



## KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Języki obiektowe - podstawy programowania [N1MiBM2>JOPP]

### Przedmiot

Kierunek studiów

Mechanika i budowa maszyn

Rok/Semestr

1/1

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

polski

Forma studiów

niestacjonarne

Wymagalność

obieralny

### Liczba godzin

Wykład

8

Laboratorium

16

Inne (np. online)

0

Ćwiczenia

0

Projekty/seminaria

0

### Liczba punktów ECTS

4,00

### Koordynatorzy

### Wykładowcy

### Wymagania wstępne

Umiejętność obsługi komputera.

### Cel przedmiotu

Celem przedmiotu jest zapoznanie studentów z podstawami programowania w językach wysokiego poziomu (przy użyciu wybranego języka). W trakcie zajęć przedstawiane są podstawowe pojęcia i konstrukcje występujące w programowaniu (zmienne, typy danych, instrukcje, obsługa komunikacji z użytkownikiem, podprogramy) oraz budowa programów wykorzystujących powyższe konstrukcje (połączona z analizą problemów w kontekście algorytmicznym) i kontrola ich poprawności.

### Przedmiotowe efekty uczenia się

Wiedza:

Zna typologię języków programowania, ma wiedzę o ich podstawowych zastosowaniach. Zna narzędzia wspomagające pracę programisty. Zna źródła informacji obejmujących problemy tworzenia, programowania i użytkownika aplikacji. Potrafi formułować algorytmy i je programować z użyciem przynajmniej jednego z popularnych narzędzi.

Umiejętności:

Potrafi tworzyć proste aplikacje. Potrafi zaprojektować dobry interfejs użytkownika dla aplikacji. Potrafi

ocenić na podstawowym poziomie, przydatność rutynowych metod i narzędzi informatycznych oraz wybrać i zastosować właściwą metodę i narzędzia do typowych zadań informatycznych

Kompetencje społeczne:

Rozumie potrzebę uczenia się przez całe życie; potrafi inspirować i organizować proces uczenia się innych osób. Ma świadomość ważności i rozumienia pozatechnicznych aspektów i skutków działalności inżynierskiej.

### Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wykład: Zaliczenie na podstawie pisemnego kolokwium składającego się z 30 pytań testowych w skali 0/1. Zaliczenie w przypadku uzyskania minimum 51 %.

Laboratorium: Zaliczenie odbywa się na podstawie kolokwium praktycznego na laboratoriach za 60 punktów oraz zadań realizowanych na laboratorium - 40 punktów. Ocena końcowa jest wyznaczana zgodnie z następującą punktacją: 100 punktów zgodnie z przyjętym systemem oceniania - 3.0 od 41 punktów, 3.5 od 56 punktów, 4.0 od 71 punktów, 4.5 od 81, 5.0 od 91 punktów

### Treści programowe

Wykład:

Na wykładzie "Języki skryptowe" przedstawiony zostanie jeden wybrany przez prowadzącego język programowania (C albo python). Wykład zorganizowany jest w taki sposób aby był przystępny dla osób o bardzo małych umiejętnościach programistycznych (szczególnie w początkowej fazie wykładu), jak również przydatny dla osób zaawansowanych. W pierwszej części zaprezentowana zostanie semantyka i syntaktyka języka, podstawowe instrukcje i operatory, szczegółowo zostaną omówione zasady działania pętli. W dalszej części omówione zostaną sposoby budowania prostych i zaawansowanych struktur danych wraz z przykładami.

Laboratorium - praktyczne zastosowanie wiedzy poznanej na wykładzie, implementacja i testowanie - omówienie tematów:

- 1) Model kompilacji.
- 2) Operatory i ich pierwszeństwo.
- 3) Instrukcje preprocesora, makra
- 4) Tablice.
- 5) Wskaźniki i ich powiązanie z tablicami.
- 6) Przekazywanie wartości do funkcji.
- 7) Wskaźniki na funkcje.
- 8) Struktury.
- 9) Tworzenie dynamiczne struktury danych za pomocą wskaźników
- 10) Operacje na plikach.
- 11) Omówienie standardowej biblioteki języka C (funkcje związane z podstawowymi operacjami na łańcuchach, funkcje matematyczne, konwersje)
- 12) Omówienie funkcji bezpiecznych i niebezpiecznych (problemy z przepełnieniem bufora)
- 13) Pisanie dużych programów i zastosowanie gnu make, autoconf i configure.
- 14) Strategie poszukiwania błędów i zastosowanie narzędzi typu debugger, valgrind.

### Tematyka zajęć

brak

### Metody dydaktyczne

Wykład: prezentacja multimedialna - prezentacja ilustrowana przykładami oraz filmami, analiza problemów wraz z ich rozrysowaniem na tablicy, dyskusja i analiza problemów.

Laboratorium,: Zrealizowanie zostanie poprzez pracę przy komputerach, w trakcie której wraz z prowadzącym napisane zostaną programy ilustrujące działanie omawianego zagadnienia. Na późniejszych etapach zajęć laboratoryjnych studenci indywidualnie pisać będą programy realizujące dane zadanie, które wcześniej będzie omówione i przykładowo zaprezentowane przez prowadzącego.

### Literatura

Podstawowa:

1. C. Albing, J.P. Vossen, C. Newham : Bash. Receptury. Helion, Gliwice.
2. H. Schwichtenberg : Windows PowerShell. Podstawy. Helion, Gliwice.
3. P. Norton, A. Samuel : Python. Od podstaw. Helion, Gliwice.
4. M. Lutz : Python. Wprowadzenie. Helion, Gliwice.

Uzupełniająca:

Źródła internetowe: np. <https://docs.python.org/3/library/index.html>

### Bilans nakładu pracy przeciętnego studenta

|  | Godzin | ECTS |
|--|--------|------|
| Łączny nakład pracy  | 100    | 4,00 |
| Zajęcia wymagające bezpośredniego kontaktu z nauczycielem  | 24     | 1,00 |
| Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwίων/egzaminu, wykonanie projektu) | 76     | 3,00 |