02.06.2022

Թեմա՝

Վերջավոր գրաֆի երկկողմանիության որոշման ալգորիթմի մշակում և ծրագրային իրացում

Դասախոս՝ Գարեգին Սարգսյան

Ուսանող՝ Վահե Զիլաբյան

Խումբ ՝ 919Ս

# **Բովանդակություն**

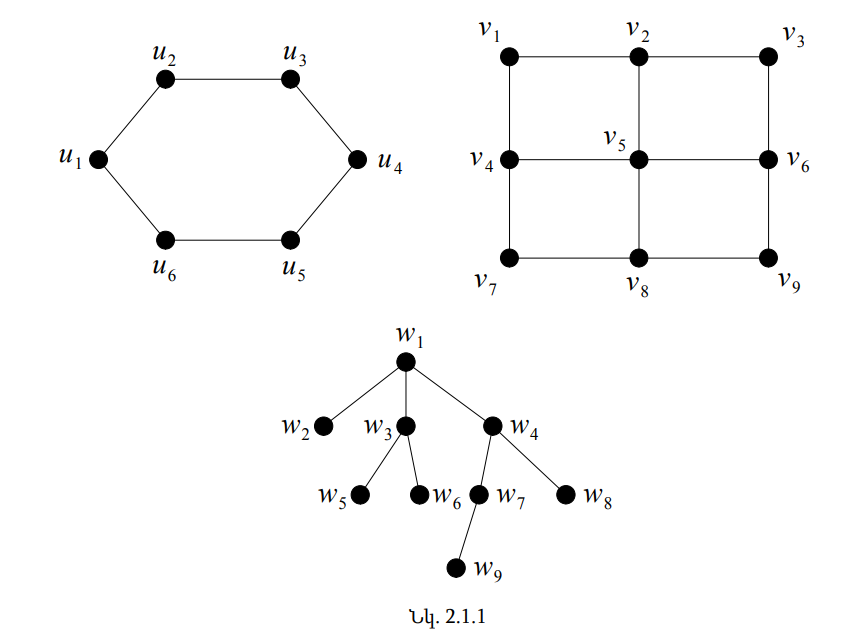
ՆԵՐԱԾՈՒԹՅՈՒՆ…………………………………………………………………………………3

ԽՆԴՐԻ ԴՐՎԱԾՔ…………………………………………………………………………………..4

ԽՆԴՐԻ ԼՈՒԾՈՒՄ…………………………………………………………………………............5

ՕԳՏԱԳՈՐԾՎԱԾ ԳՐԱԿԱՆՈՒԹՅԱՆ ՑԱՆԿ…………………………………………………..9

## **Ներածություն**



**G = (V, E)** Գրաֆը կանվանենք երկկողմանի, եթե **V** բազմությունը հնարավոր է տրոհել երկու ենթաբազմությունների -ի և -ի այնպես, որ G գրաֆի ցանկացած կող կից է մեկ գագաթի -ից և մեկ գագաթի -ից: Նշենք, որ նկ. 2.1.1-ում պատկերված երեք գրաֆներն էլ երկկողմանի են: Իրոք, դիտարկենք նրանցից առաջինի գագաթների բազմության հետևյալ տրոհումը. և , և նկատենք, որ այդ գրաֆի ցանկացած կող միացնում է մեկական գագաթ -ից և -ից: Երկրորդ գրաֆի երկկողմանիության մեջ համոզվելու համար դիտարկենք նրա գագաթների հետևյալ տրոհումը և , և նկատենք, որ այդ գրաֆի ցանկացած կող միացնում է մեկական գագաթ -ից և -ից: Եվ, վերջապես, երրորդ գրաֆի երկկողմանիության մեջ համոզվելու համար դիտարկենք և , և նկատենք, որ այդ գրաֆի ցանկացած կող միացնում է մեկական գագաթ -ից և -ից: Ստորև ներկայացված են երեք թեորեմ գրաֆի երկկողմանիության վերաբերյալ:

