

## PEA – projekt nr 2

### Temat: Implementacja i analiza efektywności algorytmu Tabu Search i/lub Symulowanego Wyżarzania dla problemu komiwojażera

Należy zaimplementować oraz dokonać analizy efektywności algorytmu **Tabu Search (TS)** i/lub **Symulowanego Wyżarzania (SW)** dla problemu komiwojażera (TSP).

Należy porównać rozwiązanie dostarczone przez algorytm z najlepszymi znanymi rozwiązaniami dla przykładów testowych.

Podczas realizacji zadania należy przyjąć następujące założenia:

- używane struktury danych powinny być alokowane dynamicznie (w zależności od aktualnego rozmiaru problemu),
- program powinien umożliwić wczytanie danych testowych z pliku - te pliki to: ftv55.atsp (1608), ftv170.atsp (2755) , rgb358.atsp (1163). W nawiasach podano najlepsze znane rozwiązanie dla danych zawartych w pliku - długość drogi.
- opis formatu pliku z danymi znajduje się na stronie: <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/>
- na stronie <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/atsp/> znajdują się wyżej wymienione pliki
- na stronie <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/XML-TSPLIB/instances/> znajdują się wersje XML-owe w/w plików – można je stosować zamiast plików w poprzednim formacie (za to jest dodatkowa premia)
- program musi umożliwiać wprowadzenia kryterium stopu jako czasu wykonania podawanego w sekundach,
- implementacje algorytmów należy dokonać zgodnie z obiektywnym paradygmatem programowania,
- używanie „okienek” nie jest konieczne i nie wpływa na ocenę (wystarczy wersja konsolowa),
- kod źródłowy powinien być komentowany.

Menu programu powinno zawierać następujące opcje:

- 1.Wczytanie danych z pliku (wspólne dla obu algorytmów)
- 2.Wprowadzenie kryterium stopu (wspólne dla obu algorytmów)
- 3.Obliczenie rozwiązania metodą zachłanną (przetestować wszystkie wierzchołki startowe i wybrać najlepsze rozwiązanie)

4. Wybór sąsiedztwa (dla TS jeśli zaimplementowano)
5. Uruchomienie algorytmu TS dla wczytanych danych i ustawionych parametrów i wyświetlenie wyników (wartość drogi + ciąg wierzchołków)
6. Ustawienie współczynnika zmiany temperatury dla SW
7. Uruchomienie algorytmu SW dla wczytanych danych i ustawionych parametrów i wyświetlenie wyników (wartość drogi + ciąg wierzchołków). Na końcu działania algorytmu wyświetlić wartość wyrażenia  $\exp(-1/T_k)$  oraz  $T_k$ , gdzie  $T_k$  jest temperaturą końcową
8. Zapis ścieżki rozwiązania do pliku txt
9. Wczytanie ścieżki rozwiązania z pliku txt i obliczenie na drogi na podstawie wczytanej tabeli kosztów w p.1.

#### **UWAGA 1!**

1. Raz ustawione parametry obowiązują do ich zmiany (chodzi o to, aby nie ustawiać ich ponownie przy wczytywaniu nowych danych jak i nie wczytywać ponownie danych przy zmianie parametrów).
2. Rozwiązanie początkowe wygenerować za pomocą metody zachłannej.
3. W trakcie działania algorytmu zawsze pamiętać najlepsze rozwiązanie.
4. Format zapisu ścieżki (wyniku) jest następujący:
  - w pierwszej linii ilość wierzchołków
  - w kolejnych liniach numery wierzchołkówŚcieżka zapisana w pliku ma być kompletna tzn. powinna się zapętląć.

#### **TS:**

5. Algorytm ma uwzględniać mechanizm dywersyfikacji przeszukiwania przestrzeni rozwiązań jak również różne definicje sąsiedztwa (typy ruchów).

#### **SW:**

6. Sposób zmiany temperatury (schładzania) zrealizować wg wzoru  $T(i+1) = a * T(i)$ , gdzie  $a$  jest stałą
7. Wybrać jeden ze sposobów definicji sąsiedztwa.
8. Obliczyć temperaturę początkową w oparciu o przetwarzane dane – zaproponować sposób liczenia i umieścić go w sprawozdaniu – temp. początkowa nie może być wpisana na sztywno do programu
9. W sprawozdaniu umieścić temperaturę końcową (wynikającą z działania algorytmu)
10. Starać się optymalnie dobrać wsp.  $a$ , aby schładzanie nie odbywało się za szybko (zwłaszcza przy wersji z jedną wartością wsp. schładzania).

