

Uniwersytet Wrocławski
Wydział Matematyki i Informatyki
Instytut Matematyczny
specjalność: Moduł ogólny

Matylda Rogatty

Analiza pewnego algorytmu kolorującego grafy

Praca licencjacka
napisana pod kierunkiem
dr. Grzegorza Jagielli

Wrocław 2023

Wstęp

W niniejszej pracy przyjrzymy się problemowi kolorowania map z matematycznego punktu widzenia. Mapę z podziałem na regiony, chcemy pokolorować tak, żeby każde dwa sąsiadujące regiony miały inny kolor. Chcemy do tego użyć jak najmniejszej liczby kolorów. Z pomocą przychodzi nam reprezentacja map przy użyciu grafów. Dowiemy się, czym są grafy i jakie mają własności. Za pomocą algorytmu zaproponowanego w pracy [3], pokolorujemy każdy graf reprezentujący mapę, używając tylko pięciu kolorów. Praca przedstawia przepis na algorytm działający w czasie liniowo zależnym od liczby regionów na mapie. Wyjaśnienia w pracy [3] są przedstawione zwięźle, dlatego zajmiemy się opracowaniem teorii kryjącej się za działaniem tego algorytmu. Przedstawimy dowody w sposób przystępny dla niespecjalisty. Wytłumaczymy obliczenia przeprowadzane w pracy i rozważymy inne możliwe modyfikacje algorytmu.

W pierwszym rozdziale wprowadzimy czytelnika do teorii grafów. Przedstawimy podstawowe definicje i twierdzenia. Teoria bazuje na skrypcie [2], skąd również została zaadaptowana terminologia.

Następnie przejdziemy do opisu algorytmu. W drugim rozdziale znajduje się uproszczona wersja algorytmu z pracy [3]. Przeprowadzimy dowód na jej działanie. Algorytm przedstawimy za pomocą pseudokodu. Zakładamy, że czytelnik ma wiedzę z zakresu studiów pierwszego stopnia na kierunku matematyka i zna podstawowe struktury danych.

W trzecim i ostatnim rozdziale przejdziemy już do konkretnego algorytmu z pracy [3]. Opiszemy warunek, który zapewnia, że złożoność czasowa algorytmu jest liniowa. Wyjaśnimy działanie algorytmu i pokażemy przykłady jego pracy.

Dziękuję doktorowi Grzegorzowi Jagielli za poświęcony czas, liczne rady i wskazówki podczas pisania niniejszej pracy. Dziękuję również doktorowi Politechniki Wrocławskiej Andrzejowi Gnatowskiemu za cenne uwagi w sprawie.

Spis treści

1	Preliminaria	4
1.1	Grafy	4
1.2	Problem kolorowania grafów	7
1.3	Twierdzenie o pięciu kolorach	8
2	Algorytm kolorujący	9
2.1	Kroki algorytmu	10
2.2	Dowód, że algorytm działa	11
3	Modyfikacja algorytmu	13
3.1	Złożoność czasowa	19
3.2	Dlaczego $k = 13$?	21
3.2.1	Dolne ograniczenie k	21
3.2.2	Rozważania dla dużych k	21
3.3	Wnioski	24
3.4	Przykłady	24

1 Preliminaria

1.1 Grafy

Grafy reprezentują połączenia między obiektami, zwanymi dalej wierzchołkami. Używane są do opisywania różnych problemów natury matematycznej. Pierwszy raz pojęcie grafu pojawia się w zagadnieniu mostów królewieckich. Problem polegał na pytaniu, czy można spacerować po Królewcu z siedmioma mostami, przechodząc każdy most dokładnie raz. W 1735 roku Leonard Euler, używając wczesnej teorii grafów, dowiódł, że jest to niemożliwe. Teraz teoria grafów jest bardzo rozbudowaną dziedziną matematyki. Używana jest między innymi w informatyce, fizyce, chemii czy naukach społecznych.

Poniżej znajdują się podstawowe pojęcia z teorii grafów. Część z nich jest intuicyjna, lecz dla precyzyjności zostały zdefiniowane wraz z oznaczeniami.

Definicja 1.1. Graf to para dwóch zbiorów $G = (V, E)$, gdzie

- $V = V(G)$ to zbiór wierzchołków,
- $E = E(G) \subseteq \{\{u, v\} : u, v \in V \wedge u \neq v\}$ to zbiór krawędzi.

Grafy nieskończone to grafy z nieskończonym zbiorem wierzchołków. Przy algorytmicznym podejściu do problemu kolorowania grafów rozważamy tylko grafy skończone. Dlatego w ramach konwencji przyjmujemy, że graf to graf skończony. Krawędzie zdefiniowane są jako para nieuporządkowana, ponieważ rozważamy grafy nieskierowane.

Oznaczenia i nazewnictwo: Niech G to graf, $v \in V(G)$, wtedy

- rząd grafu G , oznaczamy $|G|$, to moc zbioru $V(G)$,
- $e(G)$ to moc zbioru $E(G)$,
- sąsiadem v nazywamy wierzchołek, który jest połączony krawędzią z v ,
- $N(v) := \{w : \{v, w\} \in E(G)\}$ to sąsiedztwo v ,
- $d(v)$ to stopień wierzchołka v , czyli liczba jego sąsiadów,
- $d(G) := \frac{\sum_{v \in V(G)} d(v)}{|G|}$ to średni stopień grafu G .

Definicja 1.2. \bar{G} jest podgrafem grafu G , jeśli $V(\bar{G}) \subseteq V(G)$ oraz $E(\bar{G}) \subseteq E(G)$.

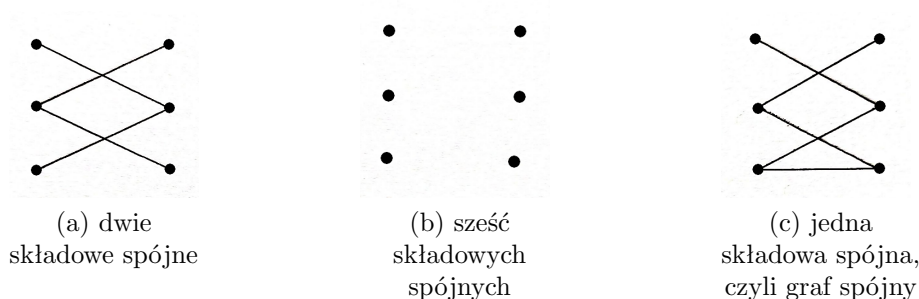
Definicja 1.3. Podgraf $G = (V, E)$ indukowany przez $A \subset V$ to podgraf $G[A] := (A, E_A)$, gdzie $E_A = \{\{v, w\} \in E : v, w \in A\}$.

Definicja 1.4. Ścieżka długości n grafu G to podgraf P grafu G taki, że $V(P) = \{a_1, a_2, a_3, \dots, a_{n+1}\}$ oraz $E(P) = \{\{a_i, a_{i+1}\} : 1 \leq i \leq n\}$, dla $n \geq 1$. Cykl długości n (oznaczamy C) to ścieżka z v do v , dla $v \in V(C)$.

Definicja 1.5. Graf planarny to graf, który można narysować na płaszczyźnie, w taki sposób, aby żadne dwie krawędzie się nie przecinały. Przedstawienie grafu planarnego na płaszczyźnie nazywane jest grafem płaskim. Ściana grafu płaskiego to część płaszczyzny, ograniczona przez krawędzie tego grafu.

Komentarz: Równoważnie graf jest planarny, jeśli da się go opisać na sferze.

Definicja 1.6. Graf G jest spójny, gdy dla każdych dwóch wierzchołków istnieje ścieżka między nimi. Spójna składowa grafu G to spójny podgraf G , który nie zawiera się w żadnym większym spójnym podgrafie G . (Przykłady na rysunku 1).



Rysunek 1

Definicja 1.7. Drzewo to graf spójny, który ma co najmniej jeden wierzchołek i nie zawiera cyklu. Liść to wierzchołek stopnia 1.

Komentarz: Ponieważ rozważamy grafy nieskierowane, to w drzewach nie będziemy wyróżniać korzeni.

Uwaga 1.8. (i) Każde drzewo z co najmniej dwoma wierzchołkami zawiera liść.

(ii) Każde drzewo jest planarne.

(iii) Drzewo rzędu n ma $n - 1$ krawędzi.

Lemat 1.9 (Lemat o uściskach dłoni). Dla grafu G

$$e(G) = \frac{|G|}{2}d(G).$$

Dowód. Niech $H = \{(e, v) : v \in V(G), e \in E(G), v \in e\}$. Wtedy

$$|H| = \sum_{e \in E(G)} |\{v : v \in V(G) \wedge v \in e\}| = \sum_{e \in E(G)} 2 = 2e(G).$$

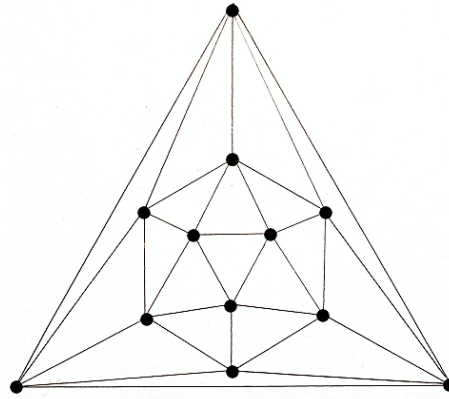
Z drugiej strony

$$|H| = \sum_{v \in V(G)} |\{e : e \in E(G) \wedge v \in e\}| = \sum_{v \in V(G)} d(v) = |G| \cdot d(G).$$

Więc $e(G) = \frac{|G|}{2}d(G)$. □

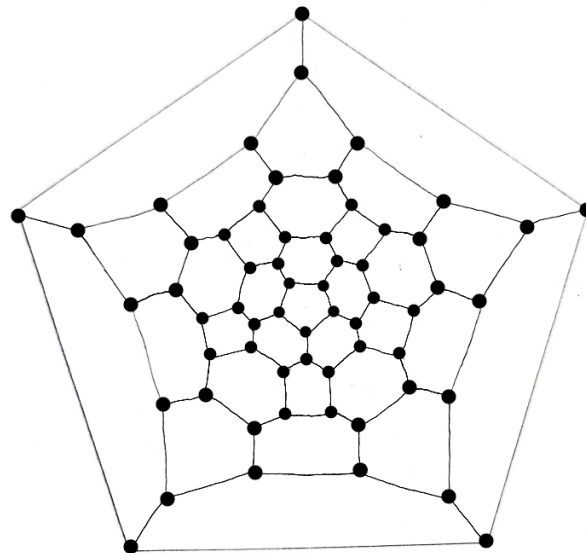
Poniżej znajdują się przykłady dwóch grafów, które mają specyficzne własności. Będą pomocne przy późniejszych rozważaniach.

