မင်းဘယ်မှာလဲ ငါသိတယ်

ဒါကတော့ ကျွန်တော်စာသင်ဖြစ်တဲ့ ကျောင်းသားအားလုံးကို ပြဖြစ်အောင်ပြတဲ့ မျက်လှည့်လေးပါ။ အခု စာဖတ်သူကိုလည်း စာအုပ်ထဲကနေပဲ မျက်လှည့်လှမ်းပြပါမယ်။ အနားမှာ ခဲတံတစ်ချောင်းဆောင်ထားပါ။ အောက်က 3 × 3 grid ထဲက A လို့ရေးထားတဲ့ အကွက်ပေါ်မှာ ကိုယ့်လက်ညိုးကိုတင်ထားလိုက်ပါ။ ကျွန်တော်က “ရွှေ့ပါ” လို့ပြောရင် ကိုယ့်လက်ရောက်နေတဲ့ အကွက်ရဲ့ အပေါ်၊ အောက်၊ ဘယ်၊ ညာကပ်လျက်မှာရှိတဲ့ နှစ်သက်ရာအကွက်ဆီသို့ လက်ကိုရွှေ့နိုင်ပါတယ်။ ရွှေ့တဲ့အခါမှာ grid ရဲ့ အပြင်ကိုထွက်လို့ မရပါဘူး။ နောက်ဆုံးအရေးကြီးတာက ရွှေ့ဖို့ပြောရင် ရွှေ့ကိုရွှေ့ရပါမယ်၊ နေရာမှာ ရပ်နေခွင့်မရှိပါဘူး။

A	B	C
D	E	F
G	H	I

ပုံ 1 - မျက်လှည့်ပြမယ့် အကွက်

ကဲ… မျက်လှည့်စပါပြီ။

“နှစ်ကြိမ်ရွှေ့လိုက်ပါ။”

“စာဖတ်သူရဲ့လက်ညိုးဟာ D မှာမရှိပါဘူး။ D ပါတဲ့အကွက်ကို ခဲတံနဲ့ခြစ်ပြီး ဖျက်ဆီးလိုက်ပါ။”

ဖျက်ဆီးလိုက်တဲ့အကွက်တွေပေါ်ကို ရွှေ့ခွင့်မရှိတော့ပါဘူး။ စာဖတ်သူရဲ့လက်ညိုးရှိနေတဲ့အကွက်ကို မဖျက်ဆီးမိဖို့ အာမခံပါတယ်။ ကဲ… ဆက်ရအောင်။

“သုံးကြိမ်ရွှေ့လိုက်ပါ။”

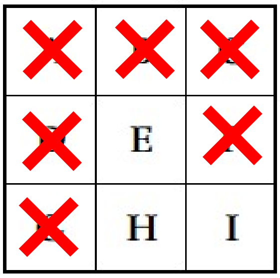
“စာဖတ်သူရဲ့လက်ညိုးက G နဲ့ C မှာမရှိပါဘူး။ အဲ့အကွက်တွေကို ဖျက်ဆီးလိုက်ပါ။”

“ဆက်ပြီးတော့ နှစ်ကြိမ်ရွှေ့ပေးပါဦး။”

“စာဖတ်သူရဲ့ လက်ညှိုးက A မှာမရှိပါဘူး။ A ပါတဲ့အကွက်ကို ခဲတံနဲ့ခြစ်ထုတ်လိုက်ပါ။”

“ဆက်ပြီးတော့ တစ်ကြိမ်ပဲရွှေ့ပေးပါ။”

“စာဖတ်သူရဲ့ လက်ညှိုးက B နဲ့ F မှာမရှိပါဘူး။ အဲ့အကွက်တွေကို ဖျက်လိုက်ပါ။”



ပုံ 2 – B နဲ့ F ကိုဖျက်အပြီးမှာ ရမယ့်ပုံ

“ဆက်ပြီးတော့ သုံးကြိမ်ရွှေ့ပေးပါဦး။”

“စာဖတ်သူရဲ့ လက်ညှိုးက I မှာမရှိပါဘူး။ I ကိုဖျက်လိုက်ပါ။”

“ဆက်ပြီးတော့ တစ်ကြိမ်ရွှေ့ပေးပါ။”

