

**POLITECHNIKA WROCŁAWSKA  
WYDZIAŁ INFORMATYKI I TELEKOMUNIKACJI**

**Karol Puchała**

# **Programowanie Obiektowe**

---

**INEW00003**

**Mini PPROJEKT**

Kierunek:

Informatyka techniczna (ITE)  
Teleinformatyka

**Wrocław**

## PROJEKT - REALIZACJA

Zadaniem grupy projektowej jest zaprojektowanie, implementacja w wybranym języku programowania obiektowego (do wyboru **C++ lub Java**) aplikacji oraz wykonanie dokumentacji.

### ZESPOŁY PROJEKTOWE

- ✓ Zespoły projektowe powinny składać się z **dwóch** lub ewentualnie **trzech** osób (po wyrażeniu zgody przez prowadzącego).
  - W przypadku grupy projektowej składającej się z trzech osób złożoność projektu **musi być większa** niż dla grup dwuosobowych.
- ✓ Skład zespołu projektowego **musi** być uzgodniony i zgłoszony prowadzącemu. (pierwszy etap projektu)
- ✓ Członkowie zespołu ponoszą **wspólną** odpowiedzialność za całość projektu.
- ✓ Każda grupa projektowa musi mieć przedstawiciela – **lidera**, który będzie dodatkowo (poza obowiązkami związanymi z realizacją projektu) odpowiedzialny za:
  - Koordynowanie prac członków zespołu.
  - Komunikację z prowadzącym.
  - Przesyłanie dokumentów związanych z rozliczeniem kolejnych etapów projektu.
    - Natomiast każdy członek zespołu będzie zobowiązany do **potwierdzenia** wysłania zadania w systemie ePortal)
  - Lider zespołu powinien rozdzielić zadania dla każdego członka grupy

### ZAWARTOŚĆ PROJEKTU

Projekt składa się z dwóch części:

#### I Aplikacja/Program:

1. Kod programu
2. Komentarze w kodzie
3. Plik wykonywalny

Przy sprawdzaniu aplikacji, ocenie poddawane będą umiejętności i poprawności definiowania klas:

- >czy student definiuje własne struktury klas,
- >czy używa tylko klas predefiniowanych;
- >czy klasy zdefiniowane przez studenta są hermetyczne;
- >czy i jak wykorzystywane są mechanizmy dziedziczenia oraz kompozycji;
- >czy zaimplementowany został oraz w jakim stopniu polimorfizm;
- >czy wykorzystywane są klasy abstrakcyjne, interfejsy itp.).

Również ocenie będzie podlegał interfejs aplikacji z użytkownikami oraz poprawność działania aplikacji.

## II Dokumentacja:

1. Temat projektu wraz ze składem grupy projektowej z wyszczególnieniem lidera.
2. Spis narzędzi użytych przy tworzeniu projektu.
3. Założenia i opis funkcjonalny programu.
4. Klasy
  - 4.1. Diagram UML własnych klas
  - 4.2. Kod własnych klas
5. Opis GUI programu\*
6. Uwagi i wnioski

---

### DODATKOWE ELEMENTY OCENIANIA

1. Estetyka oraz czytelność dokumentacji.
2. Estetyka oraz czytelność kodu.

## ETAPY PROJEKTU – KROKI MIŁOWE:

Projekt należy przedstawiać prowadzącemu etapami w wyznaczonych terminach, które zostaną ogłoszone na ePortalu.

Gotowy dokumenty należy przesłać na ePortal-> **zgłaszając go do oceny.**

### Do oddania będzie pięć etapów:

#### I Stworzenie grupy ETAP1.

Termin oddania : **daty zostaną podane na e-portalu.**

Należy przygotować dokument zawierający następujące informacje:

1. Dane studentów wchodzących w skład grupy. Wskazanie, który ze studentów będzie pełnił rolę **lidera.**;
2. Należy wskazać język obiektowy, w którym zostanie napisany program projektowy. Wyboru dokonujemy na ePortalu.

**Zostanie nadany Państwa grupie numer, którym należy posługiwać się podczas kolejnych zajęć projektowych.**

## II Wybór tematu ETAP2.

Termin oddania: **daty zostaną podane na e-portalu.**

1. Grupa przedstawia dwa tematy projektu.
2. Dla każdego z tematów należy przygotować opis założeń oraz opis funkcjonalny.

Prowadzący wybiera jeden z tych tematów lub zleca ponowne sporządzenie opisu.

Nieprzyjęcie przez prowadzącego żadnego z zaprezentowanych tematów oznacza, że etap projektu nie zostaje zaliczony.

## III Diagram klas ETAP3. - [ocena: F11].

Termin oddania na do: **daty zostaną podane na e-portalu.**

Studenci projektują aplikację w języku UML – projektują diagramy klas uwzględniając przy tym **wymagania dotyczące projektu.**

1. Należy przygotować dokumentację projektową zawierającą:
  - 1.1. Diagram UML klas projektowanego systemu

### WAŻNA UWAGA:

PODCZAS PROJEKTOWANIU SYSTEMU I USTALANIU POZIOMU HERMETYZACJI METOD I KLAS NALEŻY MIEĆ NA UWADZE PEWNE OGRANICZENIE!