Rysunek 2 przedstawia dwudziestościan, który w teorii grafów jest spójnym grafem planarnym rzędu 12. Istotne jest to, że wszystkie jego wierzchołki mają stopień 5.



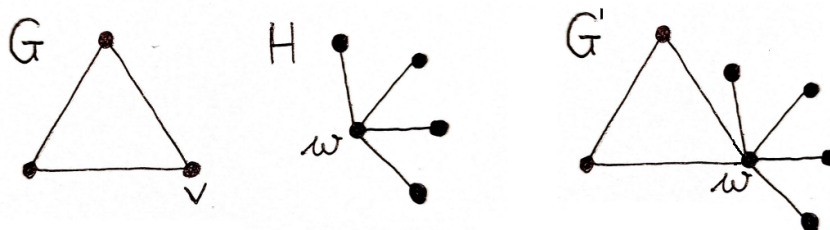
Rysunek 2: Dwudziestościan jako graf płaski

Rysunek 3 przedstawia dwudziestościan ścięty (piłkę nożną). Powstaje przez ścięcie wierzchołków dwudziestościanu foremnego. Ściany otrzymanego grafu to pięciokąty lub sześciokąty. Jest grafem spójnym i planarnym, a jego sześćdziesiąt wierzchołków ma stopień równy 3.



Rysunek 3: Dwudziestościan ścięty jako graf płaski

Definicja 1.10. Niech G, H to grafy i $v \in V(G), w \in V(H)$. Sklejenie grafów G i H względem wierzchołków v i w to graf G' taki, że $V(G') = V(G) \cup V(H) \setminus \{v\}$, $E(G') = E(G) \setminus \{\{v, x\} : x \in V(G)\} \cup E(H) \cup \{\{w, x\} : x \in V(G)\}$. (Rysunek 4 przedstawia przykład sklejenia grafów przy tych samych oznaczeniach).

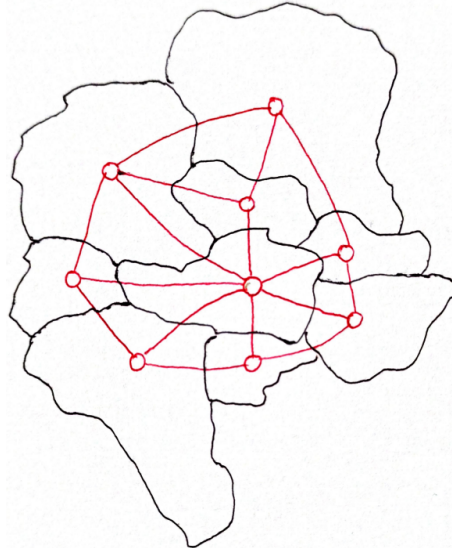


Rysunek 4

Komentarz: Sklejenie grafów to sposób na uzyskanie nowych grafów z wierzchołkami o wyższych stopniach.

1.2 Problem kolorowania grafów

O kolorowaniu grafów można myśleć jak o kolorowaniu map z podziałem na państwa. Chcemy, aby graniczące ze sobą państwa miały różne kolory. Mapę reprezentujemy jako graf planarny. Każdy region jest oznaczany wierzchołkiem, a granicy między nimi odpowiada krawędź, która je łączy.



Rysunek 5

Definicja 1.11. (i) Kolorowanie grafu G to funkcja $\phi : V(G) \longrightarrow K$, gdzie K to zbiór kolorów.

(ii) Rezultatem ϕ jest podział $V(G)$ na parami rozłączne podzbiory, sumujące się do całego zbioru $V(G)$, zwany pokolorowaniem.

(iii) Poprawne pokolorowanie grafu G jest wtedy, gdy funkcja ϕ spełnia warunek

$$\forall_{u,v \in V(G), u \neq v} (\phi(u) = \phi(v) \Rightarrow \{u, v\} \notin E(G)).$$

(iv) Liczba chromatyczna $\chi(G)$ to najmniejsza liczba kolorów potrzebnych do prawidłowego pokolorowania grafu G . Mówimy, że graf G jest k -dzielny wtedy i tylko wtedy, gdy $\chi(G) \leq k$.

Twierdzenie 1.12 (Wzór Eulera dla grafów płaskich). Niech G będzie spójny i płaski, niech $|G| = n$, $e(G) = m$ oraz liczba ścian wynosi l . Wtedy $n - m + l = 2$.

Dowód z [2]. Przeprowadzimy dowód przez indukcję względem liczby cykli w G .

Przypadek 1. G nie ma cykli, czyli jest drzewem. Wtedy $m = n - 1$ i każde przedstawienie G na płaszczyźnie ma jedną ścianę, $l = 1$. Czyli $n - m + l = n - (n - 1) + 1 = 2$.

Przypadek 2. G ma cykl. Niech $e \in E(G)$ będzie krawędzią w tym cyklu. Wtedy $H := (V(G), E(G) \setminus \{e\})$ jest spójny i planarny, mamy $|H| = n$ i $e(H) = m - 1$. Czyli krawędź e dzieli jedną ścianę z H na dwie, co oznacza, że H ma przedstawienie na płaszczyźnie z $l - 1$ ścianami.

Z założenia indukcyjnego dostajemy $n - (m - 1) + (l - 1) = 2$. Stąd wynika, że $n - m + l = 2$. \square

Twierdzenie 1.13. Niech G jest planarny i $|G| = n \geq 3$. Wtedy $e(G) \leq 3n - 6$.

Dowód z [2]. Załóżmy, że G jest grafem spójnym, $|G| = n$, $e(G) = m$, a przedstawienie G na płaszczyźnie ma l ścian.

Dla $n = 3$, graf G jest cyklem długości 3 lub ścieżką długości 2. Czyli $e(G) = 3$ lub $e(G) = 2$. Załóżmy więc, bez straty ogólności, że $n \geq 4$. Ze wzoru Eulera mamy $n - m + l = 2$.

Każda krawędź należy do co najwyżej dwóch ścian i każda ściana ma co najmniej trzy krawędzie, stąd:

$$l \leq \frac{2m}{3}.$$

Czyli

$$m = 3 \left(m - \frac{2m}{3} \right) \leq 3(m - l) = 3(n - 2) = 3n - 6. \quad \square$$

1.3 Twierdzenie o pięciu kolorach

Percy Heawood w 1890 roku przedstawił dowód na to, że do poprawnego pokolorowania grafu planarnego wystarczy użyć pięciu kolorów. Istnieje mocniejsze twierdzenie, mówiące, że wystarczy cztery kolory. Dowód został przeprowadzony częściowo za pomocą komputerów, dlatego nie wszyscy go uznają, gdyż ciężko sprawdzić, czy obliczenia są poprawne. Poniżej przeprowadzimy dowód dla twierdzenia o pięciu kolorach.

Twierdzenie 1.14 (Twierdzenie o pięciu kolorach). *Jeśli graf G jest planarny, to $\chi(G) \leq 5$.*

Dowód z [2]. Niech G to graf planarny i $|G| = n$. Przeprowadzimy dowód przez indukcję względem n .

Gdy $n \leq 5$, teza jest oczywista. Więc załóżmy, że $n \geq 6$.

Ponieważ G jest planarny, z Twierdzenia 1.13 mamy, że

$$e(G) \leq 3n - 6,$$

oraz z Lematu 1.9

$$e(G) = \frac{n}{2}d(G).$$

Stąd

$$d(G) = \frac{2e(G)}{n} \leq \frac{2(3n - 6)}{n} = \frac{6n - 12}{n} = 6 - \frac{12}{n} < 6.$$

Wnioskujemy, że

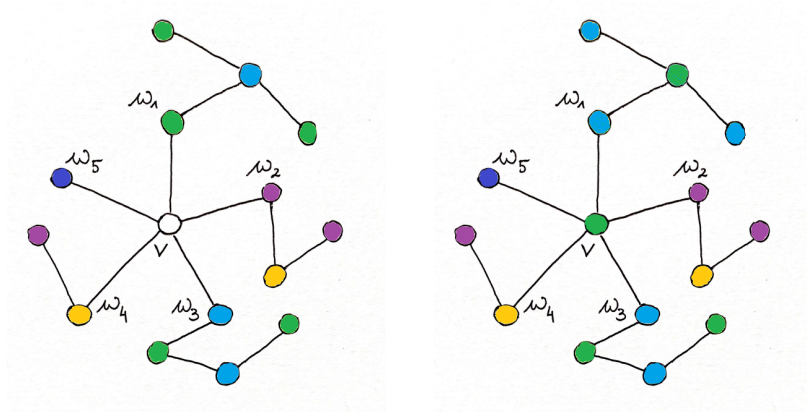
$$\exists_{v \in V(G)} d(v) \leq 5.$$

Ustalmy, że $H := G[V(G) \setminus \{v\}]$, z założenia indukcyjnego $\phi : V(H) \rightarrow \{1, 2, 3, 4, 5\}$ jest poprawnym kolorowaniem H , gdzie kolory wyrażone są za pomocą liczb naturalnych. Pokażemy, że jeśli dodamy v do H , nadal będzie istniało poprawne pokolorowanie grafu.

