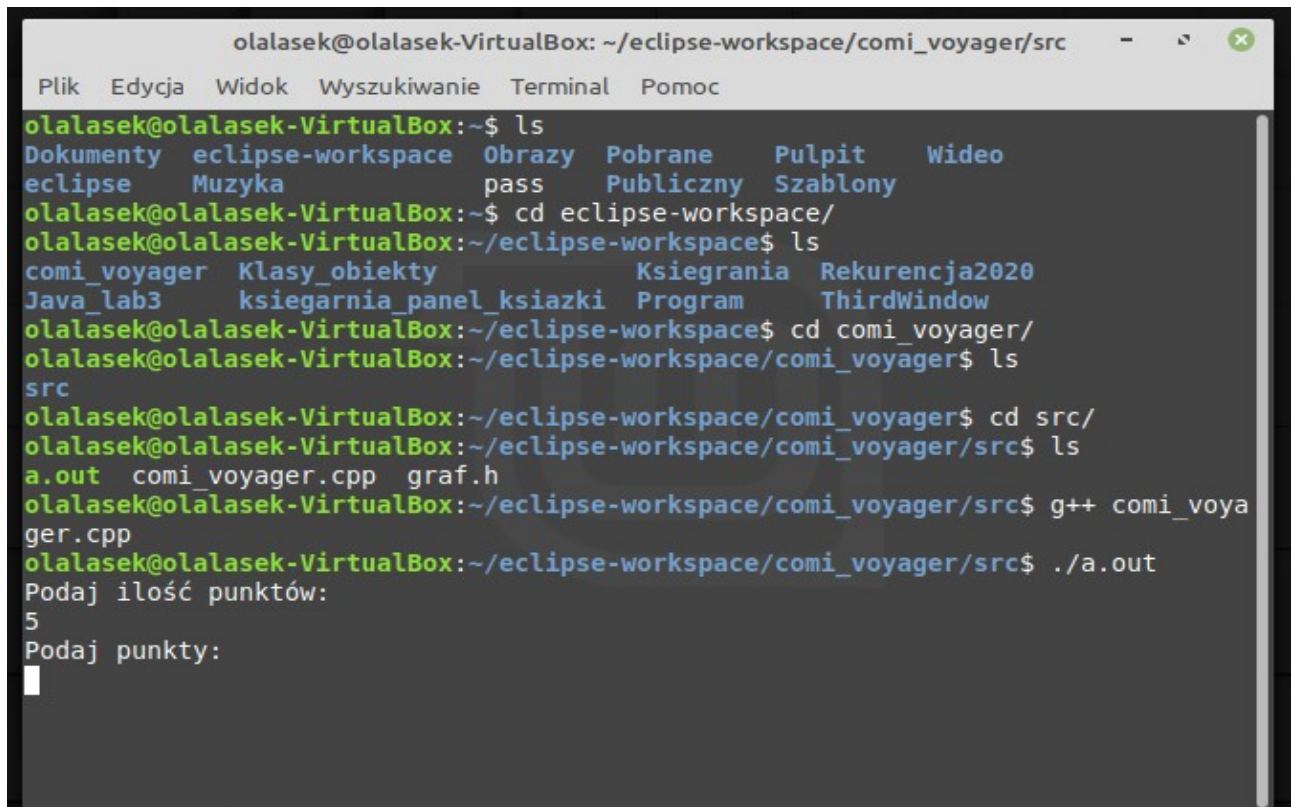


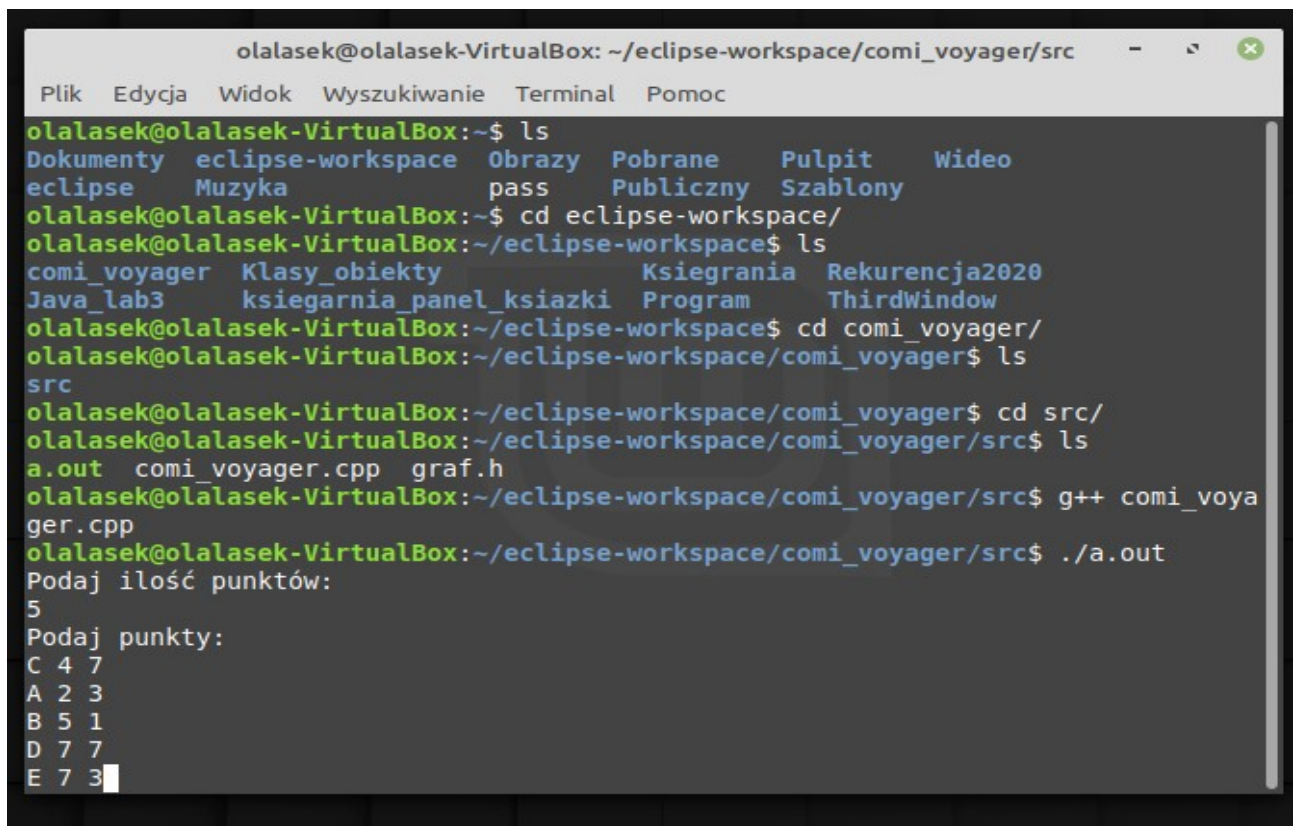
1. Olga Łasek s6054, 2020/2021, Sztuczna inteligencja – laboratorium, prow. Konrad Ożdżyński.
2. Wyznaczanie najkrótszej ścieżki w grafie skierowanym za pomocą algorytmu Dijkstry.
 - a. wykonanie polega na uruchomieniu programu i wprowadzeniu ilości punktów z konsoli



```
olalasek@olalasek-VirtualBox: ~/eclipse-workspace/comi_voyager/src
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc
olalasek@olalasek-VirtualBox:~$ ls
Dokumenty  eclipse-workspace  Obrazy  Pobrane  Pulpit  Wideo
eclipse    Muzyka             pass    Publiczny Szablony
olalasek@olalasek-VirtualBox:~$ cd eclipse-workspace/
olalasek@olalasek-VirtualBox:~/eclipse-workspace$ ls
comi_voyager  Klasy_obiekty  Ksiegrania  Rekurencja2020
Java_lab3     ksiegarnia_panel_ksiazki  Program      ThirdWindow
olalasek@olalasek-VirtualBox:~/eclipse-workspace$ cd comi_voyager/
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager$ ls
src
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager$ cd src/
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager/src$ ls
a.out  comi_voyager.cpp  graf.h
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager/src$ g++ comi_voya
ger.cpp
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager/src$ ./a.out
Podaj ilość punktów:
5
Podaj punkty:

