

**NAZWA PRZEDMIOTU/MODUŁU KSZTAŁCENIA:**

**Procesing i programowanie**

**Kod przedmiotu: GSO\_20**

**Rodzaj przedmiotu: obieralny**

**Specjalność: Projektowanie gier i rzeczywistości wirtualnej**

**Wydział: Informatyki**

**Kierunek: Grafika**

**Poziom studiów: pierwszego stopnia - VI poziom PRK**

**Profil studiów: praktyczny**

**Forma studiów: stacjonarna/niestacjonarna**

**Rok: 3**

**Semestr: 5, 6**

**Formy zajęć i liczba godzin:**

**Forma stacjonarna**

**wyklady – 18 ( 10 + 8);**

**laboratorium – 44 ( 20 + 24 );**

**Forma niestacjonarna**

**wyklady – 12 ( 6 + 6 );**

**laboratorium – 26 ( 12 + 14 );**

**Zajęcia prowadzone są w języku polskim.**

**Liczba punktów ECTS: 7 ( 4 + 3 )**

**Osoby prowadzące:**

**wykład:**

**laboratorium :**

---

**1. Założenia i cele przedmiotu:**

Celem kursu przedmiotowego jest zapoznanie studentów z problematyką programowania grafiki komputerowej w środowisku Processing. Studenci zajmujący się grafiką komputerową zwykle wykorzystują specjalizowane/profesjonalne narzędzia do przetwarzania obrazu, tworzenia animacji czy filmów. Wykorzystanie środowiska programistycznego Processing, opartego na języku Java, pozwana studentom zdobyć pewne doświadczenie programistyczne do zastosowań graficznych i samodzielnie opracowywać algorytmy przekształceń obrazu/animacji/filmu. Celem przedmiotu jest przekazanie studentom wiedzy i umiejętności pozwalających na łączenie gotowych animacji w sekwencje, skryptowanie gotowych blueprintów, „obróbka cyfrowa” komponentów w grach drogą programistyczną. W zakresie treści przedmiotowych obecne są również elementy programowania obiektowego, opisu sceny, modeli, oświetlenia, geometrycznych przekształceń afinicznych, w tym zastosowanie OpenGL i zewnętrznych komponentów dedykowanych do Processing.

## 2. Określenie przedmiotów wprowadzających wraz z wymaganiami wstępnymi:

Podstawowa znajomość języka programowania Java i projektowania obiektowego

Przedmioty wprowadzające to: Podstawy programowania, Podstawy animacji.

## 3. Opis form zajęć

### a) *Wykłady*

#### • **Treści programowe:**

- Omówienie zintegrowanego środowiska programistycznego IDE Processing.
- Podstawy programowania Java, wykorzystanie funkcji setup i draw. Funkcje graficzne.
- Interfejs użytkownika, obsługa klawiatury i myszy, konsola i debugowanie programu.
- Modele barw, ekstrakcja składowych koloru, algorytmy przekształcania obrazu.
- Przykładowe programowanie gry Snake, tworzenie animacji komputerowych,
- Programowanie wizualizacji computer-arts, systemy dynamiczne, system odwzorowań iterowanych.

#### • **Metody dydaktyczne**

- Wykład prowadzony metodą tradycyjną z wykorzystaniem rzutnika multimedialnego i prezentacją środowiska Processing.

#### • **Forma i warunki zaliczenia:**

- Warunkiem zaliczenia przedmiotu jest wykonanie projektu polegającego na implementacji programu do przetwarzania obrazu, wytworzenia specjalnych efektów wizualnych, animacji komputerowej lub prostej gry

#### • **Wykaz literatury podstawowej:**

1. D. Shiffmann: Learning Processing - A Beginner's Guide to Programming Images, Animation, and Interaction; Morgan-Kaufmann, 2008.
2. M. Pearson: Generative Art.; Manning Publications, 2011.
3. A. S. Glassner: Processing for Visual Artists: How to Create Expressive Images and Interactive Art.; A K Peters, 2010

#### • **Wykaz literatury uzupełniającej**

1. H. Bohnacker, B. Gross, J. Laub: Generative Design: Visualize, Program, and Create with Processing, Princeton Architectural Press, 2012
2. Ben Fry: Visualizing Data; O'Reilly, 2007.
3. Casey Reas and Ben Fry: Processing: A Programming Handbook for Visual Designers, Second Edition, The MIT Press, 2014

### b) *Laboratorium*

#### • **Treści programowe:**

- Wprowadzenie do środowiska Processing, omówienie elementów IDE, zasady edycji kodu źródłowego programu, kompilacji i uruchamiania projektu, debugowanie programu, dostęp do dokumentacji funkcji składowych i konfiguracji. Powtórzenie z podstaw programowania w języku Java.

