

## INFORMATYKA 2 - Pracownia specjalistyczna 30 h

Kod przedmiotu: EZ1E3012, ECTS: 4 pkt.

Kierunek: Elektrotechnika, studia niestacjonarne I stopnia

Semestr: III, rok akademicki: 2020/2021

Grupa: PS1, sobota, godz. 11:00-13:30, Teams

dr inż. Jarosław Forenc

WE-204, tel. (0-85) 746-93-97

e-mail: [j.forenc@pb.edu.pl](mailto:j.forenc@pb.edu.pl)

WWW: <http://jforenc.prv.pl/> (Dydaktyka)

**Konsultacje:**  
wtorek, 16:00-17:00, WE-204 / Teams  
piątek, 09:00-10:00, WE-204 / Teams  
piątek, 14:00-15:00, WE-204 / Teams  
sobota, 16:00 -17:00, Teams (zaoczne)  
niedziela, 14:00 - 15:30, Teams (zaoczne)

### Program szczegółowy:

1. Zajęcia organizacyjne. Tablice dwuwymiarowe, operacje na tablicach dwuwymiarowych.
2. Łańcuchy znaków. Plik nagłówkowy string.h.
3. Struktury, odwołania do pól struktury. Inicjalizacja zmiennej strukturalnej.
4. Wskaźniki. Dynamiczny przydział pamięci w języku C.
5. Funkcje, ogólna struktura funkcji. Umieszczanie definicji funkcji w programie. Przekazywanie argumentów do funkcji przez wartość i wskaźnik. Rekurencyjne wywołanie funkcji.
6. Kolokwium nr 1. Programy wielomodułowe.
7. Zaawansowane operacje wejścia-wyjścia w języku C. Pliki tekstowe w języku C.
8. Pliki binarne w języku C.
9. Operacje na plikach tekstowych i binarnych. Kolokwium nr 2.
10. Operatory bitowe. Zaliczenie zajęć.

### Literatura:

1. Prata S.: *Język C. Szkoła programowania. Wydanie VI.* Helion, Gliwice, 2016.
2. Kernighan B.W., Ritchie D.M.: *Język ANSI C. Programowanie. Wydanie II.* Helion, Gliwice, 2010.
3. Deitel P.J., Deitel H.: *Język C. Solidna wiedza w praktyce. Wydanie VIII.* Helion, Gliwice, 2020.
4. King K.N.: *Język C. Nowoczesne programowanie. Wydanie II.* Helion, Gliwice, 2011.
5. Kochan S.G.: *Język C. Kompendium wiedzy. Wydanie IV.* Helion, Gliwice, 2015.
6. Reese R.: *Wskaźniki w języku C. Przewodnik.* Helion, Gliwice, 2014.
7. Instrukcje do pracowni specjalistycznej.
8. Materiały na stronie internetowej: <http://jforenc.prv.pl/dydaktyka.html>

## System oceniania obowiązujący w okresie normalnego funkcjonowania Uczelni

1. Obecność na zajęciach:
  - więcej niż dwie nieusprawiedliwione nieobecności skutkują niezaliczeniem pracowni.
2. Realizacja w trakcie zajęć zadań przedstawionych przez prowadzącego.
3. Zaliczenie dwóch kolokwii - praktycznych sprawdzianów pisania programów komputerowych:
  - kolokwia odbędą się na zajęciach nr 6 (ocena efektów EU1 i EU2) oraz na zajęciach nr 9 (ocena efektów EU3 i EU4) na terenie Uczelni,
  - poprawy kolokwii odbędą się poza zajęciami,
  - na kolokwiach można korzystać z materiałów w formie drukowanej (instrukcje do przedmiotu, książki, notatki, itp.),
  - za każde kolokwium można otrzymać od 0 do 100 pkt.,
  - na podstawie otrzymanych punktów wystawiana jest ocena:

<u>punkty</u>	<u>ocena</u>	<u>punkty</u>	<u>ocena</u>
91 - 100	5,0	61 - 70	3,5
81 - 90	4,5	51 - 60	3,0
71 - 80	4,0	0 - 50	2,0