Jeśli $\phi[N_G(v)] \neq \{1, 2, 3, 4, 5\}$, wtedy możemy rozszerzyć ϕ do poprawnego kolorowania G używającego pięciu kolorów. Czyli możemy założyć, że $\phi[N_G(v)] = \{1, 2, 3, 4, 5\}$. Więc $d(v) = 5$ wtedy i tylko wtedy, gdy każdy sąsiad v jest innego koloru.

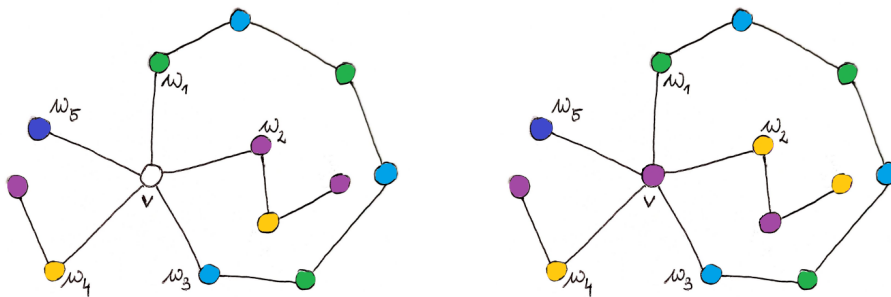
Niech $N_G(v) = \{w_1, w_2, w_3, w_4, w_5\}$. Rozważmy rysunek przedstawiający G , na którym numerujemy sąsiady v kolejno od w_1 do w_5 , zgodnie z ruchem wskazówek zegara. Załóżmy, że $\phi(w_i) = i$, dla $i \in \{1, 2, 3, 4, 5\}$. Dla każdego koloru i rozważmy $V_i := \{u \in V(H) : \phi(u) = i\}$ oraz dla każdych $i \neq j$ rozważmy dwukolorowy podgraf $H_{ij} := H[V_i \cup V_j]$.

Jeśli w_1 i w_3 są w różnych spójnych składowych H_{13} , możemy zdefiniować poprawne kolorowanie $\phi' : G \rightarrow \{1, 2, 3, 4, 5\}$ takie, że zamieniamy kolory 1 i 3 w spójnej składowej H_{13} zawierającej w_1 , następnie ustawić $\phi'(v) = 1$. Przykładową sytuację przedstawia rysunek 6.



Rysunek 6: w_1 i w_3 w różnych spójnych składowych H_{13}

W przeciwnym razie w_2 i w_4 są w różnych ścianach rysunku ograniczonego do $G[\{v\} \cup V(H_{13})]$, czyli muszą być w różnych spójnych składowych H_{24} . Więc ponownie możemy zdefiniować $\phi' : G \rightarrow \{1, 2, 3, 4, 5\}$ zamieniając kolory 2 i 4 w spójnej składowej H_{24} zawierającej w_2 i ustawić $\phi'(v) = 2$. Taką sytuację przedstawia rysunek 7.



Rysunek 7: w_1 i w_3 w jednej spójnej składowej H_{13}

□

W obu przypadkach mogliśmy zamienić kolory sąsiadów v w taki sposób, aby otrzymać poprawne pokolorowanie grafu przy użyciu pięciu kolorów.

2 Algorytm kolorujący

Głównym celem jest zrozumienie algorytmu z pracy [3]. Dla lepszego zrozumienia najpierw przedstawimy uproszczoną wersję tego algorytmu, a w trzecim rozdziale przejdziemy do pierwotnej wersji. Uproszczenie polega na pominięciu jednego warunku, który zapewnia, że czas działania algorytmu jest liniowy.

Niniejszy algorytm polega na sukcesywnym usuwaniu wierzchołków o stopniach nie większych niż 5. Usuwanie jest przerywane w momencie, gdy pozostanie co najwyżej 5 wierzchołków. Wtedy są one kolorowane, każdy na inny kolor. Ostatni krok to przywracanie usuniętych wierzchołków i odpowiednie kolorowanie ich jeden po drugim.

Podczas redukcji grafu, czyli zmniejszania liczby wierzchołków i krawędzi, stosowane są dwie metody, w zależności od stopnia usuwanego wierzchołka. Gdy wierzchołek v grafu G ma stopień mniejszy niż 5, wykonywana jest *redukcja pierwszego typu*, czyli usunięcie z grafu wierzchołka v . Jeśli wierzchołek v ma stopień równy 5 stosowana jest *redukcja drugiego typu*. Polega ona na wybraniu dwóch niesąsiadujących ze sobą sąsiadów v i zidentyfikowaniu ich oraz na usunięciu v .

Dokładniej, jeśli u i w to sąsiady v , które nie mają krawędzi między sobą, to do w przypisane są sąsiady u , a następnie u i v są usuwane. Metody usunięcia i identyfikacji wyrażają następujące funkcje.

Definicja 2.1. Usunięcie wierzchołka v z grafu G jest wyrażone funkcją ζ taką, że

$$\zeta(G, v) = (V(G) \setminus \{v\}, E(G) \setminus \{\{x, v\} : x \in V(G)\}).$$

Definicja 2.2. Identyfikacja wierzchołków u z w w grafie G jest wyrażone funkcją η taką, że

$$\eta(G, u, w) = \zeta((V(G), E(G) \cup \{\{w, x\} : x \in N(u)\}), u).$$

W podrozdziale 3.1 powyższe metody zostaną szczegółowo opisane przy pomocy pseudokodu.

2.1 Kroki algorytmu

Niech G to graf planarny, który będziemy kolorować. Zakładamy, że wierzchołki są ponumerowane i sortujemy je rosnąco. Podczas inicjalizacji algorytmu wykonywana jest iteracja po wierzchołkach G i tworzone są obiekty, które będą używane w trakcie kolorowania.

- L - lista połączeń, gdzie $L[v] = N(v)$,
- Q_4 - kolejka wierzchołków $v \in G$, takich, że $d(v) \leq 4$,
- Q_5 - kolejka wierzchołków $v \in G$, takich, że $d(v) = 5$,
- D - lista stopni wierzchołków,
- S - pusty stos, do którego będą dodawane pary (v, p) , gdzie v to wierzchołek, który został usunięty, a p to:
 - lista sąsiadów wierzchołka v w momencie usuwania v , jeśli wierzchołek był usunięty za pomocą redukcji pierwszego typu, lub
 - wierzchołek u , z którym v był zidentyfikowany, jeśli wierzchołek był usunięty za pomocą redukcji drugiego typu.

$L[v]$, Q_4 i Q_5 to kolekcje wierzchołków ze stałym czasem dopisywania i usuwania z nich elementów, również ze środka oraz z możliwością sprawdzenia w czasie stałym, czy dany element się w nich znajduje, na przykład listy podwójnie łączone.

Pierwszym krokiem algorytmu jest redukcja grafu (Algorytm 1). Priorytet w usuwaniu wierzchołków zawsze dostają wierzchołki z kolejki Q_4 . Jeśli Q_4 jest niepuste, kolejno usuwane są jego elementy. Z każdym usunięciem wierzchołka v z Q_4 , na stos S trafiają pary $(v, N(v))$. Następnie, jeśli Q_4 jest puste, a w Q_5 pozostały jakieś wierzchołki, wyciągany jest pierwszy z nich, powiedzmy w i wykonywane jest usunięcie w i identyfikacja jego dwóch sąsiadów, a do stosu S dodana jest para $(w, N(w))$ i para zidentyfikowanych wierzchołków. Podczas usuwania wierzchołków, stopnie ich sąsiadów zmieniają się, więc w trakcie jednej pętli algorytmu zawartości do Q_4 mogą trafić nowe wierzchołki, wtedy znowu one dostają priorytet.

Algorytm 1 Redukcja grafu

```
while  $|G| > 5$  do
  if  $Q_4 \neq \emptyset$  then
     $v \leftarrow$  pierwszy element z kolejki  $Q_4$ 
     $L_v \leftarrow$  kopia  $L[v]$ 
     $G \leftarrow \zeta(G, v)$  ▷ redukcja pierwszego typu
    dodaj  $(v, L_v)$  na stos  $S$ 
  else
     $v \leftarrow$  pierwszy element z kolejki  $Q_5$ 
     $u, w \leftarrow$  sąsiady  $v$  takie, że  $u \neq w \wedge u \notin N(w)$ 
     $L_v \leftarrow$  kopia  $L[v]$ 
     $G \leftarrow \zeta(G, v)$  ▷ redukcja —————
     $G \leftarrow \eta(u, w)$  ▷ ————— drugiego typu
    dodaj  $(v, L_v)$  na stos  $S$ 
    dodaj  $(u, w)$  na stos  $S$ 
  end if
end while
```

Redukcja jest przerywana, gdy pozostanie maksymalnie 5 wierzchołków. Wtedy algorytm zaczyna kolorowanie (Algorytm 2). Do nieusuniętych wierzchołków przypisane jest poprawne pokolorowanie (każdy wierzchołek otrzymuje inny kolor). Następnie wyciągane są elementy ze stosu S :

- jeśli element składa się z wierzchołka v i kolekcji sąsiadów $L[v]$, to przypisany jest mu kolor inny niż te, które zostały użyte dla elementów w $L[v]$,
- jeśli para składa się z dwóch wierzchołków v i u , to do v zostaje przypisany ten sam kolor, który był przypisany do u .