- Zasadnicze funkcje programu: setup i draw. Ustawienia okna programu, wykorzystanie funkcji tworzących obiekty geometryczne, wykorzystanie palety kolorów i modeli barw, wypełnianie figur i konturów. Przegląd dostępnych predefiniowanych funkcji graficznych.
- Przetwarzania obrazu, dostęp do pikseli obrazu, wydobywanie składowych koloru RGB, manipulacja na pikselach obrazu, progowanie, wzmacnianie/zmniejszanie kontrastu, jasności, intensywności koloru, przekształcenia obrazu do poziomu szarości. Zagadnienia computer-arts, wizualizacja obiektów dynamicznych.
- Animacje komputerowe, wykonanie prostej gry 2D. Obsługa klawiatury i myszki, komponowanie logiki gry, wykrywanie kolizji obiektów, sterowanie obiektami. Konstrukcja klas w języku Java.
- Przekształcenia afiniczne: translacja, rotacja i skalowanie obiektów geometrycznych, scena, model, oświetlenie, zagadnienia tekstuowania, wykorzystanie operacji macierzowych, stosu macierzy przekształceń, budowa sceny 3D
- **Metody dydaktyczne:**
  - Ćwiczenia laboratoryjne dotyczą zastosowania środowiska programistycznego Processing do tworzenia grafiki komputerowej, w tym przekształcania obrazu, tworzenia wizualizacji w technice computer-arts, programowania gry komputerowej 2D oraz projektowania sceny. Studenci będą rozwiązywać zadania problemowe, wyrażać swoje wizje w zapisie języka programowania Java oraz realizować przykładowe projekty. Całość prac laboratoryjnych ma na celu wskazanie studentom, że u podstaw grafiki komputerowej leżą formalizmy matematyczne oraz ich programowanie. Dzięki pozyskaniu wiedzy teoretycznej oraz zdobyciu umiejętności programowania grafiki studenci będą mogli wytwarzać dzieła bez konieczności wykorzystania drogich narzędzi komercyjnych. Processing należy do grupy wolnego oprogramowania.
- **Forma i warunki zaliczenia:**
  - Przygotowaniu projektu grafiki przy użyciu Processing w kategorii: computer-arts, prostej gry komputerowej lub przetwarzania obrazu lub animacji komputerowej.
- **Wykaz literatury podstawowej:**
  1. D. Shiffmann: Learning Processing - A Beginner's Guide to Programming Images, Animation, and Interaction; Morgan-Kaufmann, 2008.
  2. M. Pearson: Generative Art.; Manning Publications, 2011.
  3. A. S. Glassner: Processing for Visual Artists: How to Create Expressive Images and Interactive Art.; A K Peters, 2010
- **Wykaz literatury uzupełniającej**
  1. H. Bohnacker, B. Gross, J. Laub: Generative Design: Visualize, Program, and Create with Processing, Princeton Architectural Press, 2012
  2. Ben Fry: Visualizing Data; O'Reilly, 2007.
  3. Casey Reas and Ben Fry: Processing: A Programming Handbook for Visual Designers, Second Edition, The MIT Press, 2014

#### 4. Opis sposobu wyznaczania punktów ECTS

##### a. forma stacjonarna

Forma zajęć	Formy aktywności studenta	Średnia liczba godzin na zrealizowanie aktywności
Wykład	Kontakt z nauczycielem	18
	Czytanie wskazanej literatury	22
Laboratorium	Kontakt z nauczycielem	44
	Realizacja zadań dodatkowych	10
	Czytanie wskazanej literatury	10
	Projekt indywidualny	51
	Przygotowanie do przeglądu zaliczeniowego	20

<b>Całkowita ilość godzin aktywności studenta</b>	<b>175</b>
<b>Liczba punktów ECTS dla modułu/przedmiotu</b>	<b>7</b>

##### b. forma niestacjonarna

Forma zajęć	Formy aktywności studenta	Średnia liczba godzin na zrealizowanie aktywności
Wykład	Kontakt z nauczycielem	12
	Czytanie wskazanej literatury	28
Laboratorium	Kontakt z nauczycielem	26
	Realizacja zadań dodatkowych	15
	Czytanie wskazanej literatury	15
	Projekt indywidualny	59
	Przygotowanie do przeglądu zaliczeniowego	20

<b>Całkowita ilość godzin aktywności studenta</b>	<b>175</b>
<b>Liczba punktów ECTS dla modułu/przedmiotu</b>	<b>7</b>

