Algorytmy Zaawansowane
Etap 1 - Dokumentacja

Wojciech Kowalik, Konrad Miśkiewicz

1 Wstęp

1.1 Treść zadania

Są 2 kajaki, n osób oraz zbiór par, które chcą razem przepłynąć się kajakiem (jedna osoba może chcieć przepłynąć się kolejno z wieloma osobami). Zastosować algorytm znajdujący skojarzenie doskonałe do wyznaczenia rozkładu jazdy kajaków, tak aby zminimalizować ilość kursów pojedynczych kajaków. Należy przy tym spełnić życzenia wszystkich osób.

1.2 Skojarzenia w grafie

Niech $G = (V, E)$ będzie grafem nieskierowanym.

Definicja 1 *Skojarzeniem w grafie G nazywamy każdy podzbiór krawędzi E taki, w którym co najwyżej jedna krawędź z $M \subseteq E$ jest incydentna z każdym wierzchołkiem w V. O wierzchołku v incydentnym do pewniej krawędzi z M mówimy, że jest skojarzony, w przeciwnym przypadku v nazywamy wolnym. Podobnie jeżeli krawędź e należy do skojarzenia, mówimy, że jest ona skojarzona a w przeciwnym wypadku mówimy, że jest to krawędź wolna.*

Definicja 2 *Skojarzenie M nazywamy maksymalnym gdy ma ono największą liczność spośród skojarzeń w G.*

Definicja 3 *Skojarzenie M nazywamy doskonałym gdy jest ono skojarzeniem maksymalnym i wszystkie wierzchołki grafu G są skojarzone (brak wierzchołków wolnych). Dla grafów o nieparzystej liczbie wierzchołków nie istnieje skojarzenie doskonałe.*

1.3 Ścieżka powiększająca

Ścieżką powiększającą nazwiemy ścieżkę prostą P taką, że jej krawędzie są na przemian skojarzone i wolne, a końce są wolne. Łatwo zauważyć, że jeżeli istnieje ścieżka powiększająca P względem M, to M nie jest skojarzeniem maksymalnym. Używając wtedy ścieżki P, możemy skonstruować skojarzenie większe biorąc $M = M \oplus p$, czyli zamieniając na ścieżce krawędzie wolne na skojarzone i na odwrót.

2 Opis rozwiązania

2.1 Idea rozwiązania problemu

Aby prawidłowo wyznaczyć rozkład jazdy kajaków konieczna będzie budowa odpowiedniego grafu, a następnie zastosowanie na tym grafie algorytmu Edmondsa, znajdującego skojarzenie maksymalne w grafie.

Cały proces wyznaczania rozkładu jazdy kajaków rozpoczynamy od zbudowania następującego grafu G:

- liczba wierzchołków odpowiadająca liczbie osób - każdy wierzchołek odpowiada jednej osobie,
- krawędzie pomiędzy wierzchołkami odpowiadającymi osobom, które chcą przepłynąć się razem kajakiem.

Następnym krokiem jest przekształcenie grafu G do takiej postaci, w której każda krawędź w nowo powstałym grafie oznaczać będzie równoczesny kurs dwóch kajaków. Graf taki możemu zbudowac stosując dwa elementarne operacje na grafach:

