

Algorytmy i struktury danych, Teleinformatyka, I rok

Raport z laboratorium nr: 4

Imię i nazwisko studenta: Amadeusz Gunia

1. W pole poniżej wklej najważniejszy (według Ciebie) fragment kodu źródłowego z zajęć (maksymalnie 15 linii).

```
1.prev=0
2.for i in range (99):
3.    min=1000000
4.    for j in range (100):
5.        r=sqrt(((X[j]-X[prev])**2)+((Y[j]-Y[prev])**2))
6.        if r<=min and M[j] not in U:
7.            min=r
8.            min_index=j
9.U.append(M[min_index])
10.prev=min_index
11.trasa=trasa+min
12.sciezka=sciezka+(M[min_index])+'->'
13.
14.
15.
```

Uzasadnij swój wybór.

Uważam, że ten fragment kodu jest najważniejszy, gdyż w każdym kroku sprawdza on odległości między aktualnym miastem a innymi miastami i wybiera jako kolejne to miasto, do którego jest najkrótsza odległość. Dzięki temu uzyskujemy najkrótszą trasę przejścia po wszystkich miastach.

2. Podsumuj wyniki uzyskane podczas wykonywania ćwiczenia. Co ciekawego zauważyłeś? Czego się nauczyłeś? Jeśli instrukcja zawierała pytania, odpowiedz na nie. Do sprawozdania możesz dodać wykresy jeśli jest taka potrzeba.

W tym ćwiczeniu wykonałem dwa algorytmy, które obliczają długość ścieżki komiwojażera. W pierwszym algorytmie ścieżka prowadzi po wszystkich miastach po kolei według listy z pliku. W tym wariantcie długość ścieżki wynosi ponad 5084 jednostki. Algorytm ten jest

W drugim algorytmie ścieżka jest wybierana na podstawie odległości między aktualnym miastem w danym kroku a innymi miastami, których jeszcze nie odwiedziliśmy i jako kolejne miasto na trasie wybierane jest to, do którego odległość jest najkrótsza. Miastem początkowym jest miasto nr 1. Długość ścieżki wyznaczonej tym algorytmem wynosi około 992 jednostki, więc jest to trasa ponad 5 razy krótsza, niż ta wyznaczona przez algorytm pierwszy. Po przejściu wszystkich miast, będąc w ostatnim mieście naszej trasy, powracamy bezpośrednio do miasta nr 1. Czas wykonania tego algorytmu wyniósł 0,0217885s. Czas wykonania drugiego algorytmu jest ponad 63 razy dłuższy lecz nadal są to jedynie ułamkowe

TRASA "miasta po kolei"

Trasa komiwojazera to: 1->2->3->4->5->6->7->8->9->10->11->12->13->14->15->16->17
->18->19->20->21->22->23->24->25->26->27->28->29->30->31->32->33->34->35->36->37
->38->39->40->41->42->43->44->45->46->47->48->49->50->51->52->53->54->55->56->57
->58->59->60->61->62->63->64->65->66->67->68->69->70->71->72->73->74->75->76->77
->78->79->80->81->82->83->84->85->86->87->88->89->90->91->92->93->94->95->96->97
->98->99->100->1

Dlugosc trasy komiwojazera wyniosla: 5084.461369051879

Czas wykonywania algorytmu: 0.00034460000000000046

TRASA "optymalna"

Trasa komiwojazera to: 1->86->33->48->90->52->40->84->13->50->47->5->21->12->66
->35->49->29->96->71->36->65->19->27->23->75->70->46->74->15->76->63->7->3->43
->64->62->45->51->97->93->6->77->88->85->25->26->22->89->79->9->10->83->100->73
->99->53->58->67->32->14->54->91->18->41->61->24->16->92->78->2->30->37->42->38
->17->11->80->28->94->44->81->69->39->55->95->20->59->4->56->31->87->72->57->82
->34->98->8->68->60->1

Dlugosc trasy komiwojazera wyniosla: 992.3393569059313

Czas wykonywania algorytmu: 0.021788500000000002