

Nazwa w jęz. angielskim: IT and Introduction to Programming
Nazwa w języku polskim: Informatyka i podstawy programowania

Dane dotyczące zajęć:
Information on course:

Jednostka oferująca: Wydział Budownictwa // dr hab. inż. Grzegorz Wandzik, prof. PŚ
Course offered by: Faculty of Civil Engineering // dr hab. inż. Grzegorz Wandzik, prof. PŚ

Język wykładowy:
angielski
Language:
English
Strona WWW: Course homepage:
Skrócony opis:
Przedmiot obejmuje zagadnienia związane z podstawami informatyki i programowania komputerów. Kurs składa się z 15 godzin wykładów i 45 godzin zajęć laboratoryjnych, podczas których studenci uczą się obsługi SMath Studio i rozwiązywania podstawowych problemów z wykorzystaniem systemów CAS. Podczas laboratoriów studenci uczą się pisanie prostych aplikacji w języku programowania Pascal/C#.
Short description:
The subject covers issues related to the basics of computer science and computer programming. The course consists of 15 hours of lectures and 45 hours of laboratory during which students learn how to use SMath Studio and solve basic problems using CAS systems. During laboratories students learn also how to write simple applications in Pascal programming language.
Opis:
Treści programowe
Wykład
1. Dane w informatyce i zarządzanie danymi: Dane, pliki tekstowe i binarne, kodowanie tekstu, unicode (systemy UTF), formaty tekstowe: txt, csv, rtf, html, xml, json; Formaty graficzne – bitmapa, kompresja, grafika wektorowa, formaty stosowane do graficznej wymiany danych.
2. Programy matematyczne dla inżynierów (środowisko MathCad/SMath Studio): zmienne zakresowe i tablice, operacje na macierzach, jednostkach, funkcjach, rachunek różniczkowy i pochodne w zagadnieniach mechanicznych – obliczenia symboliczne i numeryczne, wykresy.
3. Programowanie komputerów: interpretery i kompilatory, języki (podobieństwa i różnice), składnia, Pascal, .NET Framework, Python, VBA.
4. Programowanie IDE: systemy RAD, budowanie interfejsu aplikacji, aplikacje sterowane zdarzeniami, dodawanie komponentów na formularzu, typy komponentów, inspektor obiektów, kod debugujący, punkty przerwania, stos.
5. Typy danych w programowaniu: typy danych predefiniowane i definiowane przez użytkownika, konwencje nazewnictwa, reprezentacja liczbowa, zmienne lokalne i globalne, stałe, tablice, listy, rekordy, funkcje (metody), klasy i obiekty, zdarzenia; argumenty funkcji – przekazują dane według wartości lub referencji; typy wyliczone, otwarte tablice, strumienie, pliki.
6. Organizacja programu w języku Pascal: składnia, instrukcja blokowa, instrukcje warunkowe w języku Pascal (if, case), pętle (for, while, powtórz), kompilacja kodu, tryb chroniony i wyjątki (try – z wyjątkiem), usuwanie błędów – debugowanie kodu, namierzać.
7. Algorytmy budowy i organizacja kodu: podział zadania na proste problemy, jednostki i odniesienie do jednostek, interfejs i implementacja, przeciążanie metod.
Laboratorium

1. Rozwiązywanie problemów matematycznych i/lub inżynierskich z wykorzystaniem systemu CAS (SMath Studio),
2. Opracowanie algorytmu i kodu programu dla indywidualnego problemu (z zakresu mechaniki, fizyki, geometrii) w języku Pascal/C#.

Wykład: stacjonarne: 15 h

Laboratorium: stacjonarne: 45 h

Liczba punktów ECTS: 5

Description:

Lecture

1. Data in IT and data management: Data, text and binary files, text coding, unicode (UTF systems), text formats: txt, csv, rtf, html, xml, json; Graphic formats – bitmap, compression, vector graphics, formats used for graphic data exchange.
2. Math programs for engineers (MathCad/SMath Studio environment): range variables and arrays, operations on matrices, units, functions, calculus and derivatives in mechanical problems – symbolic and numerical calculations, graphs.
3. Computer programming: interpreters and compilers, languages (similarities and differences), syntax, Pascal, .NET Framework, Python, VBA.
4. Programming IDE: RAD systems, building application interface, events driven applications, adding components on form, types of components, object inspector, debug code, breakpoints, stack.
5. Data types in programming: predefined and user-defined data types, naming conventions, numbers representation, local and global variables, constants, arrays, lists, records, functions (methods), classes and objects, events; functions arguments – transfer data by value or reference; enumerated types, open arrays, streams, files.
6. Program organization in Pascal: syntax, block statement, conditional statements in Pascal (if, case), loops (for, while, repeat), code compilation, protected mode and exceptions (try – except), removing errors – debugging code, trace.
7. Building algorithms and organization of the code: splitting task into simple problems, units and reference to units, interface and implementation, overloading.