```

- b. wprowadzenie współrzędnych punktów w formie: podanie znaku punktu typu char oraz po spacji podanie współrzędnej x i y. Punkty podajemy w dowolnej kolejności, oprócz pierwszego punktu startowego.



```
olalasek@olalasek-VirtualBox: ~/eclipse-workspace/comi_voyager/src
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc
olalasek@olalasek-VirtualBox:~$ ls
Dokumenty  eclipse-workspace  Obrazy  Pobrane  Pulpit  Wideo
eclipse    Muzyka             pass    Publiczny Szablony
olalasek@olalasek-VirtualBox:~$ cd eclipse-workspace/
olalasek@olalasek-VirtualBox:~/eclipse-workspace$ ls
comi_voyager  Klasy_obiekty  Ksiegrania  Rekurencja2020
Java_lab3     ksiegarnia_panel_ksiazki  Program      ThirdWindow
olalasek@olalasek-VirtualBox:~/eclipse-workspace$ cd comi_voyager/
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager$ ls
src
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager$ cd src/
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager/src$ ls
a.out  comi_voyager.cpp  graf.h
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager/src$ g++ comi_voya
ger.cpp
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager/src$ ./a.out
Podaj ilość punktów:
5
Podaj punkty:
C 4 7
A 2 3
B 5 1
D 7 7
E 7 3

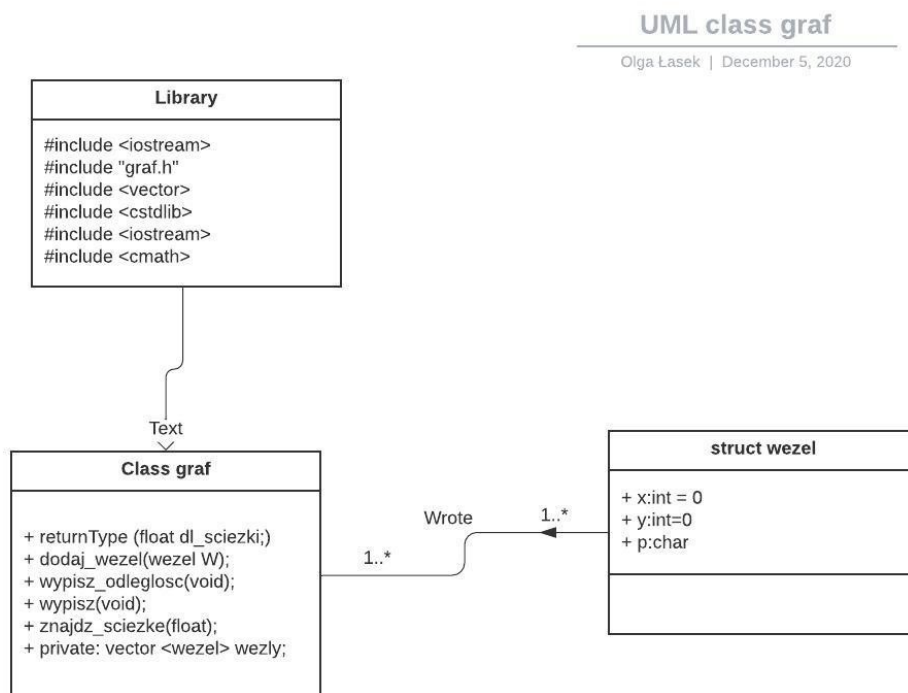
```

c. program podaje długość najkrótszej ścieżki oraz kolejność odwiedzonych punktów w grafie

```
olalasek@olalasek-VirtualBox: ~/eclipse-workspace/comi_voyager/src
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc
src
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager$ cd src/
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager/src$ ls
a.out  comi_voyager.cpp  graf.h
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager/src$ g++ comi_voya
ger.cpp
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager/src$ ./a.out
Podaj ilość punktów:
5
Podaj punkty:
C 4 7
A 2 3
B 5 1
D 7 7
E 7 3
Najkrótsza ścieżka:
13.434
Graf:
C 4 7
D 7 7
E 7 3
B 5 1
A 2 3
olalasek@olalasek-VirtualBox:~/eclipse-workspace/comi_voyager/src$
```

Wynik podaje dla punktów podanych w poleceniu, dla mojego numeru indeksu.

3. Diagram UML



4. Wzór do obliczenia długości odcinka
 $A(x_1, y_1)$, $B(x_2, y_2)$

$$\text{Długość odcinka } AB = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

5. Opis algorytmu Dijkstry

Mając dany graf z wyróżnionym wierzchołkiem (*źródłem*) algorytm znajduje odległości od źródła do wszystkich pozostałych wierzchołków. Łatwo zmodyfikować go tak, aby szukał wyłącznie (najkrótszej) ścieżki do jednego ustalonego wierzchołka, po prostu przerywając działanie w momencie dojścia do wierzchołka docelowego, bądź transponując tablicę incydencji grafu.

Algorytm Dijkstry znajduje w grafie wszystkie najkrótsze ścieżki pomiędzy wybranym wierzchołkiem a wszystkimi pozostałymi, przy okazji wyliczając również koszt przejścia każdej z tych ścieżek.

Algorytm Dijkstry jest przykładem algorytmu zachłannego.