“စာဖတ်သူရဲ့လက်ညှိုးက H မှာမရှိပါဘူး။ H ကိုဖျက်လိုက်ပါ။”

စာဖတ်သူရဲ့လက်ညှိုးဟာ အခုနောက်ဆုံးကျန်ရှိနေတဲ့အကွက်ဖြစ်တဲ့ E မှာရှိပါတယ်။ တကယ်လို့များ ကျွန်တော်မှားတယ်ဆိုရင် ဥပမာအားဖြင့် ထောင့်ဖြတ်အတိုင်းရွှေ့မိတာမျိုးလုပ်မိတဲ့အတွက် ဖြစ်နိုင်ပါတယ်။ လမ်းညွှန်ချက်တွေသေချာပြန်ဖတ်ပြီးတော့ နောက်တစ်ခေါက်ထပ်လုပ်ကြည့်ပါ။

ဒီပြကွက်က လူအများကြီးရှိတဲ့ စာသင်ခန်းတွေထဲမှာ ပြလို့အတော်ကောင်းပါတယ်။ လူတွေဘယ်လောက်များများ၊ သူတို့လက်တွေ ဘယ်ကိုသွားသွား နောက်ဆုံးမှာ အကွက် E ထဲကိုပဲ ဘေးမသီရန်မခရောက်လာပါလိမ့်မယ်။

နောက်ကွယ်က သင်္ချာကိုသိသွားရင် နောက်ဆုံးမှာရောက်လာမယ့်အကွက်က E မှမဟုတ်ဘဲ၊ စိတ်ကြိုက်အကွက် ရွေးခွင့်ပေးလို့ပါ ရပါတယ်။ လမ်းညွှန်ချက်တွေဖန်တီးရတာက တကယ်လွယ်လွယ်လေးပါ။

နောက်ကွယ်က သင်္ချာ

မျက်လှည့်ရဲ့ လျှို့ဝှက်ချက်ကတော့ အရမ်းကိုရိုးစင်းလွန်းလို့ လေးတန်း၊ ငါးတန်းကလေးကိုတောင် အပြည့်အဝနားလည် အောင်ရှင်းပြလို့ရပါတယ်။ အိုင်ဒီယာကတော့ ပေးရင်း 3×3 အကွက်ကို စစ်တုရင်ခုံပေါ်က အကွက်တွေကို အရောင်ခြယ် သလိုမျိုး အဖြူတစ်လှည့် အနက်တစ်လှည့် ခြယ်လိုက်ဖို့ပါပဲ။

A	B	C
D	E	F
G	H	I

ပုံ 3 – အဖြူ၊ အမည်း တစ်လှည့်စီခြယ်ပြီးပုံ

ဒီလိုခြယ်လိုက်ခြင်းကြောင့် အောက်ပါဂုဏ်သတ္တိနှစ်ခုဖြစ်လာပါတယ်။

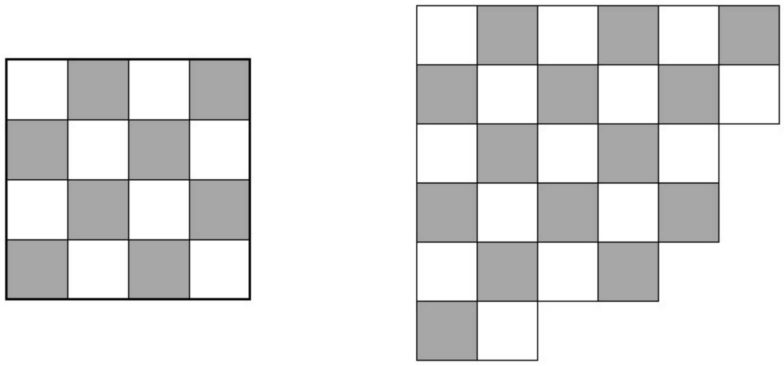
- **အဖြူရောင်အကွက်တစ်ခုမှနေ၍ တစ်ကြိမ်ရွှေ့သော် အနက်ရောင်အကွက်ထဲသို့ ရောက်မည်။**
- **အနက်ရောင်အကွက်တစ်ခုမှနေ၍ တစ်ကြိမ်ရွှေ့သော် အဖြူရောင်အကွက်ထဲသို့ ရောက်မည်။**