#### 5. Wskaźniki sumaryczne

##### a. forma stacjonarna

- a) liczba godzin dydaktycznych (tzw. kontaktowych) i liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich
  - Liczba godzin kontaktowych – 62
  - Liczba punktów ECTS – 2,5
- b) liczba godzin dydaktycznych (tzw. kontaktowych) i liczba punktów ECTS na zajęciach o charakterze praktycznym.
  - Liczba godzin kontaktowych – 44
  - Liczba punktów ECTS – 5,4

##### b. forma niestacjonarna

- a) liczba godzin dydaktycznych (tzw. kontaktowych) i liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich
- Liczba godzin kontaktowych – 38
  - Liczba punktów ECTS – 1,9
- b) liczba godzin dydaktycznych (tzw. kontaktowych) i liczba punktów ECTS na zajęciach o charakterze praktycznym.
- Liczba godzin kontaktowych – 26
  - Liczba punktów ECTS – 5,4

## 5. Zakładane efekty kształcenia

Efekt przedmiotowy (Symbol)	Efekty kształcenia dla przedmiotu	Odniesienie do kierunkowych efektów kształcenia
GSO_20_W01	zna podstawy programowania w języku Java w środowisku Processing, zna strukturę programu, dedykowane funkcje graficzne, zna zasady obsługi klawiatury i myszki, kompilacji i debugowania programu.	<b>K_W08</b>
GSO_20_W02	zna podstawy przetwarzania obrazu, manipulacje kontrastem, jasnością, intensywnością koloru, zna metody progowania, generowania negatywu, wykonywania zniekształceń na obrazie. Zna modele barw, wydobywania składowych barw z pikseli obrazu.	<b>K_W03</b>
GSO_20_W03	zna zasady tworzenia animacji za pomocą programowania, zna techniki i algorytmy wizualizacji computer-arts, w tym systemy dynamiczne, układ iterowanych odwzorowań IFS.	<b>K_W03</b>
GSO_20_U01	potrafi opracować prosty program w Javie w środowisku Processing. Umie dobrać właściwy algorytm do rozwiązania postawionego problemu, potrafi go zaimplementować, debugować i uruchomić. Umie wykorzystać dokumentację projektu Processing, odpowiednio wykorzystać dostępne funkcje graficzne. Potrafi wykonać prosty interfejs użytkownika, w tym umożliwić obsługę klawiatury i myszki.	<b>K_U04</b>
GSO_20_U02	potrafi oprogramować wybrane zagadnienia przekształcania obrazu, w tym umie przygotować procedury do zmiany kontrastu, jasności i intensywności kolorów obrazu, umie operować na reprezentacji tablicowej obrazu, umie manipulować na pikselach.	<b>K_U01 K_U10</b>
GSO_20_U03	potrafi posługiwać się Processingiem do budowy animacji komputerowej, kreowania sceny 2D/3D, zastosować operacje na macierzach przekształceń, umie wykonać wizualizację systemów dynamicznych, systemów iterowanych odwzorowań IFS. Umie wytworzyć prostą grę 2D.	<b>K_U01 K_U07</b>
GSO_20_K01	potrafi pracować samodzielnie lub zespołowo, umie wykazać kreatywność lub przewodzić grupie, organizować realizację przedsięwzięcia z zachowaniem bezpieczeństwa, higieny i ergonomii pracy.	<b>K_K09</b>
GSO_20_K02	rozumie potrzebę ustawicznego rozwoju intelektualnego, w szczególności w zakresie szybko rozwijającej się dziedziny nowych technologii oraz dziedzinowego języka obcego.	<b>K_K01</b>

## 6. Odniesienie efektów kształcenia do form zajęć i sposób oceny osiągnięcia przez studenta efektów kształcenia.

Efekt przedmiotowy (Symbol)	Forma zajęć		Sposób sprawdzenia osiągnięcia efektu
	Wykład	Laboratorium	

GSO_20_W01	v		Ocena projektu
GSO_20_W02	v		Ocena projektu
GSO_20_W03	v		Ocena projektu
GSO_20_U01		v	Sprawozdanie z realizacji projektu
GSO_20_U02		v	Sprawozdanie z realizacji projektu
GSO_20_U03		v	Sprawozdanie z realizacji projektu
GSO_20_K01		v	Obserwacja pracy studenta lub zespołu projektowego
GSO_20_K02		v	Ocena doboru literatury w dokumentacji projektu

