

SZCZEGÓŁOWY OPIS PRZEDMIOTU ZAMÓWIENIA (parametry i wymagania minimalne)

Zgodnie z projektem umowy § 1 ust. 2 i 3 stanowiącym załącznik nr 3 do SIWZ w ramach każdego zadania Do obowiązków Wykonawcy, w ramach ceny podanej w ust. 1, należy również:

- a) dostawa, oprogramowania zakupionego w oficjalnym kanale sprzedaży co oznacza zapewnienie stosownego pakietu usług gwarancyjnych, wsparcia technicznego – serwisowego kierowanego do użytkowników z obszaru Rzeczypospolitej Polskiej,
- b) w przypadku dostarczenie oprogramowania zapisanego na nośnikach, każdy z takich nośników musi być fizycznie nowy, posiadać kod aktywacyjny wraz z instrukcją aktywacyjną (oryginalnie zapakowany, zabezpieczony taśmą, nieposiadający śladów otwierania i użytkowania).

Wartość oferty zawiera wszystkie niezbędne koszty i opłaty, podatki etc. związane z prawidłową realizacją przedmiotu zamówienia.

Przedmiotem zamówienia jest dostawa oprogramowania komputerowego lub aktualizacji dla Wydziału Matematyczno – Przyrodniczego Uniwersytetu Rzeszowskiego.

Zamówienie zostało podzielone na 2 zadania:

Zadanie nr 1: Dostawa oprogramowania komputerowego lub aktualizacji dla Wydziału Matematyczno– Przyrodniczego Uniwersytetu Rzeszowskiego o następujących parametrach minimalnych: 2 szt.

Dostawa aktualizacji wraz z kontraktem serwisowym na 3 lata dla dwóch posiadanych licencji oprogramowania Mathematica o następujących parametrach minimalne dla Wydziału Matematyczno-Przyrodniczego.

Ilość	Opis przedmiotu zamówienia i minimalne wymagania
2	<p>Aktualizacja posiadanego oprogramowania dla licencji numer: L4800-9251 L4800-9318</p> <p>wraz z kontraktem serwisowym na 3 lata, wnosząca następującą funkcjonalność:</p> <ul style="list-style-type: none"> - możliwość wprowadzania poleceń w języku angielskim bez konieczności znajomości funkcji i składni środowiska programu, - możliwość tworzenia i zapisu dokumentów elektronicznych zawierających elementy interaktywne pozwalające ingerować nie tylko w treść wizualizacji, ale też w realizowany w dokumencie proces obliczeniowy, - algorytmy do analizy i przetwarzania obrazów i wyszukiwania szczegółów w czasie rzeczywistym, - funkcje do ciągłej i dyskretnej analizy falkowej , - funkcje do obliczeń pochodnych, obliczania wartości obligacji, miar wrażliwości, obliczania wartości pieniądza w czasie przy użyciu różnych scenariuszy stóp procentowych, - możliwość wykonywania obliczeń wykorzystujących procesor karty graficznej GPU oraz technologię CUDA i środowisko OpenCL, - pasek podpowiedzi ułatwiający dobierane funkcji w kolejności od potencjalnie najbardziej pożądanych, - narzędzie ułatwiające zmianę jednostek, ich przeliczanie oraz przejścia między systemami jednostek, - stochastyczne równania różniczkowe i szeregi czasowe, - rozwiązywanie równań różniczkowych z nieciągłością, równań różniczkowych z parametrami, algebraicznych równań różniczkowych. <p>Licencja powinna zawierać 3 letni kontrakt serwisowy gwarantujący w okresie jego trwania m.in.: bezpłatną pomoc techniczną (przez e-mail, telefon lub fax) świadczoną przez producenta; automatyczny dostęp do wszelkich nowych wersji programu z bezpłatnymi uaktualnieniami.</p>
<p><i>Przykładowe oprogramowanie spełniające wymagania:</i> Upgrade z wersji 9.0.1 do wersji 10.4 dla licencji L4800-9251 Upgrade z wersji 10.0.2 do wersji 10.4 dla licencji L4800-9318</p>	

Zadanie nr 2: Dostawa oprogramowania komputerowego lub aktualizacji dla Wydziału Matematyczno-Przyrodniczego Uniwersytetu Rzeszowskiego o następujących parametrach minimalnych: 1 szt.