ဒါကြောင့် လက်ညိုးရှိတဲ့အကွက်ရဲ့အရောင်သည် တစ်ကြိမ်ရွှေ့လိုက်တိုင်းရွှေ့လိုက်တိုင်း အဖြူမှအနက်၊ အနက်မှအဖြူ တစ်လှည့်စီပြောင်းနေမှာပါ။ ရွှေ့ရမယ့်အကြိမ်အရေအတွက်ကို ကျွန်တော်ကပြောရတာဖြစ်လို့ လက်ညိုးရဲ့တည်နေရာ အတိအကျကို မသိနိုင်ပေမယ့် လက်ညိုးရှိတဲ့ အကွက်ရဲ့အရောင်ကိုတော့ ကျွန်တော်သိနိုင်ပါတယ်။ အကွက်တွေကို ဖျက်တဲ့အခါမှာ လက်ညိုးနဲ့ အရောင်မတူမယ့်အကွက်တွေကိုချည်း ဖျက်သွားရုံပါပဲ။

ဥပမာပြပါမယ်။ စစချင်းမှာ လက်က A ပေါ်မှာရှိပါတယ်။ ကျွန်တော်က သုံးခါရွှေ့ခိုင်းလိုက်တယ် ဆိုပါစို့။ ဒါဆိုရင် လက်ညိုးရှိတဲ့အကွက်ရဲ့ အရောင်သည် မူလအဖြူရောင်ကနေ အနက်၊ အနက်ကနေ အဖြူ၊ အဖြူကနေ အနက်ပြန် ရောက်တဲ့အတွက် အနက်ရောင်ဖြစ်ပါတယ်။ ဒါကြောင့် ကျွန်တော်က အဖြူရောင်အကွက်ဖြစ်တဲ့ A, C, E, G, I ထဲက နှစ်သက်ရာကို ဖျက်ပစ်နိုင်ပါတယ်။ ဒီနေရာမှာ သတိထားရမှာတစ်ခုပဲရှိပါတယ်။ အဲ့တာကတော့ ကျန်ရှိနေတဲ့အကွက်တွေကို တစ်ဆက်တည်းဖြစ်နေစေဖို့ပါပဲ။ ဥပမာ ခုနကဥပမာမှာ A, E, G သုံးကွက်ကိုဖျက်မိသွားရင် လက်ညိုးက အကွက် D ထဲမှာ ပိတ်မိသွားနိုင်တဲ့အတွက် အန္တရာယ်ရှိပါတယ်။ ဒါလေးတစ်ခုပဲ သတိထားရုံပါ။

Bipartite graph များ

ဒီမျက်လှည့်ဟာ 3 × 3 grid တွေတင်မကဘဲ ဘယ်လိုပုံစံရှိတဲ့ board ပေါ်မှာမဆို ပြလို့ရပါတယ်။ ဥပမာအနေနဲ့ အောက်ပါပုံတွေကို ကြည့်ကြည့်ပါ။



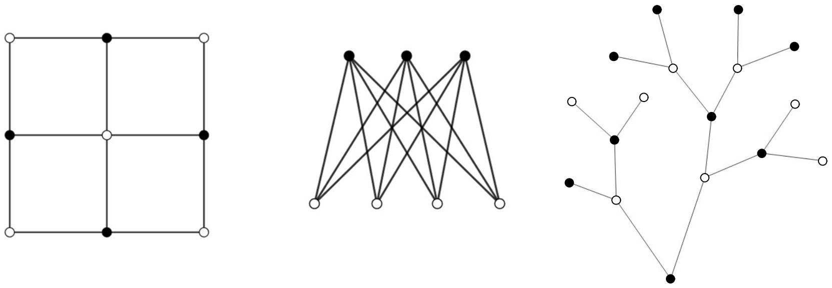
ပုံ 4 – Board အားလုံးကို အဖြူ၊ အမည်း တစ်လှည့်စီခြယ်လို့ရတယ်။

အဓိကက တစ်ဆက်တည်းဖြစ်နေဖို့နဲ့ အကွက်တွေကို အောက်ပါဂုဏ်သတ္တိနဲ့ပြေလည်အောင် အဖြူ၊ အမည်း အရောင်ခြယ်နိုင်ဖို့ပါပဲ။

