AiSD - laboratorium

Projekt zespołowy - specyfikacja funckjonalna

Kacper Baczyński, Michał Kiełczykowski, Marek Knosala, Edward Sucharda

14 grudnia 2020

1 Wtęp

W tym dokumencie opisany został sposób korzytsania z programu, który jest celem projektu zespołowego na labolatorium przedmiotu Algorytmy i Struktury Danych prowadzonego przez Pawła Zawadzkiego w roku akademickim 2020/2021 na Wydziale Elektrycznym Politechniki Warszawskiej. Program służy do symulacji działania zespołu karetek przewożących pacjentów do szpitali. Każdy szpital ma swoją liczbę łóżek i informację ile z nich jest jeszcze wolnych. Więc nie zawsze szpital może przyjąć nowego pacjenta.

Szpitale mają swoje współrzędne kartezjańskie i leżą na terenie pewnego państwa. Granice tego państwa wyznacza największy wypukły wielokąt o wierchałkach opartch na szpitalach lub obiektu. Jedyna funkcjonalność obiektu to ewentualne poszerzenie granic państwa. Wybrane pary szpitali są połączone drogami, które leżą na prostej łączące współrzędne początkowego i końcowego szpitala. Gdy dwie przecinają się to jest tam skrzyżowanie. Tak więc droga z jednego szpitala do drugiego może byż bezpośrednia (prosta droga z jednego punktu na mapie do drugiego) lub pośrednia, czyli taka która zawiera zmianę drogi przy przejeździe przez skrzyżowanie lub przy przejeździe przez jakiś szpital.

Gdy tylko pojawia się jakiś pacjent na terenie państwa natychmiast znajduje sie przy nim karetka i wyrusza do najbliższego szpitala w linii prostej obierając kierunek prosto na szpital, gdyż chwilowo nie musi się poruszać po drogach. Jeżeli szpital, do którego dotarła nie ma wolnych łóżek to karetka musi podróżować do kolejnego szpitala po drogach. Zawsze gdy szpital jest pełny karetka wybiera ten szpital do którego droga bezpośrednia lub pośrednia jest najkrótsza aż znajdzie szpital, który może przyjąć pacjenta. Karetka nie sprawdza dwa razy tego samego szpitala. Gdy karetka odwiedza ostatni szpital i również ten nie ma wolnych łóżek to wtedy ustawia się w kolejce do tego szpitala czekając aż się zwolni łóżko.

Poniżej znajduje się krótka specyfikacja funckjonalna, w ktorej podane zostały paramtery wejściowe, sposób uruchomienia, wygląd i działanie interfejsu graficznego oraz opis możliwych nieprawdłowych użyć i odpowiadające im komunikaty.

2 Obowiązkowy plik wejściowy

Obowiązkowym plikiem wejściowym jest plik w formacie .txt, który składa się z trzech sekcji: szpitale, obiekty oraz drogi. Sekcje są w kolejności jak podano. Każda sekcja rozpoczyna się nagłówkiem, czyli jedną linijką tekstu składającą się ze znaku "#" oraz nazwy sekcji. W dalszej części sekcji są wiersze z danymi dotyczące danej sekcji.

Każda z sekcji ma swój unikalny porządek danych. W przypadku szpitali każdy wiersz składa się z:

- id szpitala, które musi być unikalną, nieujemną liczbą całkowitą
- nazwy szpitala, która jest dowolnym ciągiem znaków za wyjątkiem znaku "|"
- współrzędenj w osi x, która oznacza położenie w tej osi danego szpitala, która musi być całkowita
- współrzędenj w osi y, która oznacza położenie w tej osi danego szpitala, która musi być całkowita
- liczby łóżek w szpitalu, która musi być liczbą dodatnią całkowitą
- liczby wolnych łóżek w szpitalu, która musi być liczbą nieujemną całkowitą

W przypadku obiektów każdy wiersz składa się z:

- id obiketu, które musi być unikalną, nieujemną liczbą całkowitą
- nazwy obiektu, która jest dowolnym ciągiem znaków za wyjątkiem znaku "|"
- współrzędenj w osi x, która oznacza położenie w tej osi danego szpitala, która musi być całkowita
- współrzędenj w osi y, która oznacza położenie w tej osi danego szpitala, która musi być całkowita

W przypadku połączeń dróg każdy wiersz składa się z:

- id drogi, które musi być unikalną, nieujemną liczbą całkowitą
- id szpitala, które musi istenieć w sekcji szpitali
- id szpitala, który musi istenieć w sekcji szpitali
- odległości między szpitalami z poprzednich dwóch punktów, która musi być liczbą całkowitą, dodatnią

Każdy obiekt w linijce jest oddzielony od kolejnego dowolną liczbą spacji przed i po dokładnie jednym znaku "|". Po ostanim obiekcie jak i przed pierwszym nie ma znaku "|", ale może być dowolnie dużo spacji. W pliku niedozwolone są puste linie. Każda sekcja musi zawierać dane z przynajmniej jednym wierszem.