Dostawa aktualizacji licencji nr 894937 Oprogramowania Matlab (z wyłączeniem Bioinformatics Toolbox oraz Parallel Computing Toolbox) lub dostawa nowego oprogramowania o podanych parametrach minimalnych dla Wydziału Matematyczno-Przyrodniczego.

W przypadku dostawy nowego oprogramowania należy dostarczyć oprogramowanie o parametrach minimalnych:

Elementy składowe:

- a) Środowisko programistyczne przeznaczone do rozwijania algorytmów, wizualizacji i analizy danych oraz prowadzenia obliczeń numerycznych. Szybsze rozwiązywanie problemów, niż przy wykorzystaniu tradycyjnych języków programowania takich jak C, C++ czy Fortran. Może zostać wykorzystany w szerokim polu zastosowań. Przykładowe obszary, w których program jest często podstawowym narzędziem pracy to:
 - przetwarzanie sygnałów,
 - przetwarzanie obrazów,
 - telekomunikacja,
 - projektowanie układów sterowania,
 - matematyka finansowa.
- b) Platforma do wielodomenowej symulacji i projektowania systemów dynamicznych z wykorzystaniem modeli. Stanowi interaktywne środowisko graficzne i zbiór bibliotek konfigurowalnych z blokami pozwalającymi dokładnie projektować, symulować, implementować i kontrolować przetwarzanie sygnałów, telekomunikację i inne systemy zdefiniowane w dziedzinie czasu.

Główne cechy programu:

- Obszerne i dające się powiększać biblioteki zdefiniowanych bloków.
 - Interaktywny edytor graficzny pozwalający intuicyjnie asemblować i zarządzać diagramami bloków.
 - Zdolność zarządzania złożonymi projektami poprzez segmentację modelu i hierarchizację komponentów projektu.
 - Eksplorator modelu (Model Explorer) nawigujący, tworzący, konfigurujący i przeszukujący wszystkie sygnały, parametry i właściwości modelu.
 - Zdolność do komunikowania się z innymi programami symulacyjnymi i dołączania ręcznie pisanego kodu włącznie z algorytmami minimum MATLABa.
 - Możliwość uruchamiania stało- lub zmiennokrokowych symulacji w systemach w dziedzinie czasu interaktywnie lub za pomocą symulacji batch.
 - Funkcje interaktywnego definiowania wejść i wyjść podglądowych do oszacowywania zachowania się modelu.
 - Graficzny debugger do sprawdzania wyników symulacji i diagnozowania nieoczekiwanego zachowania się projektu.
 - Pełny dostęp do minimum MATLABa w celu analizy i wizualizacji danych, poszerzenia graficznego interfejsu i definiowania danych i parametrów modelu.
 - Narzędzia diagnozy i analizy modelu pozwalające upewnić się, co do spójności modelu i wykryć błędy modelowania.
- c) Oprogramowanie musi zapewnić minimum wyspecjalizowane narzędzia modelowania, projektowania i analizy systemów ze sprzężeniami, obejmując zarówno klasyczne jak i nowoczesne metody projektowe.
 - LTI Viewer - interaktywny graficzny interfejs użytkownika (GUI) do analizy i porównywania liniowych systemów niezmiennych w czasie (LTI systems).
 - Wykresy w dziedzinie czasu odpowiedzi na skok jednostkowy, impulsowej i zero-biegunowej oraz odpowiedzi w dziedzinie częstotliwości (Bode, Nyquist i Nichols) w jednym oknie.
 - Narzędzie projektowania systemów z pojedynczym wejściem i pojedynczym wyjściem (SISO Design Tool) - interaktywny GUI do analizy i konfiguracji systemów kontroli SISO używając wykresów linii pierwiastkowych, Bodego i Nicholasa.
 - Graficzny kompensator ograniczający w SISO Design Tool.
 - Interaktywny wykres odpowiedzi w zamkniętej pętli ciągle aktualizowany wraz z modyfikacją kompensatora w SISO Design Tool.
 - Opcje zapisu i odczytu umożliwiające archiwizację projektów z wielokrotnymi kompensatorami.
 - Pełna kontrola nad wizualizacją w dziedzinie czasu i częstotliwości.