NIE MOŻECIE PAŃSTWO W SWOICH PROGRAMACH UŻYWAĆ FUNKCJI TYPU **GET()** I **SET()**.  
JEST TO ZABRONIONE PODCZAS TEGO PROJEKTU!

## IV Finalna wersja aplikacji -> implementacja (ETAP4) [ocena: F12]

Studenci kodują oraz uruchamiają aplikację obiektywnym języku programowania, korzystając z wybranych narzędzi.

Termin oddania do: **daty zostaną podane na e-portalu.**

Należy:

1. **Finalny** kod programu stworzony na podstawie projektu przedstawionego w „Diagram klas ETAP3”.
2. Zaprezentować działanie programu.

### UWAGA:

Aplikacja, która nie kompilują się poprawnie otrzymują 0 punktów.

Będzie to jednoznaczne z otrzymaniem oceny NDST z całego laboratorium.

## V Dokumentacja projektu - (ETAP5)

Studenci wykonują dokumentację projektową, prezentują prowadzącemu działające aplikacje i przedstawiają do oceny wykonaną dokumentację w terminie do **czternastych** zajęć. [ocena: F2]. **daty zostaną podane na e-portalu.**

---

### ZALICZENIE ETAPÓW PROJEKTU:

- I Terminy rozliczenia się z etapów zostaną podane przez prowadzącego na zajęciach. Niedotrzymywanie terminów skutkować będzie obniżeniem ocen za projekt.
  - ✓ Za każdy tydzień spóźnienia dowolnego etapu Studenci w grupie otrzymuje minusa „-”. Jest to jednoznaczne z obniżeniem oceny końcowej projektu o **0.5** stopnia.
    - UWAGA: **ETAP 4 i 5** należy przedstawić najpóźniej do wyznaczonych dat, spóźnienie jak wyżej, ocena niedostateczna z projektu!
    - Poniższe zasady nie dotyczą **ETAPÓW 4 i 5!**
  - ✓ Na każdych zajęciach jest możliwość rozliczenia się z bieżącego lub zaległych etapów .
  - ✓ Zaległości w etapach należy nadrabiać sukcesywnie.

---

✓ **MIEDZY ODDAWANIEM ETAPÓW ISTNIEJE KARENCA 1 TYGODNIA ZAJĘĆ.**

**Wszystkie ETAPY muszą zostać oddane [zgłoszone do oceny na ePortalu] i pozytywnie ocenione do 14 zajęć!**

- II Koniecznym jest oddanie każdego z etapów projektu. Aby zaliczyć krok milowy projektu, każda z grup musi:
  - ✓ Załadować wymagane dokumenty na E-portal. →(dokument **musi** być załadowany na e-portal **przed** spotkaniem z prowadzącym oraz potwierdzony przez każdego członka z grupy – **zgłoszony do oceny**).
  - ✓ Stawić się na zajęciach (w pełnym składzie) i omówić z prowadzącym wskazany etap (**najpóźniej** w dniu wskazanym jako deadline).
- III W przypadku niespełnienia chociaż jednego z tych warunków, etap uznaje się za niezaliczony.

## WYMAGANIA – DOTYCZĄCE PROJEKTU

W każdym projekcie powinny zostać zaimplementować następujące mechanizmy programowania obiektowego:

- ✓ definiowanie klas [MIN. 3 klasy]
- ✓ hermetyzacja danych i metod
- ✓ dziedziczenie i kompozycja
- ✓ przeciążanie metod i funkcji
- ✓ polimorfizm

dodatkowo :

- ✓ GUI graficzny interfejs użytkownika-

Przy czym, dla uzyskania poszczególnych ocen wystarczy zaimplementować:

L.P.	OCENA	PUNKTY MAX	ZAGADNIENIE
1	3,0	55	zdefiniować minimum trzy klasy, między którymi będzie zachodziło uogólnienie (dziedziczenie) <a href="#">[patrz UML slajd nr 3 ( 22 )]</a>
2	3,5	65	P1 + hermetyzacja <a href="#">[patrz UML slajd nr 2 ( 21 )]</a> oraz implementacja odpowiednich konstruktorów
3	4,0	75	P2 + [minimum 4 klasy] + implementacja <b>zależności</b> między klasami <a href="#">[patrz UML slajd nr 4 ( 23 )]</a>
4	4,5	85	P3 + przeciążenie metod
5	5,0	100	P4 + Polimorfizm i klasy abstrakcyjne

Poziom złożoności będzie sprawdzony oraz zostaną wystawione oceny częściowe F11 oraz F12

## W PROJEKCIE OCENIANE BĘDĄ:

- ▶ Jakość wykonania:
  - ▶ Kod nie powinien zawierać błędów uniemożliwiających kompilację (programy niekompilujące się otrzymują zero punktów za cały projekt).
  - ▶ Poprawność zaprojektowania i zaimplementowania programu:
  - ▶ Czytelność kodu (komentarze, formatowanie).
  - ▶ Estetyka oraz czytelność dokumentacji.
- ▶ Sposób doboru technik projektowania obiektowego do realizowanego zadania.
- ▶ Pomysłowość zaimplementowanego rozwiązania.
- ▶ Złożoność oprogramowania.
- ▶ Terminowość oddawania zadań (kary za przekroczenie terminów oddania poszczególnych etapów)