- **အဖြူရောင်အကွက်ကနေရွှေ့ရင် အနက်ရောင်အကွက်ထဲရောက်တယ် (အဖြူရောင်ထဲ ပြန်မရောက်ဘူး)။**
- **အနက်ရောင်အကွက်ကနေရွှေ့ရင် အဖြူရောင်အကွက်ထဲရောက်တယ် (အနက်ရောင်ထဲ ပြန်မရောက်ဘူး)။**

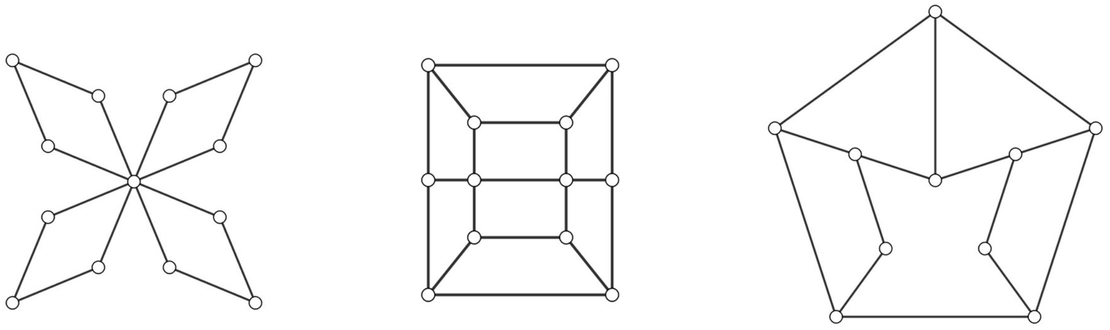
Board တွေပေါ်မှာတင်မကသေးပါဘူး။ ဒီအိုင်ဒီယာကို graph တွေပေါ်အထိ generalize လုပ်လို့ရပါတယ်။

Graph တစ်ခုရဲ့ vertex တွေကို အဖြူ သို့မဟုတ် အနက်နဲ့ အရောင်ခြယ်မယ်လို့ တွေးကြည့်လိုက်ပါ။ တကယ်လို့များ vertex အဖြူအချင်းချင်းရော၊ အနက်အချင်းချင်းပါ ချိတ်ဆက်မထားအောင် ခြယ်လို့ရခဲ့မယ်ဆိုရင် အဲ့ဒီ graph ကို bipartite graph လို့ခေါ်ပါတယ်။ အောက်ဖော်ပြပါ graph တွေဟာ bipartite graph တွေဖြစ်ကြပါတယ်။



ပုံ 5 – Bipartite graph အချို့

မျက်လှည့်ကို တစ်ဆက်တည်းရှိတဲ့ (ဘယ် vertex ကနေမဆို ကြိုက်ရာ vertex ကို edge တွေတလျှောက်ခရီးနှင်လို့ရတဲ့) bipartite graph အားလုံးမှာပြလို့ရတယ်ဆိုတာကို သတိထားမိမှာပါ။ Graph တစ်ခုဟာ bipartite ဖြစ်မဖြစ်ကို ပုံကြည့်ပြီး တန်းပြောနိုင်တဲ့ နည်းတစ်ခုရှိပါတယ်။ ဒါကိုကွက်ကွက်ကွင်းကွင်းမြင်ဖို့အတွက် အောက်ဖော်ပြပါ graph တွေကို bipartite ဖြစ်မဖြစ် စာဖတ်သူကိုယ်တိုင်ဆုံးဖြတ်ကြည့်စေချင်ပါတယ်။ ဖြစ်တယ်ဆိုရင် အရောင်ခြယ်ပြပါ၊ မဖြစ်ဘူးဆိုရင် ဘာကြောင့်မဖြစ်ရတာလဲ ရှင်းပြကြည့်ပါ။