- Specjalne struktury danych zwane obiektami LTI (LTI Objects) dla zwartej reprezentacji formatów danych funkcji przejścia, przestrzeni stanu, zer i biegunów i odpowiedzi częstotliwościowych.
 - Obsługa systemów wielowejściowych i wielowyjściowych (MIMO), systemów czasu ciągłego i próbkowanych danych oraz systemów z opóźnieniami czasowymi.
 - Obsługa różnorodnych metod przekształceń dyskretnych na ciągłe.
 - Narzędzia nowoczesnych technik kontroli projektowania (umieszczanie biegunów, regulacje LQR-LQG, projektowanie filtrów Kalmana, wyznaczanie obserwowalności i sterowalności i rozwiązania równań Riccatiego i Lapunowa).
- d) Rozszerzenie środowiska o narzędzia do projektowania systemów opartych na logice rozmytej. Graficzny interfejs użytkownika (GUI) prowadzi przez kolejne kroki systemu projektowania wnioskowania rozmytego. Moduł zapewnia funkcje dla wielu typowych metod logiki rozmytej.
- Wyspecjalizowane graficzne interfejsy użytkownika (GUI) do budowy rozmytych systemów wnioskujących oraz podglądu i analizy wyników.
 - Funkcje przynależności do tworzenia systemów rozmytego wnioskowania.
 - Obsługa logiki AND, OR i NOT w zdefiniowany przez użytkownika sposób.
 - Standardowy system wnioskowania rozmytego typu Mamdani i Sugeno.
 - Zautomatyzowane funkcje przynależności kształtujące poprzez techniki uczenia neuroadaptywne i rozmytej analizy skupień.
 - Zdolność umieszczania systemu rozmytego wnioskowania w modelu Simulinka.
 - Zdolność do generowania kodu C lub samodzielnie wykonywalnego enginu rozmytego wnioskowania.
- e) Oprogramowanie do generowania niezależnego kodu C/C++ z kodu języka minimum MATLAB. Wygenerowany kod źródłowy jest przenośny i czytelny. Wspiera podzbiór cech i właściwości języka MATLAB włączając w to:
- instrukcje sterujące przepływem programu
 - funkcje
 - operacje macierzowe
 - Potrafi także wygenerować gotową MEX-funkcję, co pozwala na znaczne przyspieszenie działania projektowanego algorytmu oraz łatwe przetestowanie wygenerowanego kodu źródłowego.
- f) Oprogramowanie rozszerzające minimum środowiska MATLABa o funkcje do projektowania, implementacji, wizualizacji i symulacji sieci neuronowych. Sieci neuronowe to potężne narzędzie stosowane wszędzie tam, gdzie formalna analiza byłaby trudna, lub wręcz niemożliwa, w tym do rozpoznawania wzorców i identyfikacji nieliniowej oraz sterowania.
- Graficzny interfejs użytkownika do tworzenia, uczenia i symulowania sieci neuronowych.
 - Wsparcie dla najpowszechniejszych architektur (nadzorowanych i bez nauczyciela).
 - Kompletny zestaw funkcji uczących się.
 - Zbiór bloków Simulinka wraz z dokumentacją i demonstracjami aplikacji systemów sterowania.
 - Automatyczne tworzenie modeli Simulinka z obiektów sieci neuronowych.
 - Modułowa reprezentacja sieci, pozwalająca na zadawanie dowolnej liczby warstw wejściowych i dowolnej liczby połączeń między warstwami.
 - Zwiększanie skuteczności uczenia się dzięki funkcjom przetwarzającym dane przed i po uczeniu sieci.
 - Procedury poprawiające generalizację.
 - Funkcje wizualizacyjne pokazujące skuteczność sieci.
- g) Oprogramowanie pozwalające uruchamiać modele minimum Simulinka i Stateflow w czasie rzeczywistym na komputerze PC typu desktop lub laptop. Pozwala na tworzenie i kontrolowanie uruchamiania w czasie rzeczywistym z poziomu Simulinka. Używając Simulink Coder można generować kod C, kompilować go i rozpocząć uruchamianie w czasie rzeczywistym w systemie Windows równocześnie podłączając się do kart wejścia-wyjścia komputera PC. Inne aplikacje systemu Windows nie przestają pracować w trakcie wykonywania zadania.
- Uruchamia modele w czasie rzeczywistym na komputerze PC.
 - Zapewnia szybkie przetwarzanie danych punkt po punkcie z minimalnym opóźnieniem, co jest wymagane w aplikacjach systemów kontrolnych.
 - Uzyskuje próbki z częstotliwością przekraczającą 10 KHz dla większości modeli.
 - Współpracuje z ponad 250 płytami wejścia-wyjścia dla wejść i wyjść czasu rzeczywistego, wliczając karty PCMCIA dla przenośnego szybkiego prototypowania na laptopach.
 - Umożliwia kontrolę uruchamiania modeli czasu rzeczywistego wprost z paska narzędzi Simulinka tworząc środowisko dla prototypu "PC-in-the-loop".
 - Obsługuje zewnętrzny tryb Simulinka umożliwiając użycie innych produktów i skopów (scope) Simulinka bez dodatkowych bloków interfejsowych.
 - Umożliwia rejestrację sygnałów i regulację parametrów.
 - Wykorzystuje kompilator C do budowy kodu czasu rzeczywistego.