$$G \Rightarrow G' \Rightarrow G''.$$

Pierwszym przekształceniem jest zamiana grafu G w graf krawędziowy $G' = L(G)$, tzn. taki którego zbiorem wierzchołków jest zbiór krawędzi grafu $G : V(G') = E(G)$, natomiast zbiorem krawędzi $E(G')$ jest zbiór par elementów zbioru $E(G)$, przy czym wierzchołki $v1, v2$, $v3, v4$ grafu G' są połączone wtedy i tylko wtedy, gdy $v1 = v3$ lub $v1 = v4$ lub $v2 = v3$ lub $v2 = v4$. W drugim kroku należy wyznaczyć dopełnienie grafu G' i w ten sposób otrzymamy graf G''. W grafie tym, każda krawędź odpowiada równoczesnemu kursowi dwóch kajaków - każdy z dwoma osobami. Nietrudno jednak zauważyć, że do jednego wierzchołka w grafie G'', który de facto odpowiada jednej parze osób, które razem chcą popłynąć dochodzić może wiele krawędzi. Aby zapewnić, że każda para popłynie tylko raz należy więc wyznaczyć skojarzenie maksymalne (doskonałe) w tym grafie, a następnie po kolei usuwać krawędzie skojarzone wraz z wierzchołkami i dopisywać pary odpowiadające usuwanym wierzchołkom do rozkładu jazdy (usunięcie jednej krawędzi spowoduje dopisanie do rozkładu dwóch par osób, które jednocześnie będą płynąć). Gdy usunięte zostaną wszystkie krawędzie skojarzone wraz z ich wierzchołkami, a w grafie G'' nadal zostaną jakieś wierzchołki, należy po kolei, każdy z tych wierzchołków usunąć z grafu i dopisać odpowiadającą mu parę osób do rozkładu jazdy. Pary odpowiadające tym wierzchołkom, będą pływać tylko pojedynczymi kursami kajaków.

Głównym problemem w konstrukcji algorytmu, znajdującego maksymalne skojarzenie w dowolnym grafie, jest problem znalezienia ścieżki powiększającej. Niestety, jeśli rozważany graf może być dowolnej postaci, to mogą istnieć naprzemienne ścieżki zaczynające się w wierzchołku wolnym i kończące się krawędzią skojarzoną i nie skojarzoną w wybranym wierzchołku. Nie możemy więc po prostu pamiętać, czy wierzchołek został już odwiedzony i użyć dzięki temu prostego przeszukiwania grafu. Aby rozwiązać ten problem posłużymy się przytoczonym poniżej lematem.

2.2 Lemat o ściąganiu cykli

Dla grafu G niech:

- M będzie skojarzeniem w G,
- Z będzie cyklem długości $2k + 1$ zawierającym k krawędzi z M i rozłącznym z resztą M.

Skonstruujmy nowy graf G' z G poprzez ściągnięcie Z do jednego wierzchołka. Skojarzenie $M' = M - E(Z)$ jest maksymalne w G' wtedy i tylko wtedy, gdy skojarzenie M jest maksymalne w G.

Dowód lematu zostanie przedstawiony w dalszej części dokumentu.

2.3 Algorytm Edmondsa

Niech M będzie pewnym skojarzeniem oraz niech S oznacza zbiór wierzchołków wolnych dla M. Lasem $M-alternujcym$ nazwiemy las L taki, że:

- korzeniem każdego drzewa w L jest wierzchołek S i drzewo to nie zawiera innych wierzchołków z S,
- każdy wierzchołek S należy do jednej składowej L,
- każda krawędź w odległości nieparzystej od korzenia należy do M.

Ponadto wprowadźmy następujące pojęcia:

- wierzchołki zewnętrzne - każdy punkt w lesie L w odległości parzystej od S,
- wierzchołki wewnętrzne - pozostałe punkty w L.

Dla przykładu, las składający się tylko z punktów S jest $M-alternujcy$.

Poniżej przedstawiono kompletny pseudokod algorytmu Edmondsa.

