

Grafy i Sieci – Projekt

Autorzy: Pracki Kamil, Pszczółkowski Mateusz

Treść zadania:

Opracować i zaimplementować algorytm optymalnego przypisania n pracowników do n maszyn.

Opis algorytmu:

Problem opisany w treści zadania sprowadza się do znalezienia maksymalnego skojarzenia w grafie dwudzielnym.

W przypadku naszego zadania, aby:

- Żadna maszyna ani pracownik nie byli przypisani więcej niż raz
- Przypisanie maszyny z pracownikiem było możliwe tylko wtedy, gdy pracownik obsługuje maszynę

Do rozwiązania problemu został wykorzystany algorytm Forda-Fulkersona z BFS. Celem algorytmu jest wyznaczenie maksymalnej ilości przepływu ze źródła do ujścia, nie przekraczając pojemności krawędzi i zachowując zasadę zachowania przepływu.

Działanie algorytmu:

1. Graf dwudzielny jest przekształcany w sieć przepływową poprzez:
 - Dodanie sztucznego źródła, które łączy się ze wszystkimi pracownikami (każda krawędź o pojemności 1).
 - Dodanie sztucznego ujścia, do którego prowadzą krawędzie ze wszystkich maszyn (krawędzie o pojemności 1).
 - Dodanie krawędzi od pracowników do maszyn zgodnie z macierzą zgodności
2. Algorytm szuka ścieżki powiększającej w grafie residualnym za pomocą BFS.
3. Aktualizuje przepływy w znalezionej ścieżce.
4. Powtarza do momentu aż nie da się znaleźć kolejnej ścieżki.
5. Zwraca przypisania z grafu residualnego (jeśli maszyna oddaje przepływ do pracownika to znaczy, że zostali skojarzeni).

Złożoność obliczeniowa: $\Theta(V \cdot E^2)$

E - liczba krawędzi

V - liczba wierzchołków

Złożoność pamięciowa: $\Theta(N \cdot M)$

N - liczba pracowników

M - liczba maszyn

Przykład działania:

1. Wywołanie programu - należy za pomocą konsoli przejść do folderu, w którym znajduje się `Mateusz_Pszczolkowski.exe` i wpisać:

```
Mateusz_Pszczolkowski --pracownikow 3 --maszyn 3
```

Dla przykładu tworzymy graf dwudzielny składający się z 3 pracowników oraz 3 maszyn.

2. Podanie kompatybilności:

```
Podaj kompatybilność (1 - kompatybilny, 0 - niekompatybilny):  
Pracownik 0 - Maszyna 0: 1  
Pracownik 0 - Maszyna 1: 0
```

Podajemy kompatybilność pracowników z maszynami. (1, jeśli i -ty pracownik może obsługiwać k -tą maszynę, 0 w przeciwnym wypadku).

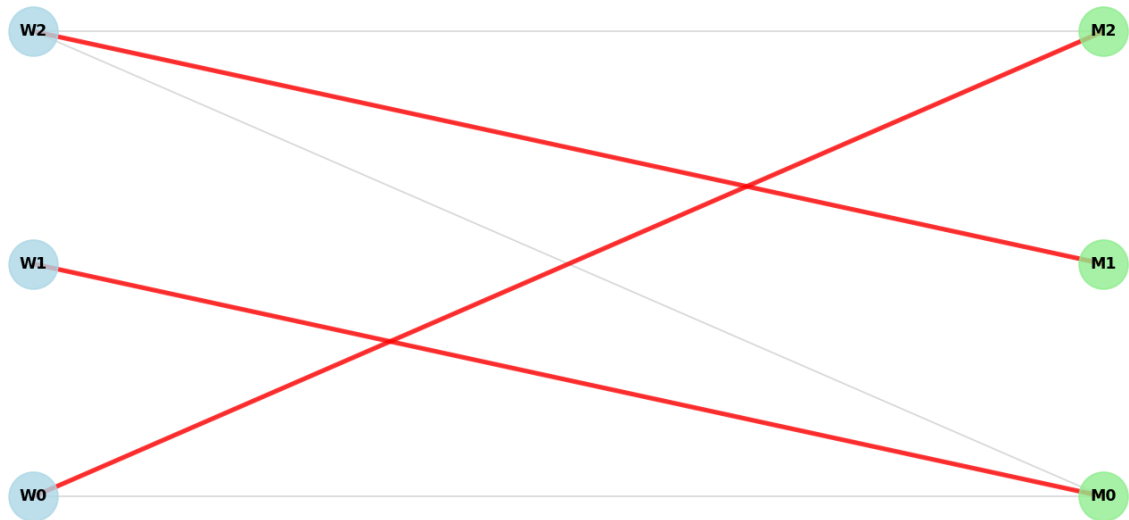
Na powyższym zrzucie ekranu Pracownik 0 jest w stanie obsługiwać Maszynę 0, ale Maszynę 1 już nie.

Dla przykładu:

```
Pracownik 0 - Maszyna 0: 1  
Pracownik 0 - Maszyna 1: 0  
Pracownik 0 - Maszyna 2: 1  
Pracownik 1 - Maszyna 0: 1  
Pracownik 1 - Maszyna 1: 0  
Pracownik 1 - Maszyna 2: 0  
Pracownik 2 - Maszyna 0: 1  
Pracownik 2 - Maszyna 1: 1  
Pracownik 2 - Maszyna 2: 1
```

3. Ukazuje się wizualizacja optymalnego przypisania pracowników do maszyn.

Graf dwudzielny: Przypisanie pracowników do maszyn
Czerwone krawędzie = optymalne przypisania



Czerwone linie - optymalne przypisanie i-tego pracownika do k-tej maszyny.

Szare linie – pozostałe maszyny obsługiwane przez i-tego pracownika.

Program zwraca także optymalne przypisanie w wersji tekstowej,

```
Optymalne przypisania:  
Pracownik 0 -> Maszyna 2  
Pracownik 1 -> Maszyna 0  
Pracownik 2 -> Maszyna 1
```

oraz macierz zgodności grafu (✓ - jeśli pracownik może obsługiwać daną maszynę, X jeśli nie).

Macierz zgodności:

| | M 0 | M 1 | M 2 |
|-----|-----|-----|-----|
| W 0 | ✓ | X | ✓ |
| W 1 | ✓ | X | X |
| W 2 | ✓ | ✓ | ✓ |