



Nowoczesny C++ i STL

Sprawy organizacyjne



Kontakt

- ▶ Wykładowca:
Paweł Rzechonek
- ▶ Mail:
prz@cs.uni.wroc.pl
- ▶ Konsultacje:
Instytut Informatyki U.Wr. pokój 308 (III piętro)
- ▶ Materiały do wykładu i zadania laboratoryjne:
<https://www.ii.uni.wroc.pl/~prz/>



Cele

➤ Cele kursu:

- Wykształcenie umiejętności używania wszystkich dostępnych narzędzi językowych do efektywnego programowania.
- Zapoznanie z obiektowymi i funkcyjnymi technikami w najnowszej wersji języka C++.
- WYROBIENIE nawyku szukania gotowych rozwiązań w bibliotece standardowej STL oraz ich maksymalnego wykorzystania.

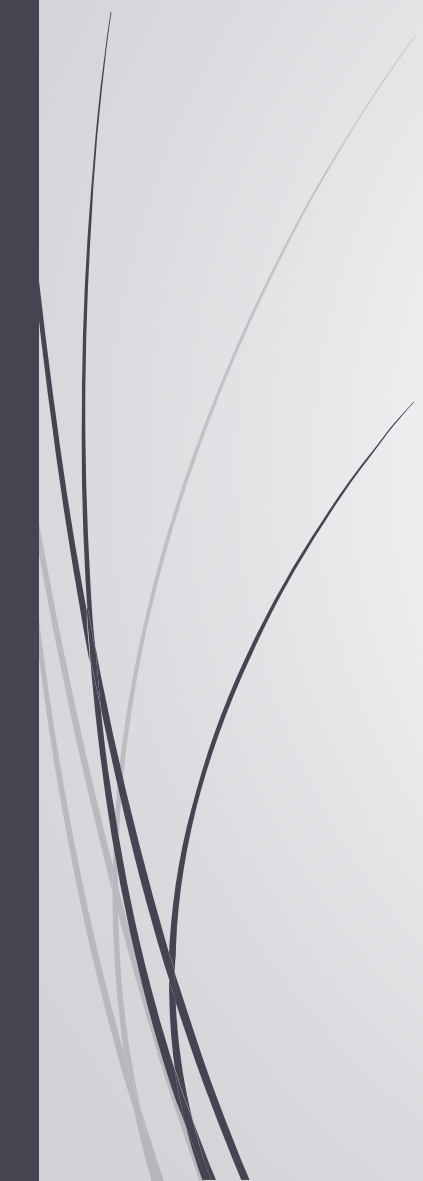


Wymagania

- ▶ Wymagania wstępne :
 - ▶ Umiejętność programowania obiektowego w języku C++.
 - ▶ Umiejętność czytania anglojęzycznej dokumentacji.
 - ▶ Podstawy algorytmiki i systemów operacyjnych.



Wykład

- ▶ Programowanie obiektowe w C++ z maksymalnym wykorzystaniem szablonów.
 - ▶ Używanie szablonów różnych funkcji i klas z biblioteki standardowej STL.
- 



Laboratorium

- ▶ W semestrze będzie do zrealizowania około 12 prostych zadań.
- ▶ Za każde zaprogramowane zadanie będzie można dostać do 10 punktów.
- ▶ Aby zaliczyć kurs należy do końca semestru zgromadzić co najmniej 50% z możliwych do zdobycia punktów.
- ▶ Zadania należy oddawać w wyznaczonym terminie.
- ▶ Studenci powinni osobiście prezentować swoje programy w czasie trwania ćwiczeń laboratoryjnych i odpowiadać na zadawane pytania dotyczące zadania.



Literatura

- N.M.Josuttis: *C++. Biblioteka standardowa. Wydanie 2.* Wydawnictwo Helion, Gliwice 2014.
- J.Galowicz: *C++17 STL. Receptury.* Wydawnictwo Helion, Gliwice 2018.
- S.Meyers: *Skuteczny nowoczesny C++. 42 sposoby lepszego postępowania się językami C++11 i C++14.* Wydawnictwo Helion, Gliwice 2016.
- B.Stroustrup: *Język C++. Kompendium wiedzy. Wydanie 4.* Wydawnictwo Helion, Gliwice 2014.



Szczegółowy spis treści



- nowe elementy języka
- inteligentne wskaźniki
- czasomierze
- obsługa błędów i wyjątków
- pary i krotki
- klasy narzędziowe
- kontenery i adaptatory kontenerów
- iteratory i adaptatory iteratorów
- algorytmy niemodyfikujące i modyfikujące
- obiekty funkcyjne i lambdy
- łańcuchy i konwersje liczbowe
- wyrażenia regularne
- lokalizacja
- liczby i rozkłady pseudolosowe
- współbieżność i synchronizacja wątków