Algorithm 1 Pseudokod algorytmu Edmondsa

```
G - graf utworzony na podstawie osób, które chcą przepłynąć się kajakiem
M = ∅
las L to zbiór wierzchołków wolnych
while true do
  if istnieje zewnętrzny wierzchołek x ∈ L sąsiedni z y ∉ L then
    znajdź wierzchołek z, taki że yz ∈ M
    L = L + xy + yz
  else
    if x1, x2 ∈ L to wierzchołki zewnętrzne połączone krawędzią then
      if x1, x2 należą do różnych składowych L then
        niech Pi to ścieżka z xi do korzenia jego składowej
        rozwiń wszystkie ściągnięte cykle w grafie G
        M = M ⊕ (P1 + P2 + x1x2)
        las L to zbiór wierzchołków wolnych
      else
        niech C będzie cyklem utworzonym przez krawędź x1x2 oraz ścieżkę x − y w L
        niech P będzie ścieżką w L łączącą C z korzeniem drzewa
        M = M ⊕ P
        utwórz graf G' poprzez ściągnięcie C
        G = G'
      end if
    end if
  else
    rozwiń wszystkie ściągnięte cykle w grafie G
    return M
  end if
end while
```

Przy bezpośredniej implementacji złożonosc algorytmu Edmondsa wynosi $O(n^2 \cdot m)$, gdzie n - ilość wierzchołków grafu G, a m - ilość jego krawędzi. Jednak przy odpowiedniej implementacji możliwe jest osiągnięcie pesymistycznej złożoności $O(n^3)$.

3 Analiza poprawności i złożoności czasowej

3.1 Dowód poprawności i analiza złożoności algorytmu wyznaczającego rozkład jazdy kajaków

Dowód poprawności algorytmu rozpoczniemy od udowodnienia, że algorytm zaprezentowany w punkcie 2.1 zakończy swoje działanie. Jest to oczywiste, gdyż konstrukcja grafu G odbywa się w skończonym czasie. Utworzony zostaje graf, który zawiera tyle wierzchołków ile jest osób. Do każdej osoby zostaje przypisany wierzchołek grafu, a następnie dodawane są do niego krawędzie pomiędzy wierzchołkami odpowiadającymi osobom, które razem chcą przepłynąć się kajakiem.

Następnym krokiem jest przekształcenie grafu G w graf krawędziowy G'. W grafie G' wierzchołki są ze sobą połączone krawędzią, wtedy i tylko wtedy, gdy w grafie G odpowiadające im krawędzie miały wspólny wierzchołek. Operacja ta w sposób oczywisty również jest skończona.

Ostatnim już krokiem prowadzącym do utworzenia poprawnego grafu jest wyznaczenie dopełnienia grafu G'. Oczywiście również ta operacja jest skończona, gdyż polega na usunięciu obecnych krawędzi z grafu G' i dodaniu ich pomiędzy tymi wierzchołkami, pomiędzy którymi ich wcześniej nie było. W wyniku tej operacji powstanie graf G''.

Pokazaliśmy zatem, że cały proces utworzenia grafu G'' jest skończony. Kolejnym krokiem prowadzącym do ułożenia rozkładu jazdy kajaków jest wyznaczenie skojarzenia doskonałego w grafie G'' za pomocą algorytmu Edmondsa. Operacja ta jest również skończona, co zostanie udowodnione w punkcie 3.3. Na koniec pozostaje już tylko usuwanie krawędzi skojarzonych wraz z wierzchołkami, a następnie (jeśli zostaną) to pojedynczych wierzchołków. Z racji tego, że jest ich skończona ilość, to również cała operacja będzie skończona, a więc cały algorytm wyznaczania rozkładu jazdy kajaków zakończy swoje działanie.