G_1 G_2 G_3

ပုံ 6 – Bipartite ဖြစ်သလား။

ပုံသုံးပုံထဲမှာ G_3 က bipartite မဖြစ်ပါဘူး။ မဖြစ်ရတဲ့ အဓိကအကြောင်းရင်းသည် အပြင်ကပတ်ထားတဲ့ ပဉ္စဂံကြောင့်ပါ။ တကယ်လို့များ bipartite ဖြစ်ခဲ့မယ်ဆိုရင် ဘယ်ပတ်လမ်း (cycle) ကိုကြည့်ကြည့် vertex တွေရဲ့အရောင်သည် အဖြူတစ်လှည့် အနက်တစ်လှည့်ဖြစ်နေရမှာဖြစ်လို့ cycle တိုင်းမှာ vertex အရေအတွက် စုံဂဏန်းသာပါရပါမယ်။ G_2 မှာပါတဲ့ ပဉ္စဂံသည် vertex အရေအတွက် မကိန်းရှိတဲ့ cycle ဖြစ်နေတဲ့အတွက် G_2 ဟာ bipartite မဖြစ်ပါဘူး။ အခုလိုမျိုး vertex အရေအတွက် မဂဏန်းပါတဲ့ cycle တွေကို odd cycle လို့ခေါ်ပြီး စုံဂဏန်းပါတာတွေကိုတော့ even cycle လို့ခေါ်ပါတယ်။ ဒါဆို ဒီ result ကိုရပါပြီ။

If a graph is bipartite, then it does NOT contain any odd cycles.

သူ့ရဲ့ပြောင်းပြန်ကရော မှန်သလား။ ဆိုလိုတာက odd cycle မပါဘူးဆိုတာကိုသိရင် bipartite ဖြစ်တယ်လို့ တန်းပြောနိုင်ပါ့မလား။ အဖြေကိုတော့ ဒီ theorem မှာတွေ့နိုင်ပါတယ်။

Theorem. A graph is bipartite if and only if it does NOT contain any odd cycles.

Proof sketch. If G is bipartite, then it does NOT contain any odd cycles ဆိုတဲ့အပိုင်းက အပေါ်မှာပြောပြီးသွားပါပြီ။ ကျန်တဲ့တစ်ပိုင်းကိုပြဖို့ odd cycle မပါတဲ့ graph တစ်ခု G ကိုယူလိုက်ပါ။ သူ့ရဲ့ vertex တွေကို အဖြူအဖြူချင်းမချိတ်၊ အနက်အနက်ချင်းမချိတ်အောင်လို့ အဖြူနဲ့ အနက်နှစ်ရောင်သုံးပြီး ခြယ်ပြပါမယ်။

အရင်ဆုံး G ကတစ်ဆက်တည်းဖြစ်၊ တစ်နည်းအားဖြင့် ဘယ် vertex နှစ်ခုယူယူ အချင်းချင်း edge တွေတလျှောက် ဖြတ်သန်းပြီးခရီးနှင်လို့ရတဲ့ အခြေအနေမှာ အရင်ခြယ်ကြည့်ပါမယ်။ တစ်ဆက်တည်းမဖြစ်တဲ့ အခြေအနေတွေမှာလည်း အစိတ်အပိုင်းတစ်ခုချင်းစီကို ဒီနည်းလမ်းနဲ့ အရောင်ခြယ်လိုက်ရုံပါပဲ။ ခြယ်ရမယ့်နည်းလမ်း ကတော့ အောက်ပါအတိုင်းပါ။

Step 1: မိမိနှစ်သက်ရာ vertex ကို အဖြူရောင်ခြယ်ပါ။

Step 2: Step 1 မှာအဖြူခြယ်ထားတဲ့ vertex နဲ့ဆက်ထားသော အရောင်မဲ့ vertex များကို အနက်ခြယ်ပါ။

