

Dokumentacja funkcjonalna symulatora podziału grafu

Maciej Palkowski, Wojciech Kornatowski

12.03.2025

Cel projektu

Symulator podziału grafu ma na celu wyznaczenie podgrafów zgodnie z wytycznymi użytkownika. Po otrzymaniu danych wejściowych program dzieli graf w taki sposób, aby jak najmniej połączeń zostało przeciętych oraz liczba węzłów podgrafów nie różniła się o więcej niż o podaną wartość wyrażoną w procentach. Program działa w pełni z linii poleceń oraz oferuje wiele flag, które dostosowują program do potrzeb użytkownika.

Dane wejściowe

Plik wejściowy do programu ma następującą postać:

- pierwszy wiersz opisuje ile maksymalnie węzłów może znaleźć się w jednym wierszu grafu, jednak nie jest to pewne, że dany graf musi mieć wiersz z taką ilością węzłów.
- drugi wiersz opisuje indeksy w których występują węzły w wierszu
- trzeci wiersz opisuje, który węzeł jest pierwszy w danym wierszu
- czwarty wiersz opisuje połączenia między węzłami
- piąty wiersz opisuje indeksy pierwszych węzłów z wiersza czwartego.

Kompilacja programu

Aby skompilować program należy wykonać:

`make`

Alternatywnie:

`make all`

Lub:

`make gsplit`

Plik wykonywalny utworzony jako wynik kompilacji projektu to "gsplit"

Kompilacja pojedynczego pliku bez linkowania:

```
make [ścieżka do pliku źródłowego]
```

Ponowna kompilacja:

```
make fclean
```

Usuwanie plików obiektowych:

```
make clean
```

Aby uruchomić testy programu należy wykonać:

```
make test
```

Argumenty wywołania programu

Wspierane opcje:

- **-i, --input** [plik] - Plik wejściowy (domyślnie: stdin)
- **-o, --output** [plik] - Plik wyjściowy (domyślnie: stdout)
- **-p, --parts** [liczba] - Liczba oczekiwanych grafów (domyślnie: 2)
- **-m, --margin** [procent] - Dopuszczalna różnica liczby wierzchołków (domyślnie: 10%)
- **-b, --binary** - Zapis jako plik binarny (domyślnie tekstowy)
- **-v, --verbose** - Więcej szczegółów
- **-h, --help** - Wyświetlenie pomocy

Przykładowe wywołanie programu:

```
./gsplit -i graf.txt -o output.txt -p 5 -m 15
```

Powyższe argumenty oznaczają, że program podzieli graf z podanego pliku na 5 części oraz liczba węzłów w tych częściach może różnić się o maksymalnie 15%.

Alternatywnie:

```
./gsplit -p 5 -m 15 < graf.txt > output.txt
```

Wykorzystywany algorytm

Wczytanie danych wejściowych

Odczytanie pliku zawierającego strukturę grafu i utworzenie odpowiadającej mu listy sąsiedztwa, opisującej węzły oraz ich połączenia.

Definicja parametrów wejściowych

- **num_nodes** – liczba węzłów w grafie.

- **target** – docelowa liczba podgrafów = $\text{num_nodes} / \text{liczba podziałów}$.
- **margin** – margines określający dopuszczalną różnicę w wielkości podgrafów.
- **target_min** – minimalna liczba węzłów w jednym podgrafie, określona jako:

$$\text{target_min} = \lfloor \text{target} - \text{margin} \rfloor$$

- **target_max** – maksymalna liczba węzłów w jednym podgrafie, określona jako:

$$\text{target_max} = \lfloor \text{target} + \text{margin} \rfloor$$

Tworzenie struktury podziału

Inicjalizacja odpowiedniej liczby list sąsiedztwa, odpowiadających podgrafom.

Podział grafu za pomocą algorytmu DFS

- Przeszukiwanie grafu przy użyciu algorytmu **Depth-First Search (DFS)**, rozpoczynając od najmniejszego dostępnego węzła.
- Wyszukiwanie podgrafów spełniających kryteria:

$$\text{target_min} \leq \text{liczba węzłów otrzymanego podgrafu} \leq \text{target_max}$$

- Obliczanie liczby krawędzi przeciętych podczas podziału, porównanie ich z innymi podgrafami i wybranie opcji, która przetnie jak najmniej krawędzi.

Iteracyjna kontynuacja podziału

- Po znalezieniu poprawnego podgrafu powrót do poprzedniego kroku.
- Kolejna iteracja rozpoczyna się od najmniejszego dostępnego węzła, który nie został jeszcze przypisany do żadnego podgrafu.

Zapis wyników

- Zapisanie podziału grafu w formacie pliku wejściowego.

Komunikaty błędów

Program zwróci komunikat o błędzie w następujących przypadkach:

- błędy alokacji pamięci
- błędy otwierania pliku wejściowego - brak uprawnień, nieistniejący plik, plik jest katalogiem, nieprawidłowe rozszerzenie

- błędy czytania z pliku wejściowego - plik ma nieodpowiedni format, nie udało się przeczytać pliku
- przekazano błędne opcje - nieznana opcja, przekazanie opcji wymagającej argument bez wskazania wartości, wartość dla opcji jest w złym formacie
- podana liczba podziału jest większa niż liczba węzłów grafu np. podana liczba: 10, liczba węzłów: 5
- program nie będzie w stanie zachować marginesu błędu np. liczba podziału: 4, margines błędu: 10, liczba węzłów: 14