

PLAN TESTÓW

Programu do obliczania najmniejszej wartości funkcji przy pomocy
algorytmu Harmony Search

1. Cel dokumentu

Celem niniejszego dokumentu jest dostarczenie informacji o przeprowadzeniu oraz dokumentacji testów oprogramowania do liczenia najmniejszej wartości. W dalszej części zostaną przedstawione:

- Przedmiot testów wraz z jego opisem,
- Elementy testowe,
- Techniki projektowania testów,
- Wymagania dotyczące danych testowych,
- Środowisko testowe i jego wymagania,
- Scenariusze testowe,
- Lista narzędzi wykorzystanych w procesie testowania,
- Raportowanie testów.

Na końcu dokumentu powinien znaleźć się harmonogram przeprowadzenia testów, ale z uwagi na niewielki rozmiar projektu (i to, że jest to tylko próbny dokument) nie zamieszczono go.

2. Przedmiot testów

Przedmiotem testów jest weryfikacja funkcjonalności oraz przeprowadzenie testów niefunkcjonalnych oprogramowania do liczenia najmniejszej wartości funkcji w wybranym przedziale.

Program ten oparty jest o algorytm Harmony Search, który polega na losowym dobieraniu współczynników w oparciu o wcześniejszą znajomość najlepszych obliczonych kombinacji. Program potrafi wyliczyć najmniejszą wartość dla funkcji od 2 do 5 zmiennych. Do poprawnego działania algorytm potrzebuje wzór funkcji, wartości minimalne i maksymalne współczynników, ilość iteracji, rozmiar pamięci (HMS), współczynnik losowości (HMRC), współczynnik losowości (PAR) oraz przedział modyfikacji współczynników (R). Dodatkowo podczas przeliczania wartości program rysuje plansze z najlepszymi wynikami, które aktualnie trzyma w pamięci, a na koniec wyświetla tabele z pamięcią o zawartości od 2 do 5 współczynników oraz wynikiem danej funkcji dla tych współczynników.

3. Elementy testowe

Testom funkcjonalnym i niefunkcjonalnym podlegać będą:

- Graficzny interfejs,
- Wszystkie przyciski oraz pola do wprowadzania potrzebnych elementów,
- Poprawne działanie algorytmu,
- Okno z rysowanymi wartościami w pamięci,
- Okno z wynikami algorytmu,

4. Techniki projektowania testów

Testy jednostkowe zostały przeprowadzone na 2 plikach/modułach. Jeden odpowiedzialny za parsowanie funkcji wpisanej do programu. Drugi odpowiada za obliczanie wartości danej funkcji dla odpowiednich współczynników. Testy te znajdują się w plikach `test_solver_module.py` oraz `test_parser_module.py`

Do testów czarnoskrzynkowych wykorzystano technikę scenariuszy. Scenariusze znajdują się w punkcie 7.

5. Wymagania dotyczące danych testowych

Do przeprowadzenia testów czarnoskrzynkowych potrzebne będą wartości:

- Funkcja: $x_1^2 + x_2^2$
- Ilość zmiennych: 2
- X_1 min: -3
- X_1 max: 3
- X_2 min: -3
- X_2 max: 3
- Iteracje: 1000
- HMS: 10
- HMRC: 0.8
- PAR: 0.5
- R: 0.7

6. Środowisko testowe

Program działa tylko i wyłącznie na komputerze dlatego testy zostaną wykonane dla jednego środowiska testowego.

System operacyjny	Windows 10 Home
Procesor	Intel(R) Core(TM) i5-9300H CPU @ 2.40GHz 2.40 GHz
Pamięć systemu operacyjnego	16,0 GB