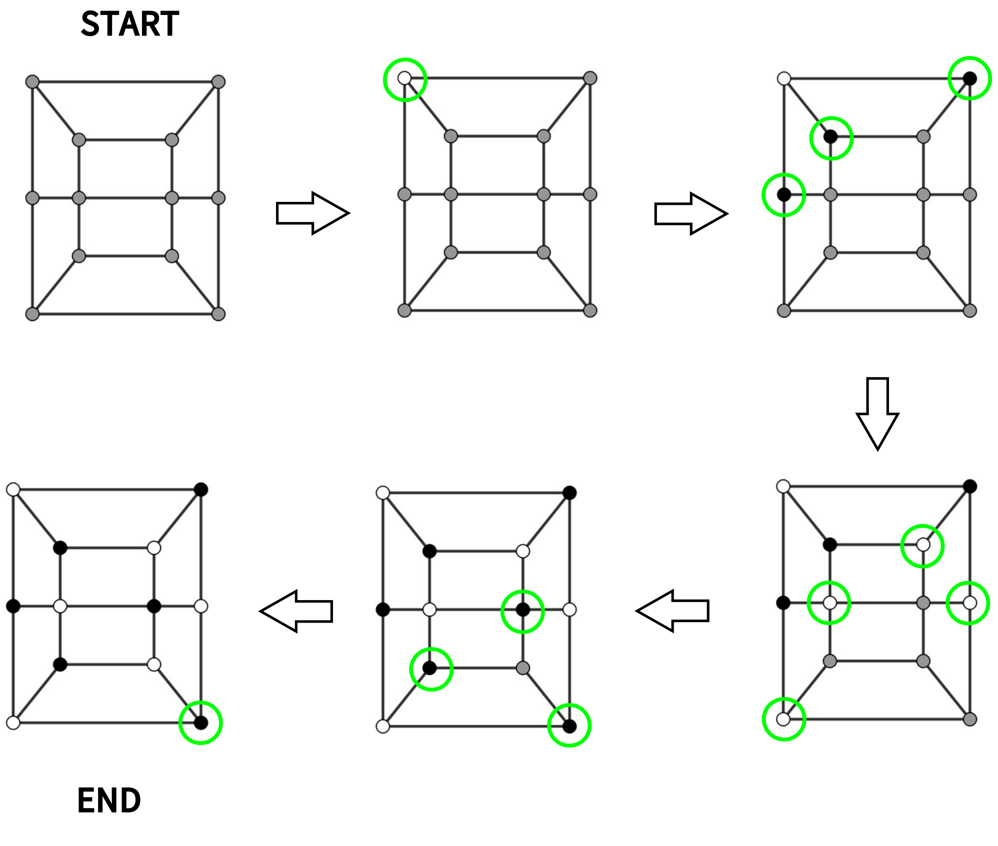
Step 3: Step 2 မှာအနက်ခြယ်ထားတဲ့ vertex နဲ့ဆက်ထားသော အရောင်မဲ့ vertex များကို အဖြူခြယ်ပါ။

Step 4: Step 3 မှာအဖြူခြယ်ထားတဲ့ vertex နဲ့ဆက်ထားသော အရောင်မဲ့ vertex များကို အနက်ခြယ်ပါ။

…

ဒီအတိုင်း ဆက်ခြယ်သွားပြီး နောက်ဆုံးမှာအားလုံးခြယ်ပြီးတဲ့အချိန်ကျ အရောင်တူ vertex နှစ်ခုဆက်မထားတဲ့ graph ထွက်လာမယ်လို့ အာမခံနိုင်ပါတယ်။ ဘာကြောင့်အာမခံနိုင်သလဲဆိုတာကိုတော့ ဒီစာအုပ်မှာ သက်သေမပြတော့ပါဘူး။ အကြမ်းဖြင်းအားဖြင့်တော့ အရောင်ခြယ်တဲ့အဆင့်တစ်ခုခုအရောက်မှာ အကယ်၍ အရောင်တူ vertex နှစ်ခု ဆက်ထားမိမယ်ဆိုရင် graph ထဲမှာ odd cycle ရှိနေမှာဖြစ်ကြောင်း သက်သေပြတာပါ။ စိတ်ဝင်စားရင် ပြကြည့်ပါ။ သိပ်အခက်ကြီးမဟုတ်ပါဘူး။

အရောင်ခြယ်နည်းကို ဥပမာပြတဲ့အနေနဲ့ ပုံ 6 ထဲက G_2 ကိုခြယ်ပြထားပါတယ်။ ပုံမှာ မီးခိုးရောင် vertex တွေသည် အရောင်မခြယ်ရသေးသော vertex တွေကိုရည်ညွှန်းပါတယ်။



ပုံ 7 – အရောင်ခြယ်မယ့်နည်းလမ်း