Algorytm 2 Kolorowanie

```
przypisz kolory do wierzchołków pozostałych w  $G$ 
while  $S \neq \emptyset$  do
   $(x, y) \leftarrow$  pierwszy element ze stosu  $S$ 
  if  $y$  jest listą then
     $\phi(x) \leftarrow$  inny kolor niż te użyte dla elementów w  $y$ 
  else
     $\phi(x) \leftarrow \phi(y)$ 
  end if
end while
```

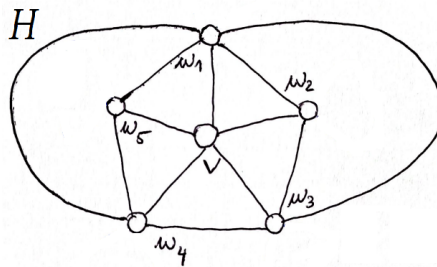
Kiedy S jest puste, algorytm się zatrzymuje.

2.2 Dowód, że algorytm działa

Algorytm przypomina dowód Twierdzenia 1.3. Zobaczmy, że dzięki rekurencyjnemu działaniu będziemy mieć pewność, że otrzymane kolorowanie będzie dozwolone. W odróżnieniu od sposobu kolorowania w wyżej wspomnianym dowodzie algorytm nie przekolorowuje wierzchołków. Podczas redukcji grafu algorytm zapamiętuje połączenia, a następnie rekurencyjnie wraca do pierwotnego grafu, kolorując go odpowiednio.

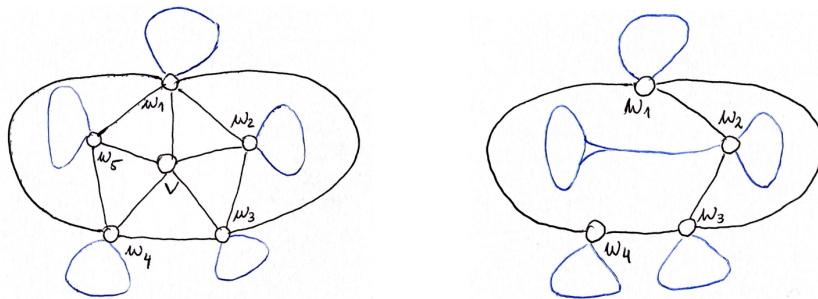
Pokażemy, że algorytm pokoloruje każdy graf planarny. Ponieważ zaczyna od redukcji, a następnie koloruje wierzchołki, udowodnimy działanie każdej z tych dwóch metod. Najpierw zajmijmy się redukcją grafu. Dowód przeprowadzimy za pomocą indukcji względem liczby wierzchołków. Niech G to graf planarny rzędu $n > 5$. Ponieważ G jest planarny to $d(G) < 6$, czyli istnieje wierzchołek $v \in G$ taki, że $d(v) \leq 5$. Mamy dwa przypadki.

- (i) Gdy $d(v) = 5$, rozważmy sytuację przedstawioną na rysunku 8 i przyjmijmy oznaczenia z tego rysunku. H jest podgrafem G , takim, że ma wierzchołek stopnia 5 i możliwie najwięcej krawędzi.



Rysunek 8

Możemy zidentyfikować trzy pary wierzchołków (w_2, w_4) , (w_3, w_5) i (w_2, w_5) . Bez straty ogólności zidentyfikujemy w_5 z w_2 . Po redukcji, czyli usunięciu v z G i identyfikacji w_5 z w_2 , stopnie sąsiadów v spadają o jeden. Rysunek 2 przedstawia G przed i po redukcji dla wierzchołka v . Niebieskie pętelki to nieformalne przedstawienie podgrafów G zawartych w odpowiednich ścianach, zawierających wierzchołki w_i , $i \in \{1, 2, \dots, 5\}$.



Rysunek 9

Otrzymujemy graf planarny rzędu $n - 2$. Ponownie musi istnieć wierzchołek stopnia co najwyżej pięć. Więc powtarzamy krok jak wyżej, jeśli wierzchołek ma stopień pięć lub przechodzimy do przypadku (ii) jeśli ma stopień mniejszy niż pięć. Indukcyjnie dochodzimy do sytuacji, gdy mamy pięć lub mniej wierzchołków.

- (ii) Gdy $d(v) \leq 4$, to wierzchołek zostaje usunięty, bez dodatkowych zabiegów. Z założenia indukcyjnego wiemy, że $G[V(G) \setminus \{v\}]$ będzie zredukowany do momentu, gdy zostanie 5 lub mniej wierzchołków.

Podobnie pokażemy poprawność kolorowania grafu tym algorytmem. Niech $|G| > 5$ i $v \in V(G)$. Przypuśćmy, że graf został zredukowany i teraz rekurencyjnie wracamy, dodając wierzchołki i przypisując im kolory. Wystarczy pokazać działanie dla jednego kroku algorytmu. Dla grafów rzędu nie większego niż 5 algorytm przypisuje każdemu wierzchołkowi inny kolor, czyli poprawne pokolorowanie. Przywracamy wierzchołek v do zredukowanego grafu kolorujemy go odpowiednio. Rozważamy dwie sytuacje.

- (i) Jeśli $d(v) = 5$, z założenia indukcyjnego wiemy, że $G[V(G) \setminus \{v, u\}]$ ma przypisane poprawne pokolorowanie, gdzie u i w to sąsiady v , które zostały zidentyfikowane podczas redukcji na wierzchołku v . G również ma poprawne pokolorowanie, ponieważ algorytm wierzchołkowi u przypisuje ten sam kolor co wierzchołkowi w , wtedy $|\phi[N(v)]| \leq 4$. Następnie v zostaje przypisany kolor inny niż te użyte dla jego sąsiadów.
- (ii) Jeśli $d(v) \leq 4$, z założenia indukcyjnego $G[V(G) \setminus \{v\}]$ ma przypisane poprawne kolorowanie. Do v zostaje przypisany inny kolor niż te użyte dla jego sąsiadów, więc G również ma poprawne pokolorowanie.

Stąd zawsze otrzymujemy poprawne pokolorowanie.

3 Modyfikacja algorytmu

Zauważmy, że stopnie wierzchołków podlegających identyfikacji nie są ograniczone, co powoduje, że nie mamy stałego ograniczenia na czas działania pojedynczej identyfikacji. Liczba wykonywanych identyfikacji jest liniowo zależna od liczby wierzchołków w grafie, więc algorytm niekoniecznie będzie działał w czasie liniowym, na czym nam zależy. (Dokładna analiza złożoności czasowej algorytmu zostanie przeprowadzona w podrozdziale 3.1.) Prostą modyfikacją zapewnimy liniową pracę algorytmu. Modyfikacja polega na ograniczeniu przez liczbę całkowitą k stopnia wierzchołków, które będą identyfikowane. W pracy [3] autor przyjmuje $k = 13$. Do łatwiejszego opisu algorytmu wprowadźmy pojęcie wierzchołka k -niezablokowanego.

Definicja 3.1. Dla $k \in \mathbb{Z}^+ \cup \{\infty\}$, wierzchołek v o stopniu 5 jest k -niezablokowany, jeśli

$$\exists_{u \neq w \in N(v)} (u \notin N(w) \wedge d(u) < k \wedge d(w) < k).$$

Wierzchołek jest k -zablokowany, gdy nie jest k -niezablokowany.

Zmodyfikowany algorytm może usunąć wierzchołek v o stopniu 5 i zidentyfikować jego dwa sąsiady tylko wtedy, gdy jest k -niezablokowany. Jeśli wierzchołek jest k -zablokowany zostaje odłożony na koniec kolejki Q_5 i „czeka”, aż usunięcie innego wierzchołka, odblokuje go. Teraz algorytm redukcji grafu wygląda następująco (Algorytm 3):

Algorytm 3 Redukcja grafu z dodatkowym warunkiem

```

while  $|G| > 5$  do
  if  $Q_4 \neq \emptyset$  then
     $v \leftarrow$  pierwszy element z kolejki  $Q_4$ 
     $G \leftarrow \zeta(G, v)$ 
  else
     $v \leftarrow$  pierwszy element z kolejki  $Q_5$ 
    if  $\exists_{u \neq w \in N(v)} (u \notin N(w) \wedge d(u) < k \wedge d(w) < k)$  then ▷ Nowy warunek!
       $G \leftarrow \zeta(G, v)$ 
       $G \leftarrow \eta(u, w)$ 
    else
      zwróć  $v$  na koniec kolejki  $Q_5$  ▷  $v$  jest  $k$ -zablokowany
    end if
  end if
end while

```

Uwaga 3.2. (i) Przy ustalonym k będziemy mówić, że wierzchołek jest niezablokowany, rozumiejąc, że jest k -niezablokowany.

- (ii) Zauważmy, że w algorytmie z poprzedniego rozdziału moglibyśmy mówić, że warunek konieczny do usunięcia wierzchołka v o stopniu 5 jest równoważny, do warunku na to, że v jest ∞ -niezablokowany.
- (iii) Jeśli graf planarny zawiera wierzchołki o stopniach mniejszych niż 5, algorytm upraszcza się do redukcji pierwszego typu i żaden wierzchołek nie będzie zablokowany. Dlatego głównie rozważać będziemy sytuacje, gdy grafy planarne mają wszystkie wierzchołki stopnia co najmniej 5.

Zauważmy, że k musi być większe niż 1, inaczej algorytm szukałby bez końca wierzchołków do redukcji. Autor w pracy [3] przyjął $k = 13$. Czy jest to wystarczająco duża liczba? Czy w każdym grafie, takim jak w punkcie (iii) Uwagi 3.2 istnieją wierzchołki 13-niezablokowane?