Przejdźmy teraz do dowodu poprawności naszego algorytmu. W pierwszym jego kroku tworzymy graf G w którym każdy wierzchołek odpowiada jednej osobie, a krawędzie znajdują się pomiędzy wierzchołkami odpowiadającymi osobom, które razem chcą się przepłynąć. Następnym krokiem jest utworzenie grafu z grafu G odpowiadającego mu grafu krawędziowego G', w którym każdy wierzchołek odpowiada jednej parze osób, które razem chciały przepłynąć się kajakiem. A zatem jedna krawędź w grafie G' odpowiada dwóm parom, które chcą przepłynąć się kajakiem. Kłopot w tym, że krawędź ta, w rzeczywistości dotyczy tylko trzech osób, a nie czterech - jedna osoba jest wspólna dla każdej z par. Wynika to z faktu, iż po zamianie krawędzi z grafu G na wierzchołki w grafie G' połączone krawędziami zostały te wierzchołki, które w grafie G odpowiadały sąsiadującym ze sobą krawędziom. Aby rozwiązać ten problem i otrzymać graf, w którym każda krawędź odpowiada dwóm niezależnym parom osób wystarczy wyznaczyć dopełnienie grafu G'. W ten sposób otrzymujemy graf G'', w którym każda krawędź odpowiada dwóm niezależnym parom osób, które razem chcą przepłynąć się kajakiem. Najprostszym rozwiązaniem okazuje się teraz wyjmowanie po kolei krawędzi (wraz z wierzchołkami) z grafu G'' i dopisywanie odpowiadających im par osób do rozkładu jazdy. Kłopot w tym, że do danego wierzchołka może dochodzić wiele krawędzi. Rozwiązaniem tego problemu okazuje się zastosowanie algorytmu Edmondsa i wyznaczenie maksymalnego skojarzenia w grafie G''. Wyznaczenie takiego skojarzenia zagwarantuje nam, że do każdego wierzchołka będzie dochodzić co najwyżej jedna krawędź skojarzona. Dzięki temu możemy po kolei usuwać krawędzie skojarzone (wraz z wierzchołkami) i dopisywać odpowiadające im pary osób do rozkładu jazdy. Po usunięciu wszystkich krawędzi skojarzonych wraz z wierzchołkami, w grafie G'' mogą pozostać pojedyncze wierzchołki. Należy je po kolei usunąć z tego grafu i dopisać odpowiadające im pary osób do rozkładu jazdy. W ten sposób zagwarantowane zostało wpisanie do rozkładu wszystkich par osób, a przy tym wpisanie ich w sposób optymalny, gdyż na początku zrealizowane zostają wszystkie możliwe kursy dwóch kajaków jednocześnie, a na końcu ewentualne kursy pojedynczych kajaków.

Powyższe rozważania dowodzą poprawności algorytmu wyznaczania rozkładu jazdy kajaków. Dowód poprawności algorytmu Edmondsa zostanie przedstawiony poniżej.

3.2 Dowód lematu o ściąganiu cykli

Oznaczenia jak w lemacie.

Udowodnijmy najpierw, że jeżeli M' jest skojarzeniem maksymalnym, to także M jest maksymalne. Załóżmy przeciwnie, że M nie jest maksymalnym skojarzeniem w G. Wtedy istnieje ścieżka powiększająca P względem M. Jeżeli P jest rozłączna z Z, to P także jest ścieżką powiększającą względem M' i M' nie jest maksymalne. Zatem P przecina się z Z. Ponieważ Z ma jeden wierzchołek wolny, to co najmniej jeden z końców P nie leży na Z. Oznaczmy ten wierzchołek przez x. Poczynając od x, niech z będzie pierwszym punktem na ścieżce Z leżącym na P. Wtedy $P[x, z]$ jest ścieżką powiększającą względem M' w G', ponieważ Z po ściągnięciu do z jest wolny.

Pokażemy teraz, że w lemacie zachodzi także wynikanie w drugim kierunku. Niech M' nie będzie maksymalnym skojarzeniem w G' oraz niech N' będzie liczniejszym skojarzeniem w G'. Rozwińmy cykl Z, aby odtworzyć G. Wtedy N' będzie skojarzeniem w G, kojarzącym co najwyżej jeden wierzchołek z Z. Możemy wtedy N' powiększyć o k krawędzi z Z, otrzymując skojarzenie N o rozmiarze:

$$|N| = |N'| + k > |M'| + k = |M|.$$

Tak więc M nie jest maksymalnym skojarzeniem w G, co kończy dowód lematu.