- Obsługuje zbiory konfiguracyjne Simulinka dla wygodnego przełączania pomiędzy symulacją a uruchamianiem kodu w czasie rzeczywistym używając tego samego modelu.
- h)** Oprogramowanie generujące i uruchamiające samodzielny kod C do rozwijania i testowania algorytmów modelowanych w minimum Simulinku. Kod wynikowy może być użyty do wielu aplikacji (w tym aplikacji czasu rzeczywistego) takich jak przyspieszanie symulacji, szybkie prototypowanie i testowanie metodą hardware-in-the-loop. Pozwala interaktywnie dostrajać i monitorować generowany kod używając bloków Simulinka i wbudowanych możliwości analizy do uruchamiania i interakcji z kodem poza środowiskiem MATLABa i Simulinka.
- Tworzy kod ANSI/ISO C i programy wykonywalne dla modeli dyskretnych, ciągłych i hybrydowych.
- Używa bloków modelu do tworzenia przyrostowego i budowy kodu dla dużych aplikacji.
- Obsługuje cechy słownika danych Simulinka dla całkowitych, zmienna- i stałoprzecinkowych typów danych.
- Tworzy kod dla modeli o jednej i zmiennej częstotliwości próbkowania oraz asynchronicznych.
- Obsługuje jedno- i wielozadaniowe systemy operacyjne oraz środowiska bezoperacyjne (bez systemu operacyjnego).
- Wykonuje optymalizację kodu polepszającą szybkość uruchamiania programu.
- Zapewnia możliwości do konfigurowania i legalnego ingerowania w kod.
- Pozwala interaktywnie dopasowywać i monitorować utworzony kod wewnątrz lub na zewnątrz Simulinka.

Wyżej wymienione oprogramowanie musi być kompatybilne z :

- Symbolic Math Toolbox
- Global Optimization Toolbox
- Financial Toolbox
- Financial Instruments Toolbox
- Optimization Toolbox
- Statistics Toolbox
- Partial Differentia Equation Toolbox

W przypadku dostawy nowego oprogramowania o podanych powyżej parametrach minimalnych

Wykonawca zobowiązany do przeszkolenia pracowników w zakresie obsługi nowego oprogramowania oraz do zapewnienia pełnej zgodności i kompatybilności nowego oprogramowania z posiadanym przez Zamawiającego umożliwiając bezproblemową ciągłość pracy u Zamawiającego. Dostarczone nowe oprogramowanie musi umożliwiać pracę na dokumentach (min. otwieranie, edytowanie, kopiowanie, zapisywanie) zapisanych przy pomocy wcześniej posiadanego przez Zamawiającego oprogramowania w tym