Lemat 3.3. *W grafie planarnym zawsze istnieje wierzchołek 13-niezablokowany.*

Dowód. Rozważmy pesymistyczny przypadek. Niech graf G ma m wierzchołków stopnia 5 i każdy z nich jest k -zablokowany, więc ma dwa sąsiady o stopniu k . Zakładamy, że wierzchołki stopnia k są połączone tylko z wierzchołkami stopnia 5. Stąd wierzchołków o stopniu k będzie $\lceil \frac{2m}{k} \rceil$. Z Twierdzenia 1.9 i 1.13 wnioskujemy, że $d(G) < 6$. Stąd, gdy $k \mid 2m$

$$d(G) = \frac{1}{|G|} \sum d(v) = \frac{5m + k \frac{2m}{k}}{m + \frac{2m}{k}} = \frac{7mk}{(k+2)m} = \frac{7k}{k+2},$$

$$\frac{7k}{k+2} < 6 \iff k < 12.$$

W przypadku, gdy $k \nmid 2m$, niech $\alpha \in [0, 1)$. Wtedy liczba wierzchołków stopnia k wynosi $\frac{2m}{k} + \alpha$ i powtarzamy rozważania.

$$d(G) = \frac{5m + k \left(\frac{2m}{k} + \alpha \right)}{m + \left(\frac{2m}{k} + \alpha \right)} < 6,$$

$$5m + k \left(\frac{2m}{k} + \alpha \right) < 6m + 6 \left(\frac{2m}{k} + \alpha \right),$$

$$2m + k\alpha < m + \frac{12m}{k} + 6\alpha,$$

$$m < \frac{12m}{k} + (6 - k)\alpha.$$

Sprawdzamy, czy k może być większe, niż 13, wtedy na pewno $(6 - k)\alpha < 0$. Więc

$$m < \frac{12m}{k} + (6 - k)\alpha < \frac{12m}{k},$$

$$1 < \frac{12}{k}.$$

Otrzymujemy

$$k < 12.$$

□

To pokazuje, że przyjmując $k = 13$ algorytm skończy pracę, bo zawsze będą istniały dozwolone do redukcji wierzchołki.

Teraz przedstawimy lekko zmodyfikowane Twierdzenie 3 z pracy [3] oraz lematy wykorzystane w dowodzie. Różnice, jakie się pojawiły w dowodzie, opiszemy na koniec. Twierdzenie

zapewnia nas, że algorytm będzie działał w czasie liniowym zależnym od liczby wierzchołków w grafie. Kluczowe do śledzenia dowodu jest zrozumienie poniższego pojęcia etapu. Przypomnijmy, że Q_4 to kolejka wierzchołków o stopniach mniejszych niż 5, a O_5 to kolejka wierzchołków o stopniach równych 5. Cały czas $k = 13$.

Definicja 3.4. *Etapem będziemy nazywać kroki algorytmu od chwili, gdy $Q_4 = \emptyset$ i $Q_5 \neq \emptyset$, do momentu, gdy $Q_4 = \emptyset$ i wszystkie elementy z Q_5 na początku etapu zostały usunięte lub zablokowane. Czyli na koniec etapu Q_5 składa się tylko z zablokowanych wierzchołków lub nowo dodanych.*

Lemat 3.5 (Górne ograniczenie liczby zablokowanych wierzchołków). *Jeśli na początku etapu graf G ma n wierzchołków o stopniach większych niż 4 oraz podczas etapu wykonane zostało e redukcji pierwszego typu i d redukcji drugiego typu, to napotkane zostało co najwyżej*

$$\sum_{i=k} \frac{in_i}{2} + \frac{(2k-4)d}{2}$$

zablokowanych wierzchołków.

Dowód. Niech B oznacza liczbę zablokowanych wierzchołków. Na początku graf G zawiera n_i wierzchołków stopnia i ($i \geq 5$), $|G| = \sum_{i=5} n_i$. Rozważmy wierzchołek v stopnia 5. W skrajnie możliwym przypadku każdy węzeł o stopniu i ($i \geq k$) ma wszystkie sąsiady zablokowane. Z drugiej strony, każdy zablokowany wierzchołek ma co najmniej 2 sąsiady stopnia co najmniej k . Więc

$$B \leq \sum_{i=k} \frac{in_i}{2}.$$

W trakcie etapu, przy identyfikacji stopień jednego z dwóch wierzchołków może się zwiększyć. Tym samym mogą powstać „nowe” wierzchołki o stopniu większym niż k . Powstanie co najwyżej d nowych wierzchołków o stopniu większym niż k .

Rozważmy wierzchołek v , który jest poddany redukcji drugiego typu. To znaczy, że ma dwa sąsiady u i w takie, że

$$d(u) \leq k-1 \quad \wedge \quad d(w) \leq k-1.$$

Po usunięciu v

$$d(u) \leq k-1-1 = k-2 \quad \wedge \quad d(w) \leq k-1-1 = k-2.$$

Następnie po wykonaniu identyfikacji u z w , w „otrzyma” sąsiady u . Czyli

$$d(w) \leq (k-2) + (k-2) = 2k-4.$$

Może powstać co najwyżej d wierzchołków stopnia co najwyżej $2k-4$. Więc liczba zablokowanych wierzchołków może wzrosnąć o co najwyżej $\frac{(2k-4)d}{2}$. Zatem należy dodać tę liczbę do oszacowania

$$B \leq \sum_{i=k} \frac{in_i}{2} + \frac{(2k-4)d}{2}.$$

□

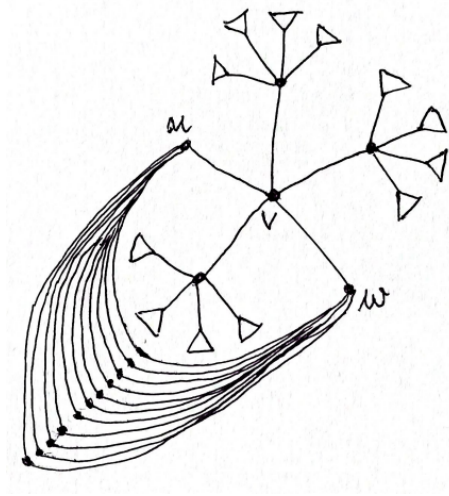
Twierdzenie 3.6. *Jeśli na początku etapu graf G ma n wierzchołków o stopniach większych niż 4 oraz podczas etapu wykonane zostało R redukcji i napotkane zostało B zablokowanych wierzchołków, to*

$$\frac{B}{R} < 362.$$

Dowód. Załóżmy, że podczas etapu wykonywane jest e redukcji pierwszego typu i d redukcji drugiego typu.

Dla każdej redukcji pierwszego typu usuwany jest jeden wierzchołek stopnia co najwyżej 4, czyli stopnie jego co najwyżej czterech sąsiadów zmniejszą się o jeden. Czyli z Q_5 zostanie usunięte od 0 do 4 wierzchołków.

Dla każdej redukcji drugiego typu usuwany jest jeden wierzchołek stopnia pięć i stopnie jego pięciu sąsiadów spadają o jeden (możliwe, że też miały stopień 5). Zauważmy, że wierzchołki, które zostały zidentyfikowane, mogły mieć wspólne sąsiady. Wtedy stopień tych sąsiadów spadnie o jeden. Pomocniczy Rysunek 10 przedstawia skrajną sytuację. Żeby zapewnić, że wszystkie wierzchołki mają stopień co najmniej pięć, sklejamy graf z dwudziestocianami względem niektórych wierzchołków, na rysunku są oznaczone przez trójkąty. Przyjmijmy oznaczenia z Rysunku 10. Załóżmy, że u i w mają maksymalny stopień, czyli 12, a ich wspólne sąsiady mają stopień 5 (czego nie zaznaczamy na rysunku). Przeprowadzamy redukcję na v , czyli usuwamy v i identyfikujemy u z w .



Rysunek 10

Trzem sąsiadom v stopień zmniejszy się z 5 na 4, tak samo dla 11 sąsiadów w . Stąd z Q_5 zostanie usunięte od 1 do 15 wierzchołków.

Zatem z Q_5 podczas jednego etapu zostanie usuniętych $fd+ge$ wierzchołków, dla $1 \leq f \leq 15$ i $0 \leq g \leq 4$. Wszystkie inne węzły w Q_5 muszą być zablokowane, więc

$$B = |Q_5| - fd - ge.$$

Rozważmy maksymalny graf planarny taki, że każda jego ściana jest ograniczona trzema krawędziami. Liczba wierzchołków stopnia 5 wynosi

$$n_5 = 12 + \sum_{i=1} in_{i+6}.$$

Jeśli graf nie jest maksymalny na początku etapu, można go rozszerzyć do grafu maksymalnego planarnego, przez dodanie m krawędzi. Wtedy

$$n_5 = 12 + 2m + \sum_{i=1} in_{i+6}.$$

Stąd

$$B = 12 + 2m + \sum_{i=1} in_{i+6} - fd - ge. \quad (1)$$

Z Lematu 3.5 mamy

$$B \leq \sum_{i=k} \frac{in_i}{2} + \frac{(2k-4)d}{2},$$

czyli

$$12 + 2m + \sum_{i=1} in_{i+6} - fd - ge \leq \sum_{i=k} \frac{in_i}{2} + \frac{(2k-4)d}{2}.$$

Dla $k = 13$

$$12 + 2m + \sum_{i=1} in_{i+6} - fd - ge \leq \sum_{i=k} \frac{in_i}{2} + 11d.$$

Po przeniesieniu, otrzymujemy

$$\begin{aligned} (11+f)d + ge &\geq 12 + 2m + \sum_{i=1} in_{i+6} - \sum_{i=13} \frac{i}{2} n_i \\ &= 12 + 2m + \sum_{i=1}^6 in_{i+6} + \sum_{i=7} \left(i - \frac{i+6}{2}\right) n_{i+6} \\ &= 12 + 2m + \sum_{i=1}^6 in_{i+6} + \sum_{i=7} \frac{i-6}{2} n_{i+6} \\ &= 12 + 2m + \sum_{i=1}^6 in_{i+6} + \sum_{i=1} \frac{i}{2} n_{i+12}. \end{aligned} \tag{2}$$

