

JĘZYKI PROGRAMOWANIA

zagadnienia na egzamin końcowy (ostatnia aktualizacja 23 września 2025)

Studia Stacjonarne i Niestacjonarne

CZĘŚĆ TEORETYCZNA

Generacje języków programowania: historia komputerów¹, generacje komputerów, prawo Moora, ewolucja języków programowania, generacje języków programowania, rankingi popularności, paradygmaty programowania, języki imperatywne i deklaratywne.

Wprowadzenie do programowania: język, język programowania, translator (definicja i podział), rodzaje translatorów (wymienić i opisać), kod źródłowy, kod maszynowy, plik nagłówkowy, biblioteka (definicja, podział i sposób użytkowania), etapy kompilacji (schemat i opis), środowisko programistyczne (definicja i najważniejsze elementy), klasyfikacja języków programowania, przykłady języków programowania.

Kompilatory Fortranu: Fortran – opis języka, standardy języka Fortran, podział kompilatorów Fortranu (podać przykłady), sposoby edycji kodu źródłowego, zmienne środowiskowe, polecenie kompilacji kodu źródłowego z zastosowaniem kompilatora gfortran (w systemach Windows i Linux), sposób uruchamiania skompilowanych programów (w systemach Windows i Linux).

Elementy języków programowania: dopuszczalne znaki i symbole, słowa kluczowe, struktura kodu, zasady edycji kodu (Fortran), identyfikatory, operacje wejścia-wyjścia, działania na plikach, typy zmiennych, formatowanie, zakresy obowiązywania zmiennych, komunikacja z systemem operacyjnym, statyczne i dynamiczne zarządzanie pamięcią, zmienna prosta, zmienna indeksowana, stała, podział zmiennych i stałych, wyjątki, kolejność działania operatorów w wyrażeniach złożonych, wyrażenia i operatory arytmetyczne, wyrażenia i operatory logiczne, wyrażenia tekstowe.

Instrukcje języków programowania: definicja oraz podział instrukcji, instrukcje czynne i bierne (definicja i przykłady), instrukcja przypisania, instrukcja skoku, instrukcja zatrzymania, instrukcja wstrzymania, instrukcja powrotu, instrukcja końca, instrukcje warunkowe (rodzaje, opis i przykłady), instrukcja wyboru (opis oraz zasady definiowania przypadków), instrukcje powtórzeń (podział, opis oraz przykłady), instrukcje wejścia-wyjścia (rodzaje, opis oraz przykłady), podstawowe rodzaje segmentów w kodach źródłowych, procedura i funkcja (definicja, cechy specyficzne, sposób użytkowania).

Algorytmy: definicja i cechy algorytmu, iteracja, rekurencja, metoda numeryczna, sposoby przedstawiania algorytmów, diagram (definicja oraz stosowane symbole), etapy tworzenia algorytmów, rodzaje błędów w kodach źródłowych, podstawowy test algorytmu, kryteria optymalizacji algorytmów.

Dobre praktyki: co to są dobre praktyki w programowaniu, podać zbiór dobrych praktyk w odniesieniu do języka Fortran.

CZĘŚĆ PRAKTYCZNA

Elementy programowania imperatywnego:

- struktura kodu źródłowego (PROGRAM, END), podstawowe operacje wejścia-wyjścia (PRINT, WRITE, READ), instrukcja przypisania (=), zmienne proste i indeksowane (INTEGER, REAL, COMPLEX, LOGICAL, CHARACTER, TRUE, FALSE, LEN), zmienne statyczne i dynamiczne (ALLOCATABLE, ALLOCATE, DEALLOCATE), stałe (PARAMETER), jawna i niejawna deklaracja typu (IMPLICIT NONE), formatowanie danych (FORMAT);
- wyrażenia i operatory arytmetyczne (+, -, *, /, **), wyrażenia i operatory logiczne (.NOT., .AND., .OR., .EQV., .NEQV.), operatory relacji (.GT., .GE., .LT., .LE., .EQ., .NE.), wyrażenia i operatory tekstowe (TRIM, //, :,), instrukcje sterujące (GOTO, IF, THEN, ELSE, SELECT CASE, DO, DO WHILE, CONTINUE);
- funkcje lokalne, wewnętrzne i zewnętrzne (FUNCTION), procedury (SUBROUTINE), pliki dołączane (INCLUDE), struktura programów.

Elementy wizualizacji danych:

- podstawy wizualizacji 2D w trybie skryptowym w środowisku graficznym Gnuplot;
- podstawy wizualizacji 3D z wykorzystaniem plików VTK i programu ParaView.

¹ Na szaro zaznaczone są zagadnienia, które były omawiane na wykładach lub ćwiczeniach, ale nie obowiązują na egzaminie.