3.3 Dowód poprawności algorytmu Edmondsa

Na potrzeby dowodu poprawności algorytmu Edmondsa przytoczmy następujący lemat.

Lemat 1 *Dla dowolnego skojarzenia M w grafie $G = (V, E)$ i dowolnego podzbioru wierzchołków $X \subseteq V$ zachodzi:*

$$|M| \leq \frac{1}{2} \left(|V| - (c_n(G - X) - |X|)\right).$$

Dowód powyższego lematu: Zauważmy, że jeżeli nieparzystej składowej z grafu $G - X$ nie skojarzymy z wierzchołkiem z X, to pozostanie w niej co najmniej jeden wierzchołek wolny. Ponieważ z $c_n(G - X)$ nieparzystych składowych możemy skojarzyć w ten sposób co najwyżej $|X|$, to dla dowolnego skojarzenia M w grafie pozostanie co najmniej $c_n(G - X) - |X|$ wierzchołków wolnych. Tak więc skojarzonych wierzchołków będzie co najwyżej $|V| - (c_n(G - X) - |X|)$, a co za tym idzie, rozmiar M jest ograniczony tak jak w nierówności z lematu.

Powyższy lemat mówi, że aby udowodnić, że skojarzenie M jest maksymalne, musimy pokazać zbiór $X \in V$, dla którego w powyższym wzorze będzie zachodziła równość.

Algorytm Edmondsa kończy działanie, gdy każdy wierzchołek zewnętrzny w L ma jako sąsiadów wierzchołki wewnętrzne. Oznaczmy przez W zbiór wierzchołków wewnętrznych w L, a przez Z zbiór wierzchołków zewnętrznych. Mamy wtedy $|Z| - |W| = |V| - 2|M|$, ponieważ w każdym drzewie jest jeden wierzchołek zewnętrzny więcej i jest to wierzchołek wolny. Jeżeli usuniemy wierzchołki W z G, otrzymamy graf składający się z izolowanych wierzchołków zewnętrznych. Dlatego $c_n(G-W) = |Z|$ i $\frac{1}{2}\left(|V| - (c_n(G - W) - |W|)\right) = \frac{1}{2}\left(|V| - |Z| + |W|\right) = \frac{1}{2}\left(2|M|\right) = |M|$, czyli nierówność z powyższego lematu zachodzi z równością. Skojarzenie M jest więc maksymalne. Zauważmy, że z lematu o ściąganiu cykli wynika, że będzie ono też maksymalne po rozwinięciu wszystkich ściągniętych cykli.

4 Opis wejścia/wyjścia

4.1 Format pliku wejściowego

Format pliku wejściowego jest bardzo prosty. W pierwszej linii znajduje się liczba całkowita, która informuje o ilości osób. Kolejne linie zawierają zawsze dwie liczby oddzielone przecinkiem, informujące o tym, które osoby chcą razem przepłynąć się kajakiem. Osoby numerowane są od 0. Przykładowy plik wejściowy przedstawiono poniżej.

```
4
0,1
0,2
0,3
1,2
```

Powyższy przykład informuje o tym, że rozważane będą 4 osoby. Osoba o numerze 0 chce przepłynąć się z osobami o numerach 1, 2 i 3. Ponadto Osoba o numerze 1 chce przepłynąć się z osobą o numerze 2.

4.2 Format pliku wyjściowego

Format pliku wyjściowego jest podobny do formatu pliku wejściowego. Kolejne linijki przedstawiają kolejne kursy kajaków. W jednej linii mogą znajdować się maksymalnie 2 pary osób oddzielone spacją (a osoby w parach oddzielone są przecinkami). Przykładowy plik wyjściowy przedstawiono poniżej.

```
0,3 1,2
0,1
0,2
```

W powyższym przykładzie w pierwszym kursie popłyną 2 kajaki, natomiast w kolejnych dwóch już tylko po jednym.