Ponieważ $(11+f) \geq g$,

$$(11+f)d + ge \leq (11+f)(d+e)$$

i $f \leq 15$

$$(11+f)(d+e) \leq 26(d+e) = 26R.$$

A więc otrzymujemy

$$26R \geq 12 + 2m + \sum_{i=1}^6 in_{i+6} + \sum_{i=1} \frac{i}{2} n_{i+12}. \tag{3}$$

Teraz, z $f \geq 1$, mamy

$$\frac{11d}{12} \leq \frac{11fd}{12}.$$

Po obustronnym dodaniu $\frac{fd}{12}$

$$\frac{(11+f)d}{12} \leq fd,$$

czyli

$$fd + ge \geq \frac{1}{12}(11+f)d + ge.$$

Z (1) i (2) otrzymujemy

$$\begin{aligned}
B &\leq 12 + 2m + \sum_{i=1}^6 in_{i+6} - \frac{1}{12} \left(12 + 2m + \sum_{i=1}^6 in_{i+6} + \sum_{i=1}^6 \frac{i}{2} n_{i+12} \right) \\
&= \frac{11}{12} \left(12 + 2m + \sum_{i=1}^6 in_{i+6} \right) + \sum_{i=7}^6 in_{i+6} - \sum_{i=1}^6 \frac{i}{24} n_{i+12} \\
&= \frac{11}{12} \left(12 + 2m + \sum_{i=1}^6 in_{i+6} \right) + \sum_{i=1}^6 (i+6) n_{i+12} - \sum_{i=1}^6 \frac{i}{24} n_{i+12} \\
&= \frac{11}{12} \left(12 + 2m + \sum_{i=1}^6 in_{i+6} \right) + \sum_{i=1}^6 \left(i+6 - \frac{i}{24} \right) n_{i+12} \\
&= \frac{11}{12} \left(12 + 2m + \sum_{i=1}^6 in_{i+6} \right) + \sum_{i=1}^6 \left(\frac{23i}{24} + 6 \right) n_{i+12}.
\end{aligned}$$

Ostatecznie

$$B \leq \frac{11}{12} \left(12 + 2m + \sum_{i=1}^6 in_{i+6} \right) + \sum_{i=1}^6 \left(\frac{23i}{24} + 6 \right) n_{i+12}. \quad (4)$$

Więc z (3) i (4)

$$\frac{B}{26R} \leq \frac{\frac{11}{12} \left(12 + 2m + \sum_{i=1}^6 in_{i+6} \right) + \sum_{i=1}^6 \left(\frac{23i}{24} + 6 \right) n_{i+12}}{12 + 2m + \sum_{i=1}^6 in_{i+6} + \sum_{i=1}^6 \frac{i}{2} n_{i+12}}.$$

Ponieważ $\frac{k-2}{k-1}$ dla k rosnących dąży do 1, to

$$\frac{B}{26R} \leq \frac{\sum_{i=1}^6 \left(\frac{23i}{24} + 6 \right) n_{i+12}}{\sum_{i=1}^6 \frac{i}{2} n_{i+12}}.$$

Porównując i -ty składnik sumy z licznika z odpowiadającym mu składnikiem sumy z mianownika zauważamy, że dla $i = 1$ ich stosunek jest największy. Dlatego

$$\frac{B}{26R} \leq \frac{\left(\frac{23}{24} + 6 \right) n_{13}}{\frac{1}{2} n_{13}} = 2 \cdot \frac{167}{24} = \frac{167}{12}.$$

Więc

$$B < 362R. \quad (5)$$

□

Uwaga 3.7 (Różnice w dowodzie). W pracy [3] przyjęto, że podczas redukcji drugiego typu z Q_5 zostanie usunięte co najwyżej 6 wierzchołków, a nie 15. Nie zostało wytłumaczone, dlaczego dozwolone było takie ograniczenie. Jednak w późniejszych obliczeniach nie ma to dużego wpływu. Ograniczenie przez 6 pojawia się między nierównościami (2) a (3), stąd otrzymana nierówność (3) ma postać

$$17R \geq 12 + 2m + \sum_{i=1}^6 in_{i+6} + \sum_{i=1}^6 \frac{i}{2} n_{i+12}.$$

Następnie $\frac{B}{17R}$

$$\frac{B}{17R} \leq \frac{\sum_{i=1}^6 \left(\frac{23i}{24} + 6 \right) n_{i+12}}{\sum_{i=1}^6 \frac{i}{2} n_{i+12}}.$$

Dlatego ostateczne ograniczenie w twierdzeniu jest mocniejsze i wynosi $\frac{B}{R} < 237$. Nadal jest to ograniczenie przez stałą liczbę naturalną.

3.1 Złożoność czasowa

Przyjrzyjmy się złożoności czasowej naszego algorytmu. Zauważmy (Algorytm 4), że usuwanie wierzchołka wykonywane jest w czasie stałym równym stopniowi tego wierzchołka, gdzie stopień jest nie większy niż 5. Funkcja usunięcia wierzchołka w ciągu kolorowania grafu rzędu n będzie wykonana co najwyżej $n - 5$ razy, więc ostatecznie usuwanie zajmie $O(5(n - 5))$, czyli $O(n)$.

Algorytm 4 Usuwanie wierzchołka

```
procedure CHECK( $G, v$ )
   $p \leftarrow$  kopia listy  $L[v]$ 
  if  $D[v] = 5$  then
    dodaj  $v$  na koniec kolejki  $Q_5$ 
  else if  $D[v] = 4$  then
    przenieś  $v$  z  $Q_5$  do  $Q_4$ 
  end if
end procedure
procedure DELETE( $v$ )
   $p \leftarrow$  kopia listy  $L[v]$ 
  for  $w \in L[v]$  do
    usuń  $v$  z  $L[w]$ 
     $D[w] \leftarrow D[w] - 1$ 
    CHECK( $w$ )
  end for
  dodaj  $(v, p)$  do stosu  $S$ 
  zmniejsz liczbę wierzchołków o 1
end procedure
```

Teraz spójrzmy na funkcję identyfikacji (Algorytm 5). Złożoność czasowa pojedynczej identyfikacji zależy od sumy stopni identyfikowanych wierzchołków. Dzięki ograniczeniu ich stopnia przez stałą k ich suma nie będzie większa niż $2k$. Stąd identyfikacja również wykonywana jest w czasie stałym. Ponownie identyfikacja może być wywołana co najwyżej $\frac{n-5}{2}$ razy, więc podczas wszystkich etapów daje złożoność czasową $O(k(n - 5)) = O(n)$.

Algorytm 5 Identyfikacja wierzchołków

```
procedure IDENTIFY( $G, u, v$ )  
   $p \leftarrow$  kopia listy  $L[v]$   
  for  $w \in L[u]$  do  
    usuń  $u$  z  $L[w]$   
    if  $w \notin L[v]$  then  
      dodaj  $w$  do  $L[v]$   
      dodaj  $v$  do  $L[w]$   
       $D[v] \leftarrow D[v] + 1$   
      if  $D[v] = 6$  then  
        usuń  $v$  z  $Q_5$   
      else if  $D[v] = 5$  then  
        przenieś  $v$  z  $Q_4$  do  $Q_5$   
      end if  
    else  
       $D[w] \leftarrow D[w] - 1$   
      CHECK( $w$ )  
    end if  
  end for  
  if  $u \in Q_5$  then  
    usuń  $u$  z  $Q_5$   
  else if  $u \in Q_4$  then  
    usuń  $u$  z  $Q_4$   
  end if  
  dodaj  $(u, v)$  do stosu  $S$   
  zmniejsz liczbę wierzchołków o 1  
end procedure
```

Algorytmy 4 i 5 realizują funkcje ζ i η zdefiniowane w Definicjach 2.1 i 2.2. Były używane w uproszczonej wersji algorytmu (Algorytm 1).

Algorytm 6 Redukcja

```
procedure REDUCE( $G$ )  
  while  $|G| > 5$  do  
    if  $Q_4$  nie jest puste then  
      DELETE(pierwszy element z  $Q_4$ )  
    else  
       $v \leftarrow$  pierwszy element z  $Q_5$   
      if  $\exists_{u \neq w \in N(v)} (u \notin N(w) \wedge d(u) < k \wedge d(w) < k)$  then  
        DELETE( $v$ )  
        IDENTIFY( $u, w$ )  
      else  
        zwróć  $v$  na koniec kolejki  $Q_5$   
      end if  
    end if  
  end while  
end procedure
```

W jednej iteracji pętli, w funkcji redukcji co najmniej jeden wierzchołek zostanie usunięty lub zablokowany. Z Twierdzenia 3.6 wiemy, że liczba zablokowanych wierzchołków napotkanych

we wszystkich etapach musi być mniejsza niż 362 razy całkowita liczba wykonanych redukcji. Redukcji będzie co najwyżej $n - 5$, więc zablokowanych wierzchołków we wszystkich etapach będzie mniej niż $362(n - 5)$. Stąd redukcja ma liniową złożoność czasową zależną od rzędu grafu.

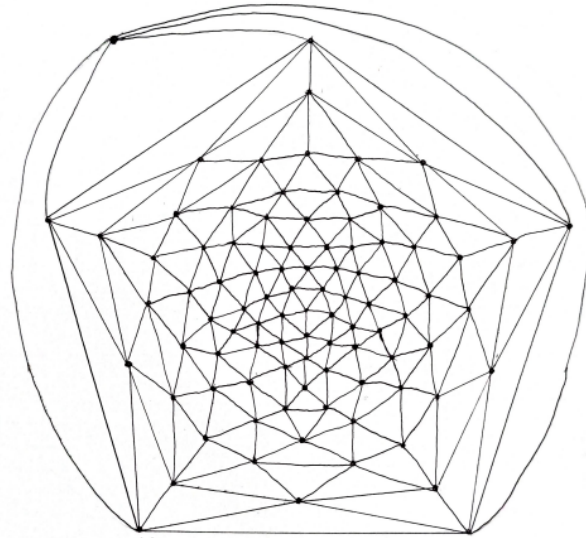
Funkcja kolorująca graf również działa w czasie liniowym, ponieważ stos S zawiera mniej niż n elementów.