- oba kolokwia muszą być zaliczone na ocenę pozytywną (min. 51 punktów).
4. Zaliczenie prac domowych:
    - prace domowe polegają na napisaniu programów komputerowych wskazanych przez prowadzącego zajęcia,
    - pracę domową należy wysłać na adres e-mailowy prowadzącego zajęcia do końca dnia (godz. 23:59), w którym odbywają się kolejne zajęcia,
    - błędy w programach mogą być poprawione do końca dnia, w którym będą odbywały się następne zajęcia,
    - za każdy poprawnie działający program student otrzymuje 1 pkt.,
    - w przypadku stwierdzenia niesamodzielności pracy domowej: nie jest ona zaliczana, nie można jej ponownie oddać, student otrzymuje -1 pkt.,
    - należy wysyłać tylko pliki z kodem źródłowym (.cpp, .c),
    - przed wysłaniem programu należy sprawdzić czy program:
      - kompiluje się oraz kompilator nie wyświetla ostrzeżeń,
      - jest poprawnym rozwiązaniem zadania,
      - ma odpowiednio sformatowany kod źródłowy (formatowanie kodu: Ctrl + K + F),
      - nosi nazwę zgodną z poniższym wzorcem:  
Nazwisko\_Imie\_NrInstrukcji\_NrZadania.cpp np. Kowalski\_Jan\_INF21Z\_3.cpp  
Nowak\_Anna\_INF25Z\_5.cpp
      - zawiera na początku komentarz nagłówkowy:

```
/*  
Nazwa: Kowalski_Jan_INF21Z_3.cpp  
Autor: Jan Kowalski  
Album: 123456  
Data: 01-10-2020  
Kod: EZ1E3012
```

Forma: Pracownia specjalistyczna  
 IDE: Microsoft Visual Studio 2008  
 Opis: Program wyświetlający tekst "Witaj świecie"

\*/

- co najmniej 70% prac domowych musi być zaliczonych.

5. Ocena końcowa wyznaczana jest na podstawie sumy otrzymanych punktów za kolokwia oraz punktów za prace domowe:

Punkty	ocena	punkty	ocena
182 – 200	5,0	122 - 141	3,5
162 – 181	4,5	102 - 121	3,0
142 – 161	4,0	0 - 101	2,0

**System oceniania obowiązujący w okresie czasowego ograniczenia lub zawieszenia funkcjonowania Uczelni**

1. Obecność na zajęciach prowadzonych zdalnie.

2. Realizacja w trakcie zajęć zadań przedstawionych przez prowadzącego.

3. Zaliczenie dwóch kolokwium - praktycznych sprawdzianów pisania programów komputerowych:

- kolokwia odbędą się na zajęciach nr 6 (ocena efektów EU1 i EU2) oraz na zajęciach nr 9 (ocena efektów EU3 i EU4) z wykorzystaniem technik zdalnego nauczania,
- poprawy kolokwium odbędą się poza zajęciami,
- na kolokwium można korzystać z własnych materiałów (instrukcje do przedmiotu, książki, notatki, itp.),
- za każde kolokwium można otrzymać od 0 do 100 pkt.,
- na podstawie otrzymanych punktów wystawiana jest ocena:

punkty	ocena	punkty	ocena
91 - 100	5,0	61 - 70	3,5
81 - 90	4,5	51 - 60	3,0
71 - 80	4,0	0 - 50	2,0

- oba kolokwia muszą być zaliczone na ocenę pozytywną (min. 51 punktów).

4. Zaliczenie prac domowych:

- prace domowe polegają na napisaniu programów komputerowych wskazanych przez prowadzącego zajęcia,
- pracę domową należy wysłać na adres e-mailowy prowadzącego zajęcia do końca dnia (godz. 23:59), w którym odbywają się kolejne zajęcia,
- błędy w programach mogą być poprawione do końca dnia, w którym będą odbywały się następne zajęcia,
- za każdy poprawnie działający program student otrzymuje 1 pkt.,
- w przypadku stwierdzenia niesamodzielności pracy domowej: nie jest ona zaliczana, nie można jej ponownie oddać, student otrzymuje -1 pkt.,
- należy wysyłać tylko pliki z kodem źródłowym (.cpp, .c),

- przed wysłaniem programu należy sprawdzić czy program:

- kompiluje się oraz kompilator nie wyświetla ostrzeżeń,
- jest poprawnym rozwiązaniem zadania,
- ma odpowiednio sformatowany kod źródłowy (formatowanie kodu: Ctrl + K + F),
- nosi nazwę zgodną z poniższym wzorcem:  
 Nazwisko\_Imie\_NrInstrukcji\_NrZadania.cpp np. Kowalski\_Jan\_INF21Z\_3.cpp  
 Nowak\_Anna\_INF25Z\_5.cpp
- zawiera na początku komentarz nagłówkowy:

```

/*
Nazwa: Kowalski_Jan_INF21Z_3.cpp
Autor: Jan Kowalski
Album: 123456
Data: 01-10-2020
Kod: E21E3012
Forma: Pracownia specjalistyczna
IDE: Microsoft Visual Studio 2008
Opis: Program wyświetlający tekst "Witaj świecie"
*/

```

- co najmniej 70% prac domowych musi być zaliczonych.