Sprawozdanie powinno zawierać:

- a) wstęp teoretyczny zawierający opis algorytmów i same algorytmy (w postaci pseudokodu lub graficznie), omówienie jego elementów (lista tabu, dywersyfikacja, opis (definicja) sąsiedztwa itd.) dla TS; analogicznie dla SW.
- b) opis najważniejszych klas w projekcie
- c) wyniki w postaci tabel - dla podanych 3 plików wykonać pomiary błędu względnego (wyrażonego go w %) w funkcji czasu wykonywania algorytmu (dla każdego parametru osobno) – ustalić naszytywno czas wykonywania algorytmów na np. 2 minuty dla rozmiaru małego, 4- średniego, 6- dużego. Dla każdego z plików i dla każdego algorytmu i każdego parametru algorytmu uruchomić algorytm 10 razy i w tabeli przedstawić dla każdego uruchomienia najlepsze rozwiązanie (wartość drogi) i moment (czas) kiedy zostało odnalezione.
- d) narysować wykres błędu w funkcji czasu dla najlepszego rozwiązania dla każdego pliku i każdej wartości parametru algorytmu (dla TS: jeżeli wybrano 3 sąsiedztwa to umieścić je na jednym wykresie, analogicznie dla SW jeśli wybrano 3 różne wartości współczynnika zmiany temperatury).
- e) dla najlepszego rozwiązania dla każdego z plików (z obu algorytmów, jeśli implementowano oba) umieścić w sprawozdaniu ścieżkę i zapisać ją pliku na dysku w formacie wg UWAGI 1. Nazwa pliku z rozwiązaniem powinna zawierać nazwę pliku źródłowego oraz nazwę algorytmu (SW, TS) np. sw\_ftv170.txt
- f) jeżeli wybrano dwa algorytmy to porównać je (w tym celu oba algorytmy dla określonego pliku z danymi powinny wykonywać się tyle samo czasu)

Błąd względny to  $|f_{zn} - f_{opt}|/f_{opt}$ , gdzie  $f_{zn}$  – wartość obliczona przez nasz algorytm,  $f_{opt}$  – wartość optymalna – najlepsze znane rozwiązanie (podane wyżej)

Ocena:

- a) algorytm tabu z dywersyfikacją i jedną definicją sąsiedztwa – 1.0
- b) algorytm tabu z dywersyfikacją i trzema definicjami sąsiedztwa – 1.4
- c) algorytm SW dla jednego schematu schładzania (wsp. a) – 1,0
- d) algorytm SW dla trzech schematów schładzania (wsp. a) – 1,4
- e) bonus za wczytanie pliku z danymi w formacie XML - 0,2

c.d. na następnej stronie

### **UWAGA 2!**

Na dysku google'a należy zamieścić projekt oraz najlepsze rozwiązanie dla danego algorytmu w postaci pliku tekstowego (format podano w UWAGI 1) oraz sprawozdanie w wersji elektronicznej (ale papierową wersję też należy dostarczyć)

### **UWAGA 3!**

Algorytm TS musi zawierać elementy losowości – nie może to być algorytm deterministyczny.

### **UWAGA 4!**

Jeżeli wybrano dwa algorytmy, to czasy eksperymentów dla obu algorytmów powinny być takie same

### **UWAGA 5!!!**

**Jeżeli najlepsze rozwiązanie za pomocą danego algorytmu jest takie same lub gorsze od metody zachłannej to obniżam ocenę za algorytm.**

### **Dodatkowe materiały internetowe:**

Z.Michalkiewicz, D.B.Vogel, *Jak to rozwiązać czyli nowoczesna heurystyka*, WNT 2006

<http://www2.imm.dtu.dk/courses/02719/tabu/4tabu2.pdf>

[http://www.pi.zarz.agh.edu.pl/intObl/notes/IntObl\\_w2.pdf](http://www.pi.zarz.agh.edu.pl/intObl/notes/IntObl_w2.pdf)

[http://155.158.112.25/~algorytmyewolucyjne/materialy/algorytm\\_symulowanego\\_wywarzania.pdf](http://155.158.112.25/~algorytmyewolucyjne/materialy/algorytm_symulowanego_wywarzania.pdf)