**Թեորեմ (Դ. Քյոնիգ):** Որպեսզի **G = (V, E)** գրաֆը լինի երկկողմանի, անհրաժեշտ է և բավարար, որ այն չպարունակի կենտ երկարություն ունեցող պարզ ցիկլ:

**Թեորեմ**: Որպեսզի, **G = (V,E)** գրաֆը լինի երկկողմանի, անհրաժեշտ է և բավարար, որ նրա կցության **B(G)** մատրիցը լինի տոտալ ունիմոդուլյար:

**Թեորեմ (Պ. Էրդյոշ):** Կամայական **G** գրաֆ պարունակում է կմախքային երկկողմանի **H** ենթագրաֆ, որում :

## **Խնդրի դրվածքը**

Տրված G(V, E) վերջավոր գրաֆի երկկողմանիության որոշման ալգորիթմի մշակում և ծրագրային իրացում։

## **Խնդրի լուծում**

Դիցուք G = (V, E) գրաֆ է.

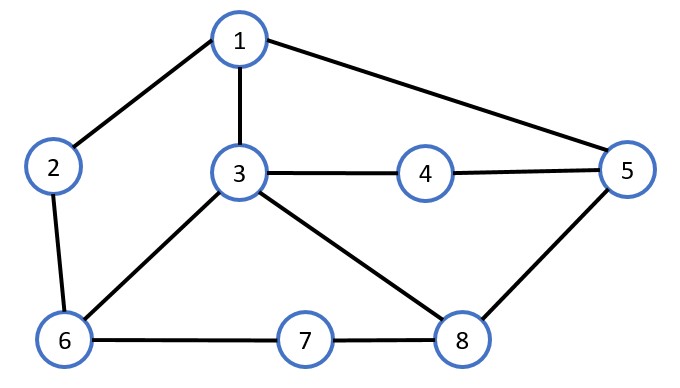
Առաջադրանքը լուծելու համար օգտվում ենք DFS ալգորիթմից (deepth-first search): Որպեսզի պարզենք G գրաֆը երկկողմանի է թե ոչ, օգտվենք հետևյալ երկու գույնով ներկման ալգորիթմից, երկկողմանի գրաֆի ցանկացած կող պետք է իրար միացնի տարբեր գույն ունեցող երկու գագաթ՝

Սկզբում գրաֆի բոլոր գագաթները համարենք անգույն, հետո վերցնենք առաջին գագաթը և ներկենք առաջին գույնով (վերագրենք 1), նրա հարևան գագաթները կարող են լինել՝

* Անգույն, այս դեպքում ներկում ենք մյուս գույնով (վերագրում ենք 2)
* Ներկված նույն գույնով ինչպես մեր գագաթը, ապա գրաֆը չի կարող լինել երկկողմանի, դուրս ենք գալիս ցիկլից
* Ներկված մյուս գույնով, այս դեպքում բաց ենք թողնում տվյալ գագաթը և անցնում մյուսին

Եթե բոլոր գագաթները ներված են և չկան նույն գույնի հարևան գագաթներ ապա գրաֆը երկկողմանի է։ Դատարկ գրաֆը նույնպես համարվում է երկկողմանի։