5. Ocena końcowa wyznaczana jest na podstawie sumy otrzymanych punktów za kolokwia oraz punktów za prace domowe:

punkty	ocena	punkty	ocena
182 - 200	5,0	122 - 141	3,5
162 - 181	4,5	102 - 121	3,0
142 - 161	4,0	0 - 101	2,0

Podstawę do zaliczenia przedmiotu (uzyskanie punktów ECTS) stanowi stwierdzenie, że każdy z założonych **efektów uczenia się** został osiągnięty w co najmniej minimalnym akceptowalnym stopniu.

**Efekty uczenia się i system ich oceniania:**

Student, który zaliczył przedmiot:

EU1	wykonuje podstawowe operacje na tablicach dwuwymiarowych w programach w języku C
Student, który zalicza na ocenę <b>dobry (3,0)</b> powinien: <ul style="list-style-type: none"> <li>- umieć definiować tablice dwuwymiarowe i wykonywać podstawowe operacje tablicowe;</li> <li>- znać sposób przechowywania i inicjalizacji łańcuchów znaków (napisów), umieć wykonywać operacje na łańcuchach znaków przy wykorzystaniu funkcji z pliku nagłówkowego string.h;</li> <li>- znać i stosować proste struktury.</li> </ul> Student, który zalicza na ocenę <b>dobry (4,0)</b> powinien (oprócz wymagań na ocenę 3): <ul style="list-style-type: none"> <li>- umieć wykonywać zaawansowane operacje na tablicach dwuwymiarowych;</li> <li>- umieć stosować złożone struktury;</li> <li>- umieć przydzielić dynamicznie i zwolnić pamięć na tablicę jednowymiarową (wektor) oraz wykonywać operacje na tej tablicy.</li> </ul>	

Student, który zalicza na ocenę **bardzo dobry (5,0)** powinien (oprócz wymagań na ocenę 4):

- umieć przydzielić dynamicznie i zwolnić pamięć na tablicę dwuwymiarową (macierz) oraz wykonywać operacje na tej tablicy.

**EU2** definiuje i wykorzystuje własne funkcje w programach w języku C

Student, który zalicza na ocenę **dostateczny (3,0)** powinien:

- znać ogólną strukturę funkcji w języku C, definiować własne funkcje zgodnie z tą strukturą, rozumieć przebieg wykonania programu zawierającego wywołania takich funkcji;
- umieć definiować funkcje bezargumentowe lub z argumentami, zwracające lub niezwracające wartości oraz odpowiednio je wywoływać.

Student, który zalicza na ocenę **dobry (4,0)** powinien (oprócz wymagań na ocenę 3):

- rozróżniać pojęcia deklaracji i definicji funkcji, stosować we właściwy sposób prototypy funkcji.

Student, który zalicza na ocenę **bardzo dobry (5,0)** powinien (oprócz wymagań na ocenę 4):

- znać różnice w przekazywaniu argumentów do funkcji przez wartość i wskaźnik oraz odpowiednio stosować obie metody w programach;
- umieć definiować własne funkcje rekurencyjne i stosować je w programach.

**EU3** tworzy programy wielomodułowe w języku C

Student, który zalicza efekt kształcenia powinien:

- umieć napisać i skompilować program w języku C składający się z kilku modułów (plików z kodem źródłowym);
- umieć odwoływać się do zmiennych i funkcji zdefiniowanych w innych modułach.

**EU4** stosuje operacje zapisu i odczytu plików w samodzielnie napisanych programach komputerowych

Student, który zalicza na ocenę **dostateczny (3,0)** powinien:

- znać schemat przetwarzania pliku i potrafić zastosować go w samodzielnie napisanych programach;
- umieć otworzyć plik w trybie tekstowym lub binarnym do zapisu lub odczytu oraz zamknąć ten plik;
- umieć pisać proste programy przetwarzające pliki tekstowe i binarne zawierające liczby całkowite i/lub rzeczywiste;
- znać i stosować w programach podstawowe funkcje do odczytu i zapisu plików tekstowych (fscanf, fprintf) oraz binarnych (fread, fwrite).

Student, który zalicza na ocenę **dobry (4,0)** powinien (oprócz wymagań na ocenę 3):

- umieć pisać programy przetwarzające pliki tekstowe i binarne zawierające dane różnych typów prostych i złożonych;
- znać i stosować w programach funkcje znakowe i łańcuchowe do odczytu i zapisu plików tekstowych oraz funkcje rekordowe (blokowe) do operacji na plikach binarnych.

Student, który zalicza na ocenę **bardzo dobry (5,0)** powinien (oprócz wymagań na ocenę 4):

- umieć pisać programy przetwarzające jednocześnie wiele plików tekstowych i binarnych.