3.2 Dlaczego $k = 13$?

Pokazaliśmy, że dla k równego 13 algorytm działa w czasie liniowym, ale dlaczego nie zdecydowano się na inną liczbę? Zastanówmy się, dla jakich k algorytm będzie działał.

3.2.1 Dolne ograniczenie k

Żeby algorytm działał, k musi być większe niż 6. Świadkiem na to jest dwudziestościan ścięty z dodatkowymi wierzchołkami. Do każdej ściany dwudziestościanu ściętego dodajemy jeden wierzchołek i łączymy z wierzchołkami tej ściany. Tym sposobem otrzymujemy poniższy graf (Rysunek 11). Widzimy, że graf ma wierzchołki stopni 5 lub 6 oraz wszystkie sąsiady wierzchołków o stopniu 5 mają stopień równy 6. Więc $k = 6$ spowodowałoby zablokowanie wszystkich wierzchołków stopnia 5 i algorytm szukałby bez końca niezablokowanych wierzchołków.



Rysunek 11

3.2.2 Rozważania dla dużych k

Przeprowadźmy ogólne obliczenia z dowodu Twierdzenia 3.6 dla liczby naturalnej k . Przyjmujemy te same oznaczenia na Q_5 , e , d . Nadal mamy

$$n_5 = 12 + 2m + \sum_{i=1} in_{i+6},$$

więc

$$B = 12 + 2m + \sum_{i=1} in_{i+6} - fd - ge. \quad (6)$$

Ponownie z Lematu 3.5 mamy

$$B \leq \sum_{i=k} \frac{in_i}{2} + \frac{(2k-4)d}{2},$$

czyli

$$12 + 2m + \sum_{i=1} in_{i+6} - fd - ge \leq \sum_{i=k} \frac{in_i}{2} + \frac{(2k-4)d}{2}.$$

Po przeniesieniu, otrzymujemy

$$\begin{aligned} (k-2+f)d + ge &\geq 12 + 2m + \sum_{i=1} in_{i+6} - \sum_{i=1} \frac{i+k-1}{2} n_{i+k-1} \\ &= 12 + 2m + \sum_{i=1}^{k-7} in_{i+6} + \sum_{i=1} \left(\frac{i+k-1}{2} - 6 \right) n_{i+k-1} \\ &= 12 + 2m + \sum_{i=1}^{k-7} in_{i+6} + \sum_{i=1} \left(\frac{i+k-13}{2} \right) n_{i+k-1}, \end{aligned}$$

ostatecznie dostajemy

$$(k-2+f)d + ge \geq 12 + 2m + \sum_{i=1}^{k-7} in_{i+6} + \sum_{i=1} \left(\frac{i+k-13}{2} \right) n_{i+k-1}, \quad (7)$$

dla $k \geq 8$. Ponieważ $(k-2+f) \geq g$,

$$(k-2+f)d + ge \leq (k-2+f)(d+e) \leq (k-2+15)(d+e) = (k+13)(d+e) = (k+13)R.$$

A więc otrzymujemy

$$(k+13)R \geq 12 + 2m + \sum_{i=1}^{k-7} in_{i+6} + \sum_{i=1} \left(\frac{i+k-13}{2} \right) n_{i+k-1}. \quad (8)$$

Teraz, z $f \geq 1$, mamy

$$\frac{(k-2)d}{k-1} \leq \frac{(k-2)fd}{k-1},$$

po obustronnym dodaniu $\frac{fd}{k-1}$

$$\frac{(k-2+f)d}{k-1} \leq fd,$$

czyli

$$fd + ge \geq \frac{1}{k-1}(k-2+f)d + ge.$$

Z (6) i (7) otrzymujemy

$$\begin{aligned} B &\leq 12 + 2m + \sum_{i=1} in_{i+6} - \frac{1}{k-1} \left(12 + 2m + \sum_{i=1}^{k-7} in_{i+6} + \sum_{i=1} \frac{i+k-13}{2} n_{i+k-1} \right) \\ &= \frac{k-2}{k-1} \left(12 + 2m + \sum_{i=1}^{k-7} in_{i+6} \right) + \sum_{i=k-6} in_{i+6} - \frac{1}{k-1} \sum_{i=1} \frac{i+k-13}{2} n_{i+k-1} \\ &= \frac{k-2}{k-1} \left(12 + 2m + \sum_{i=1}^{k-7} in_{i+6} \right) + \sum_{i=1} (i+k-7)n_{i+k-1} - \sum_{i=1} \frac{i+k-13}{2(k-1)} n_{i+k-1} \\ &= \frac{k-2}{k-1} \left(12 + 2m + \sum_{i=1}^{k-7} in_{i+6} \right) + \sum_{i=1} \left(\frac{2(k-1)(i+k-7) - (i+k-13)}{2(k-1)} \right) n_{i+k-1} \\ &= \frac{k-2}{k-1} \left(12 + 2m + \sum_{i=1}^{k-7} in_{i+6} \right) + \sum_{i=1} \left(\frac{2k^2 + (2i-17)k + 27 - 3i}{2(k-1)} \right) n_{i+k-1}. \end{aligned}$$

Ostatecznie

$$B \leq \frac{k-2}{k-1} \left(12 + 2m + \sum_{i=1}^{k-7} i n_{i+6} \right) + \sum_{i=1} \frac{2k^2 + (2i-17)k + 27 - 3i}{2(k-1)} n_{i+k-1}. \quad (9)$$

Więc z (8) i (9)

$$\frac{B}{(k+13)R} \leq \frac{\frac{k-2}{k-1} \left(12 + 2m + \sum_{i=1}^{k-7} i n_{i+6} \right) + \sum_{i=1} \frac{2k^2 + (2i-17)k + 27 - 3i}{2(k-1)} n_{i+k-1}}{12 + 2m + \sum_{i=1}^{k-7} i n_{i+6} + \sum_{i=1} \left(\frac{i+k-13}{2} \right) n_{i+k-1}}.$$

Ponieważ $\frac{k-2}{k-1}$ dla k rosnących dąży do 1, to

$$\frac{B}{(k+13)R} \leq \frac{\sum_{i=1} \frac{2k^2 + (2i-17)k + 27 - 3i}{2(k-1)} n_{i+k-1}}{\sum_{i=1} \left(\frac{i+k-13}{2} \right) n_{i+k-1}}. \quad (10)$$

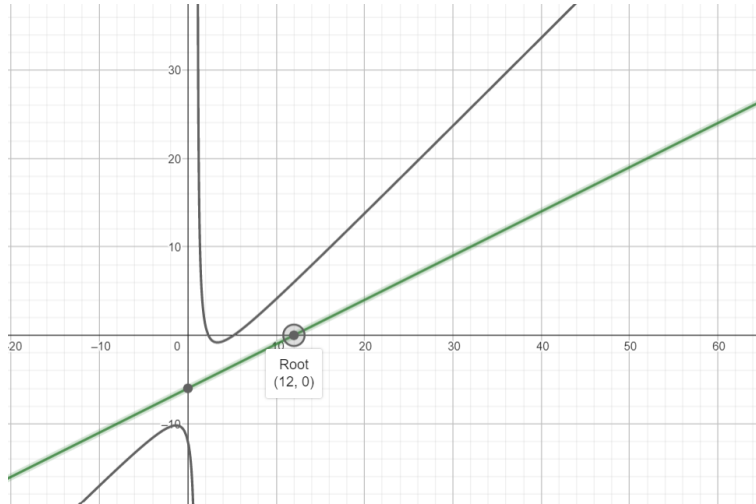
Interesuje nas najmniejszy iloczyn $\frac{B}{R}$, więc zauważmy, że porównując ze sobą odpowiednie elementy sumy z licznika i sumy z mianownika, największa proporcja występuje dla pierwszego elementu, dla $i = 1$. Pierwszy element sumy w liczniku prawej strony nierówności (10) jest postaci

$$a_1 = \frac{2k^2 - 15k + 24}{2(k-1)}.$$

Natomiast pierwszy element sumy w mianowniku wygląda następująco

$$b_1 = \frac{k-12}{2}.$$

Na Rysunku 12. zostały przedstawione te elementy, jako funkcje zależne od k . Szara krzywa to wykres funkcji $a_1(k) = \frac{2k^2-15k+24}{2(k-1)}$, a zielona prosta to wykres funkcji $b_1(k) = \frac{k-12}{2}$.

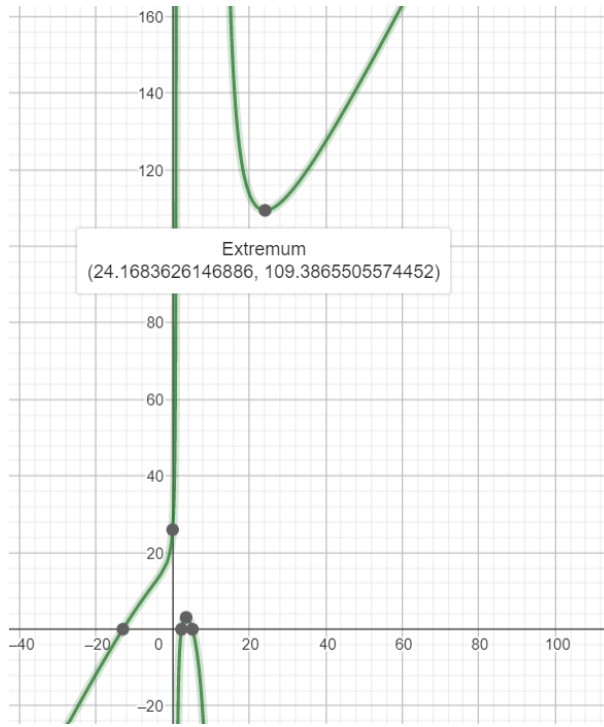


Rysunek 12

Iloraz tych funkcji pomnożony przez $(k+13)$ daje nam poniższą nierówność.

$$\frac{B}{R} \leq (k+13) \frac{2k^2 + (2-17)k + 27 - 3}{(k-1)(1+k-13)} = \frac{2k^3 + 11k^2 - 171k + 312}{k^2 - 13k + 12}$$

Na Rysunku 13. znajduje się wykres funkcji $f(k) = \frac{2k^3+11k^2-171k+312}{k^2-13k+12}$.



