

Projektowanie efektywnych algorytmów – projekt – zasady zaliczania

W ramach zajęć należy zrealizować przedstawione przez prowadzącego zadania projektowe. Zadania realizowane są w grupach jednoosobowych.

Z wykonania każdego zadania należy sporządzić sprawozdanie. Zawartość sprawozdania jest opisana w ramach każdego zadania. Na pierwszej stronie sprawozdania MUSZA być podane następujące informacje: imię, nazwisko i numer indeksu autora, nr zadania (zadania są numerowane od 1 do 3) oraz przynależność do grupy projektowej (termin np. piątek 9.00-11.00) wraz z nazwiskiem prowadzącego.

Sprawozdanie należy oddać PROWADZĄCEMU w wersji papierowej. Do sprawozdania należy dołączyć wersję elektroniczną programów (kody + wersja wykonywalna).

Proponuje się następujące terminy oddawania sprawozdań z poszczególnych zadań projektowych (w szczególnych przypadkach prowadzący może zmienić podane terminy):

1. 13 listopad 2018 dla grup wtorkowych oraz 15 listopad dla grup czwartkowych
2. 18 grudzień 2018 dla grup wtorkowych oraz 20 grudzień dla grup czwartkowych
3. 15 styczeń 2018 dla grup wtorkowych oraz 17 styczeń dla grup czwartkowych

Każdy tydzień spóźnienia oznacza obniżenie oceny o 0.5! Oczywiście zadanie można oddać wcześniej (za oddanie wcześniej o tydzień zwiększa się ocenę o 0,25 a za jeszcze wcześniej o 0,5), ale aby dostać ocenę na koniec 5,5 należy uzyskać sumaryczną ocenę z wszystkich projektów równą 14.5pkt (oceny bez bonusów za termin) oraz zrealizować zadanie dodatkowe. Ocena za pojedyncze zadanie projektowe wystawiana jest w sposób ciągły od 2 do wartości maksymalnej (określonej przez wariant zadania)

Ocena końcowa z projektu stanowi średnią arytmetyczną ocen z poszczególnych zadań z uwzględnieniem spóźnień wg wzoru: $(z1+z2+z3) / 3$

Warunkiem koniecznym zaliczenia projektu jest uzyskanie oceny pozytywnej oceny końcowej (tzn. ≥ 3.0). Ostateczny termin oddania zadania pierwszego upływa na tydzień przed terminem oddania zadania drugiego. Ostateczny termin oddania zadania drugiego upływa tydzień wcześniej przed oddaniem zadania trzeciego. Ostateczny termin zadania trzeciego wyznacza data ostatnich zajęć. Nieoddanie zadania w w/w terminach skutkuje niezaliczeniem przedmiotu.

Zajęcia z projektu trwają 2 godziny lekcyjne bez przerwy - 1,5 godziny zegarowej.

Wszelkie sprawy z projektem załatwiamy tylko na godzinach projektu (a nie konsultacji).

Przy oddawaniu projektu wymagana jest znajomość problemu (student będzie odpytywany z tej znajomości), którego dotyczy jak i programu. **Jeżeli student zostanie złapany na nieznajomości programu, który oddaje, to wówczas traktuje się to jako fałszerstwo i stawia się ocenę niedostateczną na zaliczenie (prowadzący nie ma prawa wiedzieć więcej niż student na temat jego programu).** Podobnie traktuje się osobę, która odda program prawie identyczny jak ktoś wcześniej. Twierdzenie, iż ktoś pomagał przy pisaniu nie jest wytłumaczeniem. Zabrania się kopiować programów z internetu.

Ogólnie o zadaniach z PEA

W ramach projektu należy zrealizować trzy poniższe zadania. Każde zadanie polega na zaimplementowaniu i przetestowaniu algorytmu dla **asymetrycznego** problemu komiwojażera. W kolejnych sprawozdaniach należy porównać algorytmy między sobą, tzn. porównać ich czasy działania oraz uzyskiwane wartości funkcji celu. Każde zadanie będzie szczegółowo opisane w wymaganiach.

Algorytmy do implementacji i testowania w zadaniach

Zadanie 1. Metoda przeszukiwania zupełnego, programowania dynamicznego oraz podziału i ograniczeń (ang. brute force, dynamic programming, branch and bound).

Zadanie 2. Algorytm przeszukiwania z zakazami (ang. tabu search) oraz symulowanego wyżarzania

Zadanie 3. Algorytm genetyczny (ang. genetic algorithm)

Zadanie 4 dodatkowe na 5,5. Algorytm mrówkowy lub bazujący na sztucznym systemie immunologicznym. Szczegóły dotyczące tego zadania należy konsultować z prowadzącym.

Literatura do zadań 1-3

1. D.E. Goldberg, Algorytmy genetyczne i ich zastosowania, Warszawa, WNT 1998.
2. Z Michalewicz, Algorytmy genetyczne + struktury danych = programy ewolucyjne, Warszawa, WNT 1996.
3. Z. Michalewicz, D.B. Fogel, Jak to rozwiązać, czyli nowoczesna heurystyka, WNT 2006.
4. V. Cerny, A thermodynamical approach to the travelling salesman problem: an efficient simulation algorithm, Journal of Optimization Theory and Applications, 45: 41-51, 1985.

Literatura do zadania 4.

1. M. Dorigo, T. Stutzle, Ant Colony Optimization, MIT Press 2004.
2. M. Dorigo, G. Di Caro, L.M. Gambardella, Ant Algorithms for Discrete Optimization. Artificial Life, 5(2): 137-172, 1999.
3. S.T. Wierzchoń, Sztuczne systemy immunologiczne. Teoria i Zastosowania, EXIT 2001.
4. L.N. De Castro, J.I. Timmis, Artificial Immune Systems as a Novel Soft Computing Paradigm, Soft Computing Journal, vol 7, 2003.