Laboratory

1. Solving mathematical/engineering problems with CAS system (SMath Studio),
2. Elaboration of algorithm and program code for individual problem (related to mechanics, physics, geometry) in Pascal/C# language.

Contact hours:

Lecture: full-time studies: 15 h

Laboratory: full-time studies: 45 h

Number of ECTS credits: 5

Literatura:

- [1] Borland Delphi 4 in 21 Days, Macmillan Computer Publishing, 2008
- [2] M. Cantu: Mastering Delphi 6, Sybex, 2001
- [3] G. Wandzik: IT and Introduction to programming, teaching materials, 2020.

Strony i kursy internetowe:

[4] web page: <https://www.embarcadero.com/resources>.

[5] web page : <https://en.smath.com/view/SMathStudio/summary>.

Bibliography:

LITERATURE

- [1] Borland Delphi 4 in 21 Days, Macmillan Computer Publishing, 2008

- [2] M. Cantu: Mastering Delphi 6, Sybex, 2001
 [3] G. Wandzik: IT and Introduction to programming, teaching materials, 2020.

ADDITIONAL RESOURCES:

- [4] web page: <https://www.embarcadero.com/resources>.
 [5] web page : <https://en.smath.com/view/SMathStudio/summary>.

Efekty uczenia się:

WIEDZA:

- (1) Zrozumienie typów danych i ich reprezentacji.
- (2) Zrozumienie idei programowania obiektowego, właściwości, metod, zdarzeń.
- (3) Zrozumienie idei tworzenia algorytmów.
- (4) Zasady posługiwania i dostępne metody w systemach CAS (operacje macierzowe, obliczenia symboliczne, numeryczne rozwiązywanie równań itp.).

UMIEJĘTNOŚCI:

- (4) Potrafi zestawiać obciążenia i tworzyć kombinacje oddziaływań stosowane w kontroli SGN i SGU - [efekt kierunkowy K1A_U02].
- (5) Potrafi określać nośność na działanie sił wewnętrznych (N, M, V) dla prostych elementów żelbetowych, takich jak: płyty, belki, słupy, ściany - [efekt kierunkowy K1A_U04].

Learning outcomes:

KNOWLEDGE:

- (1) Understanding data types and its representation.
- (2) Understanding the idea of the object programming, properties, methods, events.
- (3) Understanding the idea of algorithms creations.
- (4) Concepts and capabilities of CAS systems (matrix operations, symbolic calculations, solving equations using numerical methods, etc.).

SKILLS

- (1) Ability to formulate and solve problems and selection of appropriate tool.
- (2) Ability to make critical analysis of mathematical problems.
- (3) Ability of building simple algorithms dividing problems into simple tasks.

Metody i kryteria oceniania:

WARUNKI ZALICZENIE PRZEDMIOTU:

- (1) Aktywna obecność na zajęciach laboratoryjnych.
- (2) Opracowanie dokumentu w systemie CAS (SMath Studio).
- (3) Rozwiązanie problemów i opracowanie prostego programu w języku programowania Pascal lub C#.
- (4) Zaliczenie kolokwium obejmującego tematykę omawianą na wykładach i laboratoriach.

OCENA KOŃCOWA:

40% (kolokwium), 60% (laboratorium).

Wykład: Zaliczenie w formie pisemnej. Kryterium zaliczenia: minimum 50% maksymalnej liczby punktów.

Laboratorium: Samodzielne opracowanie prostego programu (opracowanie i implementacja algorytmu) i dokumentu w systemie wspomagającym projektowanie (SMath Studio).

Assessment methods and assessment criteria:

REQUIREMENTS:

- (1) Active attendance in laboratories.
- (2) Elaboration of document in CAS system (SMath studio).
- (3) Solution of problems and elaboration of the simple program in Pascal or C# programming language.
- (4) Passing the colloquium covering topics discussed during lectures and laboratories.

FINAL ASSESSMENT:

40% (colloquium), 60% (laboratories).

Lecture: Passing the course in the form of the written test. Criterion for passing the course: 50% of maximal number of points.

Laboratory: Individual elaboration of the simple program (creating algorithm) i dokumentu w systemie CAS (SMath Studio).

Element of course groups in various terms:

Opis grupy przedmiotów Course group description	Cykl pocz. First term	Cykl kon. Last term
przedmiot obowiązkowy studia stacjonarne i niestacjonarne stopień studiów – I stopień kierunek studiów – budownictwo semestr dowolny	2024/2025	