Rysunek 13

3.3 Wnioski

Duże k powoduje wydłużenie pracy identyfikacji dwóch wierzchołków. Jednak podczas identyfikacji tylko jednemu wierzchołkowi stopnie mogą znacznie wzrosnąć, a reszcie wierzchołków stopnie spadną. Z drugiej strony im k jest mniejsze, tym liczba zablokowanych wierzchołków jest większa. To powoduje, że algorytm musi dłużej szukać wierzchołka, który będzie mógł być usunięty. Stąd można by wnioskować, że im k jest większe, tym czas działania powinien być krótszy.

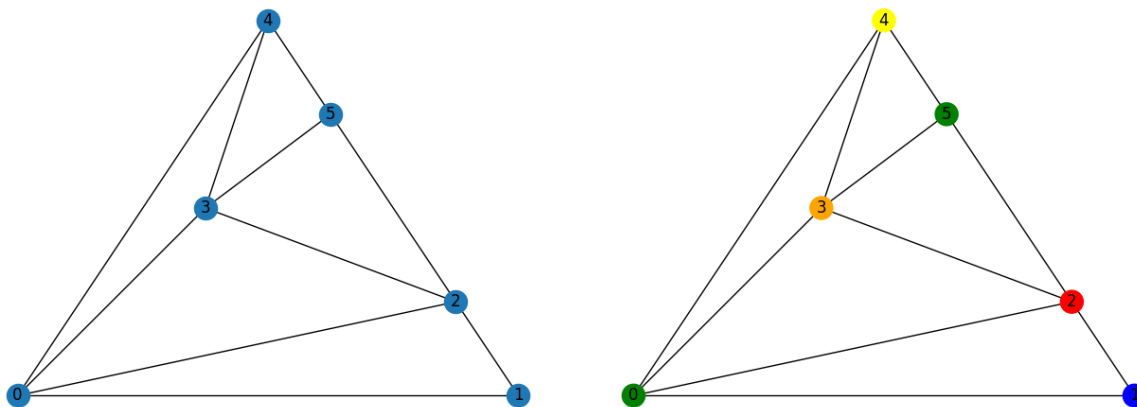
Niestety przeprowadzone obliczenia, wykonane na podstawie obliczeń z pracy [3] nie potwierdzają intuicyjnego myślenia. Powodem może być fakt, że podczas obliczeń wykonywane jest ograniczenie liczby zablokowanych wierzchołków przy rywalizujących założeniach. Mianowicie w drugiej części Lematu 3.5 rozważana jest sytuacja, gdy identyfikowane sąsiady wierzchołka v nie mają żadnych innych wspólnych sąsiadów oprócz v . Natomiast na początku dowodu Twierdzenia 3.6 rozważana sytuacja jest zupełnie odwrotna. To znaczy identyfikowane sąsiady v mają wszystkie sąsiady wspólne. Oczywiście nie może być takiej sytuacji, że wszystkie wierzchołki spełniają oba te założenia.

Ponieważ w pracy [3] wartość k narzucono odgórnie i nie poświęcono uwagi na jej wyjaśnienie, możliwe, że eksperymentalnie najlepsze wyniki zaobserwowano przy $k = 13$. To oznacza, że może istnieć lepszy wybór wartości k , przy którym algorytm działa szybciej. Jednak analiza empiryczna lub dalsze rozważania teoretyczne wykraczają poza zakres tej pracy.

3.4 Przykłady

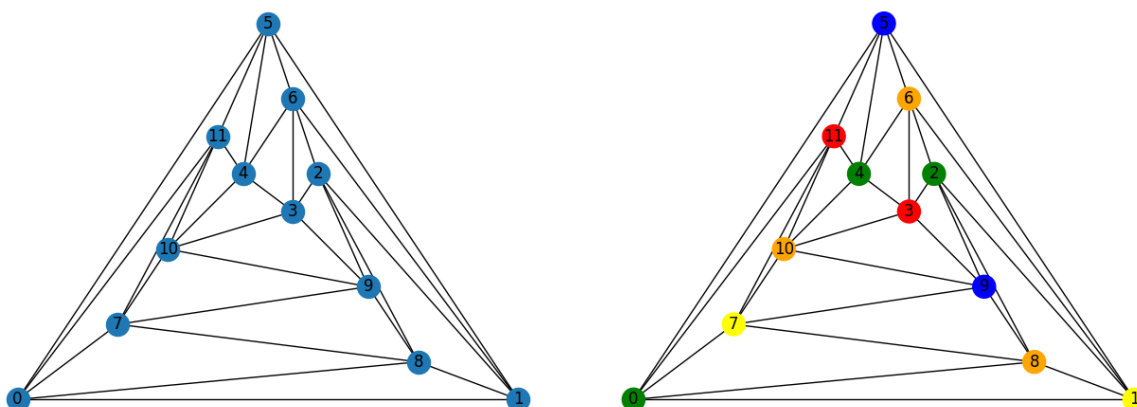
Program został napisany w języku Python, na podstawie pseudokodu zawartego w pracy [3].

Przykład 1. Rozważmy graf $G = (V, E)$, gdzie $V = \{0, 1, 2, 3, 4, 5\}$ i $E = \{(0, 1), (0, 2), (0, 3), (0, 4), (1, 2), (2, 3), (3, 4), (3, 5), (5, 4), (5, 2)\}$. Po uruchomieniu algorytmu otrzymujemy poniższe pokolorowanie.

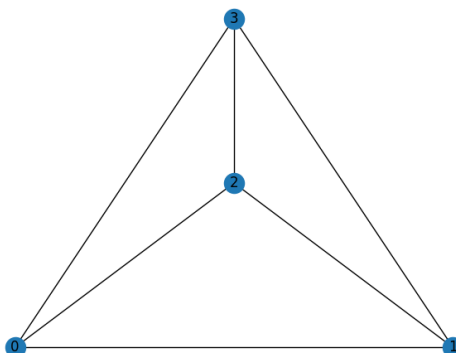


W Przykładzie 1 wszystkie wierzchołki miały stopień nie większy niż 4, czyli wykonywała się tylko redukcja pierwszego typu. Żeby przetestować działanie redukcji drugiego typu, rozważmy graf z wierzchołkami o wyższych stopniach.

Przykład 2. Dwudziestościan ma wierzchołki o stopniach równych 5. Czyli będzie musiała wywołać się redukcja drugiego typu, tym samym sprawdzimy działanie funkcji identyfikacji.

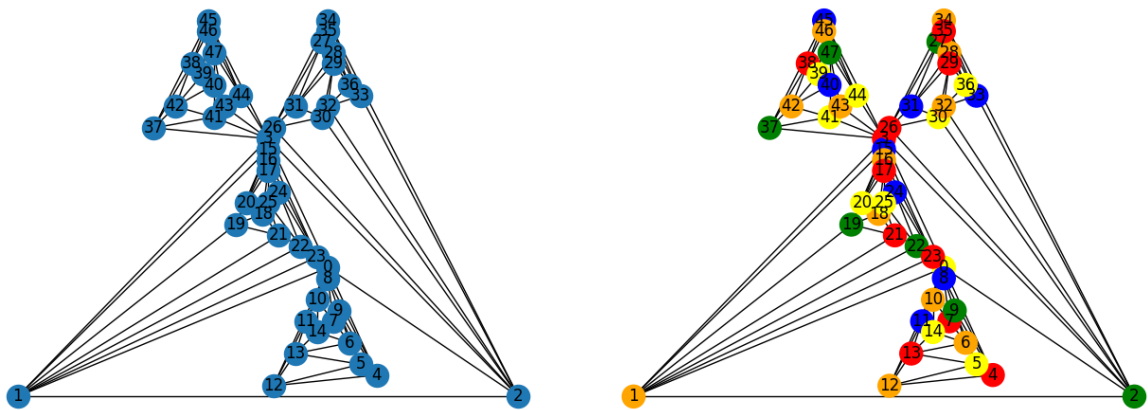


Przykład 3. Najpierw zaimplementujemy czworościan. Zmodyfikujemy go w celu zwiększenia stopni wszystkich wierzchołków.



Sklejamy czworościan z czterema dwudziestościanami względem wszystkich wierzchołków czworościanu. Dzięki temu będziemy mieli tylko wierzchołki stopnia 5 lub 8. W ten sposób

możemy modyfikować każdy graf planarny, żeby otrzymać graf z wierzchołkami o wyższych stopniach.



Literatura

- [1] N. Chiba, T. Nishizeki, N. Saito, (1981). *A linear 5-coloring algorithm of planar graphs*, Journal of Algorithms, 2(4): 317-327.
- [2] M. Valiunas, (2022). *Graph Theory*, dostęp z dnia 14.01.2023r.
<https://math.uni.wroc.pl/~valiunas/graph-theory/GraphTheory-notes.pdf>.
- [3] M. H. Williams, (1985). *A Linear Algorithm for Colouring Planar Graphs with Five Colours*, The Computer Journal 28.1: 78-81.
- [4] M. H. Williams, (1980). *Cubic map configurations*, Information Processing Letters 11.4-5: 180-185.
- [5] M. H. Williams, K. T. Milne, (1984). *The performance of algorithms for colouring planar graphs*, The Computer Journal 27.2: 165-170.