

Termin odesłania **10.01.2022 (pon) godz. 16.15** na platformie **Ms Teams** (we właściwym zespole **lab** przypisanym dla przedmiotu **Programowanie Matematyczne**). **Opóźnione** przesłanie rozwiązania zadania będzie rozliczane zgodnie z regulaminem przedmiotu.

Kolejny ostatni termin zadania jest **14.01.2022 (pt)**

Rozwiązanie zadania tj. wszystkie źródłowe m-pliki, raport (*obowiązkowy, zawierający oświadczenie o samodzielności*) w formacie **zip** o nazwie **pm6d_swojeimie_swojenazwisko.zip**

Raport (plik **pdf**) powinno być w formacie **A4** i powinno obejmować:

Dane studenta (imię, nazwisko, grupa, data)

Treść zadania (postać rozwiązywanego problemu)

Opis kroków przekształcania zadania, krótki opis algorytmu

Ciekawe przykłady obliczeniowe (również dodatkowo wskazane w treści zadania)

Analizę (omówienie) wyników obliczeniowych, testów

Ponadto należy załączyć:

Kody źródłowe wszystkich funkcji/procedur i skryptów (**brak** kompletu jest traktowany jak **brak** przesłania zadania w terminie)

Napisz **skrypt**, w którym proszę wykonać całe zadanie i wywołać odpowiednie funkcje.

1 pkt

Problem

Znaleźć **rzut punktu** $p \in R^n$ na zbiór Ω (czyli **najbliższy od punktu p, punkt** spełniający ograniczenia równościowe)
 $\Omega = \{ x \in R^n: Ax = b, A \in R^{m \times n}, m < n, r(A) = m, b \in R^m \}$

- **Podaj postać zadania** programowania kwadratowego z ograniczeniami równościowymi.
Czy zadanie posiada jakieś własności?

Dla macierzy **A** oraz **b** wylosuj losowe wartości **całkowite**, np. z przedziału [-5; 5]

Punkt **p** również należy wylosować, ustal parametry?

Wykonaj testy np. dla **n=10** (liczba zmiennych), **m=3,5,7** (liczba ograniczeń równościowych)

Może inne większe wartości **n** oraz **m**? jakie?

Rozwiązać problem wykorzystując funkcję **quadprog**

x=?

2,5 pkt

Rozwiązać problem za pomocą własnej funkcji wykorzystującej **metodę zewnętrznej kwadratowej funkcji kary**
[xx,exitflag,it]=ZFK(jakieś parametry,x0,e)

e=1e-8 (lub 1e-6, 1e-5 ? przetestuj jaka jest ok?) parametr definiujący dokładność obliczeń (zbieżność, itp.)

Do rozwiązania **zadania pomocniczego** w kolejnych iteracjach wykorzystaj funkcję **fminunc**

W **optimoptions** ustaw:

SpecifyObjectiveGradient: true

OptimalityTolerance: 1e-8 (przetestuj jaka?)

StepTolerance: 1e-8 (przetestuj jaka?)

Porównaj wyniki własne z uzyskanymi przez funkcję **quadprog**. Oblicz normę różnicy rozwiązań.

Uzasadnij na podstawie WKT czy uzyskany z algorytmu **xx** jest **RO**? Czy to zadanie zawsze posiada RO?

Zbadaj **skuteczność algorytmu**, tj. dla kolejnych **N=100** losowań oblicz powyższą normę.

1,5 pkt

Zamiast wywołania **fminunc** zastosuj **własną funkcję** wykorzystującą **algorytm DFP** (z ostatnich zajęć). Wybierz algorytm minimalizacji kierunkowej (z ostatnich zajęć, **który?**)

Opis testów

Wnioski