Դիտարկենք հետևյալ օրինակը՝

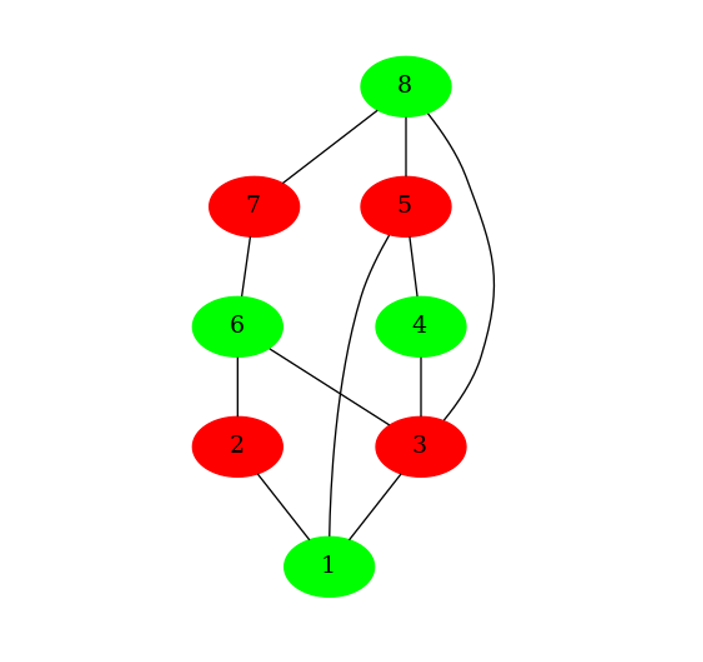


Որպեսզի ավելի հասկանալի լինի ապա 1-ով վերագրելը անվանենք ներկել կանաչ գույնով, իսկ 2-ով վերագրելը՝ ներկում կարմիրով։

Վերցնենք գագաթ 1-ը և այն ներկենք կանաչ գույնով, ապա անցնենք նրա հարևան գագաթներից 2-ին, քանի որ այն անգույն է, ապա ներկենք կարմիրով։ Հիմա ստուգենք 2-րդ գագաթի հարևանններին, անցնում ենք գագաթ 1-ի վրայով քանի որ այն արդեն ներկված է հակառակ գույնով։ Գագաթ 6-ը անգույն է, հետևաբար այն ներկենք կանաչով։ Գագաթ 6-ի հարևաններից ընտրենք 3-ը, այն անգույն է՝ ներկենք կարմիրով։ Գագաթ 3-րդի հարևաններն են 1, 4, 6։ Որոնցից 1-ը և 6-ը արդեն ներկված են հակառակ(կանաչ) գույնով, հետևաբար անցնում ենք դրանց վրայով և ընտրում գագաթ 4-ը և ներկում կանաչ։ Գագաթ 4-ի հարևաններից կարող ենք ընտրել 5-ը քանի որ այն անգույն է, նրան ներկենք կարմիր։ Գագաթ 5-ի հարևաններից անգույն է միայն 8-ը,իսկ մնացածը ներկված են հակառակ գույնով, հետևաբար գագաթ 8-ը ներկենք կանաչ։ Անգույն մնաց միայն գագաթ 7-ը, որն 8-րդ գագաթի հարևաններից է, այն ներկենք կարմիր։

Այսպիսով մենք անցանք բոլոր գագաթների վրայով և նրանց ներկեցինք 2 տարբեր գույներով։ Եվ քանի որ չհանդիպեցինք միևնույն գույնով ներկված 2 հարևան գագաթ ապա գրաֆը երկկողմանի է։

Ծրագիրը գրաֆի տվյալները կարդում է ֆայլից՝ որտեղ նշված է գրաֆի գագաթների քանակը և զույգ առ զույգ գրված են հարևան գագաթները։ Կոնսոլում արտածվում է գրաֆի կողերի քանակը, հարևանության մատրիցը, տպում է ամեն գագաթի հարևան գագաթները, եթե տվյալ գագաթը չունի որևէ հարևան ապա կտպի որ այն մեկուսացված գագաթ է, և արտածում է երկկողմանի է թե ոչ։ Ինչպես նաև graphviz գործիքի միջոցով ստանում ենք նաև գրաֆի վիզուալ պատկերը։



Հարևան գագաթները ներկվել են տարբեր գույներով, եթե որևէ կող ունենա միևնույն գույնի երկու հարևան գագաթ, ապա գրաֆը երկկողմանի չէ։



Ծրագիրը գրվել է C++ լեզվով։ Կոդը՝ [Այստեղ](https://github.com/VaheZilabyan/bipartite_graph.git)

**Օգտագործված գրականության ցանկ**

Պ․Ա․ Պետրոսյան, Վ․Վ․ Մկրտչյան, Ռ․Ռ․ Քամալյան - Գրաֆների տեսություն Ռ․Ն․ Տոնոյան - Դիսկրետ մաթեմատիկայի դասընթաց