Przykładowy plik wejściowy umieszczono poniżej:

Szpitale

1 | Szpital Wojewódzki nr 997 | 10 | 10 | 1000 | 100

2 | Krakowski Szpital Kliniczny | 100 | 120 | 999 | 99

3 | Pierwszy Szpital im. Prezesa RP | 120 | 130 | 99 | 0

4 | Drugi Szpital im. Naczelnika RP | 10 | 140 | 70 | 1

5 | Trzeci Szpital im. Króla RP | 140 | 10 | 996 | 0

Obiekty

1 | Pomnik Wikipedii | -1 | 50

2 | Pomnik Fryderyka Chopina | 110 | 55

3 | Pomnik Anonimowego Przechodnia | 40 | 70

Drogi

1 | 1 | 2 | 700

2 | 1 | 4 | 550

3 | 1 | 5 | 800

4 | 2 | 3 | 300

5 | 2 | 4 | 550

6 | 3 | 5 | 600

7 | 4 | 5 | 750

3 Opcjonalny plik wejściowy

Oprócz obowiązkowego pliku wejściowego można też podać drugi plik opcjonalny. To czy podaje się go jako wejście programu zależy od tego czy chce się wprowawdzić do programu pacjentów z pliku tekstowego czy za pomocą interfejsu graficznego. Ten plik wejściowy musi być w formacie .txt i składać się z jedego nagłówka i jednej sekcji. Nagłówek ma strukturę jak w pliku obowiązokowym. Sekcja natomiast na składa się z uporządkowanych wierszy według następującej kolejności:

- id pacjenta, które musi być unikalną, nieujemną liczbą całkowitą
- współrzędenj w osi x, która oznacza położenie w tej osi danego pacjenta, która musi być całkowita
- współrzędenj w osi y, która oznacza położenie w tej osi danego pacjenta, która musi być całkowita

Przykładowy plik wygląda następująco:

Pacjenci

1 | 20 | 20

2 | 99 | 105

3 | 23 | 40

4 Uruchomienie programu

Program można uruchomić z terminala. Aby to zrobić trzeba zainstalować na komputerze oprogramowanie języka Java najlepiej w aktualnej wersji. Gdy ten wymóg jest już spełniony należy w lokalizacji pliku ProjektZespolowy.jar otworzyć temrminal i wpisać komendę:

java -jar ProjektZespolowy.jar <lokalizacja obowiązkowego pliku wejściowego>"<nazwa obowiązkowego pliku wejściowego>"

jeśli nie chcemy podawać pacjentów z pliku tylko z interfejsu graficznego. W przeciwnym wypadku należy wpisać komendę:

java -jar ProjektZespolowy.jar <lokalizacja obowiązkowego pliku wejściowego>"<nazwa obowiązkowego pliku wejściowego>" <lokalizacja opcjonalnego pliku wejściowego>"<nazwa opcjonalnego pliku wejściowego>"

Dzięki użyciu cudzysłowia jak we wzorze program się uruchomi nawet, gdy w nazwie pliku wejściowego znajdują się spacje lub znaki specjalne. Poniżej pokazano dwie komendy. Pierszą bez pliku opcjonalnego a drugą z plikiem opcjonalnym:

java -jar ProjektZespolowy.jar /home/"SzpitaleObiektyDrogi.txt"

java -jar ProjektZespolowy.jar /home/"SzpitaleObiektyDrogi.txt" /home/"Pacjenci.txt"

5 Interfejs Graficzny - działanie programu

6 Wynik błędnego działania programu

Gdy plik wejściowy jest niezgodny z opisem z rozdziału 2. lub zostanie podana błędna ścieżka do tego pliku lub on nie istnieje, to program nie wykona swojego zadania a w terminalu pojawi się błąd. Rodzaj komunikatu będzie zależał od tego co zostało źle zrobione. Poniżej wypunktowano możliwe błędy i ich komunikaty:

- brak parametru w komendzie uruchamiającej program w postaci ścieżki do pliku wejściowego:
  Error: not given input file name
- brak pilku wejściowego lub niepoprawna ścieżka dostępu do niego:
  Error: input file not found

- błędny znak w pliku tekstowym:
  Error in line <numer linii>: incorrect char in input file
- brak nagłówka rozpoczynającego się znakiem # :
  Error in line <numer linii>: no section header
- brak danych (jakiejkolwiek linii) w którejś sekcji:
  Error in line <numer linii>: no data in this section
- brak jednej z pozycji w danych:
  Error in line <numer linii>: missing data
- znak "|" po ostatniej spodziewanej pozycji danych:
  Error in line <numer linii>: to many | chars
- niezgodny typ danych:
  Error in line <numer linii>: unexpected type of data
- liczba jest ujemna gdy wymagana jest nieujemna:
  Error in line <numer linii>: the given value should not be negative
- liczba jest niedodatnia, gdy wymagana jest dodatnia:
  Error in line <numer linii>: the given value should be positive
- id nie jest unikalne:
  Error in line <numer linii>: the id is not unique
- w połączeniach jest użyte id, które nie istnieje:
  Error in line <numer linii>: the id does not fit any object
- w połączeniach użyto dwukrotnie (lub więcej) tej samej kombinacji id apteki i producenta:
  Error in line <numer linii>: connection between this pharmacy and manufacturer has been already set
- koszt jednej szczepionki jest podany z trzema lub więcej cyframi po przecinku:
  Error in line <numer linii>: the cost of one vaccine is incorrect
- liczba połączeń nie równa się iloczynowi liczby aptek i producentów:
  Error: missing connections between pharmacy and manufacturer

7 Źródła

Wykorzystane przykłady pliku wejściowych są wersją plików stworzonych przez Pawła Zawadzkiego z Wydziału Elektrycznego Politechniki Warszawskiej w 2020 roku.