

PWR-Racing-Team-2024

Wiktor Lepak

ZADANIE 2

W swoim programie dokonałem próby implementacji własnej wersji algorytmu Jarvisa pozwalającego znaleźć otoczkę wypukłą zbioru punktów. Oryginalnie, algorytm Jarvisa polega na wybraniu ze zbioru punktów, punktu najbardziej wysuniętego do lewej strony, a następnie znalezienia punktu, z którym utworzy największy kąt liczony przeciwnie do ruchu wskazówek zegara, czyli znajduje się najbardziej „na zewnątrz” od aktualnie badanego punktu. Po znalezieniu odpowiedniego połączenia, należy przenieść swoje działania na nowo wybrany punkt i z niego kontynuować ten sam schemat działania, aż do powrotu do punktu startowego. Popularną analogią tego algorytmu jest przykład wbicia kilku gwoździ w deskę, a następnie owinięcie gwoździ dookoła za pomocą sznurka. Po napięciu sznurka utworzy się wielokąt, który będzie pełnił funkcję otoczki wypukłej.

Niestety, przez brak czasu i pomysłu, nie udało mi się sfinalizować swojej wizji, mimo to chciałbym podzielić się swoim pomysłem. Program napisany moją metodą, poprawnie wybiera pierwszy punkt. Nie potrafię zamknąć go jednak w odpowiedniej pętli, która pozwoli na poprawny wybór kolejnych punktów. Użyta przeze mnie w programie pętla jest prowizoryczna i ma jedynie charakter testowy. W swoim zamyśle, chciałem zapisać punkty tworzące obrys w wektorze, którego zawartość miała być następnie wyświetlana na końcu działania programu. Obecna budowa pętli nie pozwala mi jednak na skuteczne jego użycie. Nie chciałem korzystać z gotowych rozwiązań i przetartych schematów, jednak stworzenie własnego trochę mnie przerosło.

Plik	Edytuj	Wyświetl
8		
2 3		
1 0		
1 2		
5 3		
2.5 2		
-1 2		
4 -1		
3 1		

Przy testach używałem danych podanych w zadaniu zapisanych w odpowiednim formacie.

```
Liczba punktow wynosi: 8
(2,3)
(1,0)
(1,2)
(5,3)
(2.5,2)
(-1,2)
(4,-1)
(3,1)
Punkt wychylony maksymalnie w lewo : (-1,2)
Pierwsze polaczenie: (2,3)
Polaczenia tworzace obrys:
(-1,2)
(2,3)
(1,0)
(1,2)
(5,3)
(2.5,2)
(-1,2)
(4,-1)
(3,1)
```

Tak prezentują się dane otrzymane po kompilacji:

-W pierwszym wierszu znajduje się informacja, ile punktów odczytano z pliku. Następnie punkty te, są wypisywane w kolejności w jakiej zostały odczytane.

-Po wypisaniu wszystkich punktów, pojawia się informacja o punkcie położonym maksymalnie po lewej stronie, oraz pierwsze połączenie stworzone od tego punktu.

-W ostatniej części wypisywane są kolejne nawiązane połączenia, **niestety niepoprawne**.