7. Scenariusze testowe

Scenariusz nr 1 „Wprowadzenie danych do interfejsu”	
Przypadek 1a:	„Użytkownik poprawnie wprowadza wszystkie dane”
Warunek wstępny:	Użytkownik posiada dane wymienione w punkcie 5
Kroki testowe:	1. Włączenie programu 2. Wprowadzenie wszystkich danych wejściowych w odpowiednie pola 3. Kliknięcie przycisku „Solve”
Oczekiwany rezultat:	Pojawia się okno z planszą, na której przemieszczają się kropki. Powinny one dążyć do pozycji (0, 0). Następnie po przeliczeniu odpowiedniej ilości iteracji pojawia się kolejne okno z tabelą przedstawiającą wyniki obliczeń. Najlepszy rezultat powinien wynieść w przybliżeniu 0.
Przypadek 1b:	„Użytkownik poprawnie wprowadza wszystkie dane poza ilością iteracji”
Warunek wstępny:	Użytkownik posiada dane wymienione w punkcie 5
Kroki testowe:	1. Włączenie programu 2. Wprowadzenie wszystkich danych wejściowych w odpowiednie pola poza iteracjami, tam należy zostawić puste pole 3. Kliknięcie przycisku „Solve”
Oczekiwany rezultat:	Pojawia się okno z ostrzeżeniem „Please check your inputs”
Przypadek 1c:	„Użytkownik poprawnie wprowadza wszystkie dane poza HMS”
Warunek wstępny:	Użytkownik posiada dane wymienione w punkcie 5
Kroki testowe:	1. Włączenie programu 2. Wprowadzenie wszystkich danych wejściowych w odpowiednie pola poza HMS, tam należy zostawić puste pole 3. Kliknięcie przycisku „Solve”
Oczekiwany rezultat:	Pojawia się okno z ostrzeżeniem „Please check your inputs”

Przypadek 1d:	„Użytkownik poprawnie wprowadza wszystkie dane poza HMRC”
Warunek wstępny:	Użytkownik posiada dane wymienione w punkcie 5
Kroki testowe:	<ol style="list-style-type: none"> 1. Włączenie programu 2. Wprowadzenie wszystkich danych wejściowych w odpowiednie pola poza HMRC, tam należy zostawić puste pole 3. Kliknięcie przycisku „Solve”
Oczekiwany rezultat:	Pojawia się okno z ostrzeżeniem „Please check your inputs”
Przypadek 1e:	„Użytkownik poprawnie wprowadza wszystkie dane poza PAR”
Warunek wstępny:	Użytkownik posiada dane wymienione w punkcie 5
Kroki testowe:	<ol style="list-style-type: none"> 1. Włączenie programu 2. Wprowadzenie wszystkich danych wejściowych w odpowiednie pola poza PAR, tam należy zostawić puste pole 3. Kliknięcie przycisku „Solve”
Oczekiwany rezultat:	Pojawia się okno z ostrzeżeniem „Please check your inputs”
Przypadek 1f:	„Użytkownik poprawnie wprowadza wszystkie dane poza R”
Warunek wstępny:	Użytkownik posiada dane wymienione w punkcie 5
Kroki testowe:	<ol style="list-style-type: none"> 1. Włączenie programu 2. Wprowadzenie wszystkich danych wejściowych w odpowiednie pola poza R, tam należy zostawić puste pole 3. Kliknięcie przycisku „Solve”
Oczekiwany rezultat:	Pojawia się okno z ostrzeżeniem „Please check your inputs”
Przypadek 1g:	„Użytkownik poprawnie wprowadza wszystkie dane poza funkcją”
Warunek wstępny:	Użytkownik posiada dane wymienione w punkcie 5
Kroki testowe:	<ol style="list-style-type: none"> 1. Włączenie programu 2. Wprowadzenie wszystkich danych wejściowych w odpowiednie pola poza funkcją, tam należy zostawić puste pole 3. Kliknięcie przycisku „Solve”
Oczekiwany rezultat:	Pojawia się okno z ostrzeżeniem „Please check your inputs”
Scenariusz nr 2 „Wybieranie ilości współczynników”	
Przypadek 2a:	„Użytkownik przełącza przyciski radiowe”
Warunek wstępny:	
Kroki testowe:	<ol style="list-style-type: none"> 1. Włączenie programu 2. Przełączanie przycisków radiowych od 2 do 5 przełączając je od lewej do Prawej, a następnie w drugą stronę
Oczekiwany rezultat:	Po naciśnięciu przycisku powinny pojawić się tabela z odpowiednią ilością Pól do wypełnienia

8. Lista narzędzi, które zostaną wykorzystane w celu przeprowadzenia testów

Do testów jednostkowych użyto języka programowania Python, a dokładniej modułu pytest.