

Informatyka 2 (ES1E3017)

Politechnika Białostocka - Wydział Elektryczny
Elektrotechnika, semestr III, studia stacjonarne I stopnia
Rok akademicki 2021/2022

Pracownia nr 1 (07.10.2021)

dr inż. Jarosław Forenc

Dane podstawowe

- dr inż. Jarosław Forenc
- Politechnika Białostocka, Wydział Elektryczny,
Katedra Elektrotechniki, Energoelektroniki i Elektroenergetyki
ul. Wiejska 45D, 15-351 Białystok
WE-204
- e-mail: j.forenc@pb.edu.pl
- tel. (0-85) 746-93-97
- <http://jforenc.prv.pl>
 - Dydaktyka - dodatkowe materiały do pracowni
- Konsultacje
 - poniedziałek, 16:00-17:30, WE-204 / Teams
 - wtorek, 14:00-15:30, WE-204 / Teams
 - sobota, 10:00-11:00, 13:45-15:00, WE-204 / Teams (zaoczne)

Program przedmiotu

1. Zajęcia organizacyjne. Tablice dwu- i wielowymiarowe, operacje na tablicach - cz. 1.
2. Tablice dwu- i wielowymiarowe, operacje na tablicach - cz. 2.
3. Łańcuchy znaków. Plik nagłówkowy string.h.
4. Struktury, odwołania do pól struktury. Inicjalizacja zmiennej strukturalnej.
5. Wskaźniki. Dynamiczny przydział pamięci w języku C.
6. Funkcje, ogólna struktura funkcji. Umieszczanie definicji funkcji w programie.
7. Funkcje, przekazywanie argumentów do funkcji przez wartość i wskaźnik. Rekurencyjne wywołanie funkcji.
8. Kolokwium nr 1.

Program przedmiotu

9. Programy wielomodułowe.
10. Zaawansowane operacje wejścia-wyjścia w języku C.
11. Pliki tekstowe w języku C.
12. Pliki binarne w języku C.
13. Operacje na plikach tekstowych i binarnych.
14. Kolokwium nr 2.
15. Operatory bitowe.

Literatura

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

Warunki zaliczenia przedmiotu

- Zaliczenie dwóch kolokwiów - praktycznych sprawdzianów pisania programów komputerowych (c.d.):
 - poprawy kolokwiów odbędą się poza zajęciami
 - oba kolokwia muszą być zaliczone na ocenę pozytywną (min. 51 pkt.)
 - 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:

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

Warunki zaliczenia przedmiotu

- Obecność na zajęciach:
 - więcej niż trzy nieusprawiedliwione nieobecności skutkują niezaliczeniem pracowni
- Realizacja w trakcie zajęć zadań przedstawionych przez prowadzącego
- Zaliczenie dwóch kolokwiów - praktycznych sprawdzianów pisania programów komputerowych
 - kolokwia odbędą się na terenie Uczelni na zajęciach nr 8 i 14

	PS1 / PS2
Kolokwium nr 1	19.11.2021
Kolokwium nr 2	21.01.2022

Warunki zaliczenia przedmiotu

- Zaliczenie prac domowych:
 - **prace domowe** polegają na napisaniu programów komputerowych wskazanych przez prowadzącego zajęcia
 - pracę domową należy wysłać na adres e-mailowy: j.forenc@pb.edu.pl 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 wysłać tylko pliki z kodem źródłowym (.c)
 - przed wysłaniem programu należy sprawdzić czy program:
 - kompiluje się oraz kompilator nie wyświetla ostrzeżeń
 - jest poprawnym rozwiązaniem zadania

Warunki zaliczenia przedmiotu

- przed wysłaniem programu należy sprawdzić czy program:
 - nosi nazwę zgodną z poniższym wzorcem:
Nazwisko_Imie_NrGrupy_NrInstrukcji_NrZadania.cpp
np. **Kowalski_Jan_PS1_INF21_3.cpp**
 - zawiera na początku komentarz nagłówkowy:

```
/*  
Nazwa: Kowalski_Jan_PS1_INF21_3.cpp  
Autor: Jan Kowalski, gr. PS1  
Album: 123456  
Data: 01-10-2021  
Kod: ES1E3017  
Forma: Pracownia specjalistyczna  
IDE: Code::Blocks 20.03  
Opis: Program wyświetlający tekst "Witaj swiecie"  
*/
```

Warunki zaliczenia przedmiotu

- Zaliczenie prac domowych (c.d.):
 - zaliczenie prac domowych wymaga otrzymania min. **70%** maksymalnej liczby punktów
- Ocena końcowa wyznaczana jest na podstawie sumy otrzymanych punktów za kolokwia:

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

Efekty uczenia się i system ich oceniania

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

EU3	definiuje i wykorzystuje własne funkcje w programach w języku C
------------	---

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

Efekt uczenia się (EU3)

- Student, który zaliczył przedmiot:

definiuje i wykorzystuje własne funkcje w programach w języku C

- Student, który zalicza na ocenę **dostateczny (3)** 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
 - umieć definiować tablice dwuwymiarowe i wykonywać podstawowe operacje tablicowe przy wykorzystaniu własnych funkcji języka C
 - znać i stosować proste struktury

Efekt uczenia się (EU3)

- Student, który zalicza na ocenę **dostateczny (3)** powinien (cd.):
 - znać sposób przechowywania i inicjalizacji łańcuchów znaków (napisów), umieć wykonywać operacje na łańcuchach znaków przy wykorzystaniu funkcji z pliku nagłówkowego string.h
 - umieć przydzielić dynamicznie i zwolnić pamięć na tablicę jednowymiarową (wektor)
- Student, który zalicza na ocenę **dobry (4)** powinien (oprócz wymagań na ocenę 3):
 - rozróżnić pojęcia deklaracji i definicji funkcji, stosować we właściwy sposób prototypy funkcji
 - umieć wykonywać zaawansowane operacje na tablicach dwuwymiarowych przy wykorzystaniu własnych funkcji języka C
 - umieć stosować złożone struktury
 - umieć przydzielić dynamicznie i zwolnić pamięć na tablicę dwuwymiarową (macierz)

Efekt uczenia się (EU3)

- Student, który zalicza na ocenę **bardzo dobry (5)** 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
 - umieć deklarować, inicjalizować i wykonywać operacje na tablicach wielowymiarowych z wykorzystaniem własnych funkcji języka C

Efekt uczenia się (EU4)

- Student, który zaliczył przedmiot:
stosuje operacje zapisu i odczytu plików w samodzielnie napisanych programach komputerowych
- Student, który zalicza na ocenę **dostateczny (3)** 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)

Efekt uczenia się (EU4)

- Student, który zalicza na ocenę **dostateczny (3)** powinien (cd.):
 - umieć napisać i skompilować program w języku C składający się z kilku modułów (plików z kodem źródłowym)
- Student, który zalicza na ocenę **dobry (4)** 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)** powinien (oprócz wymagań na ocenę 4):
 - umieć pisać programy przetwarzające jednocześnie wiele plików tekstowych i binarnych