### 7. Kryteria uznania osiągnięcia przez studenta efektów kształcenia.

Efekt przedmiotowy (Symbol)	Efekt jest uznawany za osiągnięty, gdy:
<p>GSO_20_W01 GSO_20_W02 GSO_20_W03</p>	<p>Student zna podstawy programowania w języku Java w środowisku Processing, zna strukturę programu, dedykowane funkcje graficzne, zna zasady obsługi klawiatury i myszki, kompilacji i debugowania programu oraz zna podstawy przetwarzania obrazu, manipulacje kontrastem, jasnością, intensywnością koloru, zna metody progowania, generowania negatywu, wykonywania zniekształceń na obrazie. Zna modele barw, wydobywania składowych barw z pikseli obrazu. Zna zasady tworzenia animacji za pomocą programowania, zna techniki i algorytmy wizualizacji computer-arts, w tym systemy dynamiczne, układ iterowanych odwzorowań IFS.</p> <p>Projekt powinien zawierać:</p> <ol style="list-style-type: none"> <li>1. wstęp,</li> <li>2. analizę problemu i propozycję jego rozwiązania,</li> <li>3. kod źródłowy w Javie w środowisku Processing,</li> <li>4. autorskie obserwacje i wnioski.</li> </ol> <p>Ocenie podlegają:</p> <ol style="list-style-type: none"> <li>1. stopień jakości analizy zadania i powziętych założeń do jego rozwiązania,</li> <li>2. poziom przejrzystości kodu źródłowego projektu układu (czytelność językowa, zestaw komentarzy, zachowanie właściwej struktury zdań, wykorzystanie zewnętrznych bibliotek itp.),</li> <li>3. autorska treść obserwacji i wniosków, w tym poziom ich rzetelności badawczej.</li> </ol>
<p>GSO_20_U01 GSO_20_U02 GSO_20_U03</p>	<p>Ocenia się postęp w realizacji zadania projektowego - sprawozdanie <b>częściowe</b>.</p> <p>Student w ramach prac sprawozdawczych powinien wykazać umiejętności w zakresie: opracowania prostego programu w Javie w środowisku Processing, doboru właściwych algorytmów do rozwiązania postawionego problemu, implementowania, debugowania i uruchomienia. Student również powinien udowodnić, że umie wykorzystać dokumentację projektu Processing, odpowiednio wykorzystać dostępne funkcje graficzne, wykonać prosty interfejs użytkownika, w tym umożliwić obsługę klawiatury i myszki o ile jest taka potrzeba. Student również powinien wykazać, że potrafi oprogramować wybrane zagadnienia przekształcania obrazu, w tym przygotować procedury do zmiany kontrastu, jasności i intensywności kolorów obrazu, operować na reprezentacji tablicowej obrazu, manipulować na pikselach, posługiwać się Processingiem do budowy animacji komputerowej, kreowania sceny 2D/3D, zastosować operacje na macierzach przekształceń, wykonać wizualizację systemów dynamicznych, systemów iterowanych odwzorowań IFS.</p> <p>W zależności od tematyki projektu, udział poszczególnych składowych efektów kształcenia w sprawozdaniu oceniany jest w <b>różnym</b> stopniu.</p>

	<p><b>W szczególności ocenie podlegają:</b></p> <ol style="list-style-type: none"> <li>1. warstwa merytoryczna realizacji projektu,</li> <li>2. poziom zachowania odpowiedniej formy sprawozdania,</li> <li>3. poziom językowy wypowiedzi,</li> <li>4. zachowanie terminów doręczenia,</li> <li>5. precyzja określenia udziałów współautorów w realizacji sprawozdania (dla projektów grupowych),</li> <li>6. stopień wykorzystania źródeł dziedzinowych, w tym w języku angielskim,</li> <li>7. walor jakości współpracy autorów sprawozdania,</li> <li>8. postęp prac projektowych.</li> </ol>
GSO_20_K01	<p>Obserwacja pracy studenta i zespołu projektowego. Ocenie podlega:</p> <ol style="list-style-type: none"> <li>1. walor organizowania pracy indywidualnej,</li> <li>2. stopień zachowania zasad ergonomii, bezpieczeństwa i higieny pracy,</li> <li>3. kreatywność i oddziaływanie na grupę projektową,</li> <li>4. jakość współpracy w grupie, ewentualnie jej przewodniczenie.</li> </ol>
GSO_20_K02	<p>Ocena doboru literatury w dokumentacji projektu. Ocena podlega:</p> <ol style="list-style-type: none"> <li>1. tematyczny dobór literatury krajowej i międzynarodowej</li> <li>2. stopień wykorzystania książek, czasopism elektronicznych, artykułów naukowych czy sieci Internet,</li> <li>3. znajomość wykorzystanej literatury w cytowanym zakresie,</li> <li>4. krytyczność wobec postulowanych, literaturowych rozwiązań.</li> </ol>
<b>Ocena końcowa za przedmiot</b>	<p>Średnia ocen z realizacji poszczególnych efektów kształcenia (GSO_20_W01 do GSO_20_K03), przy czym każdy rodzaj efektów kształcenia (wiedza, umiejętności, kompetencje) powinien być pozytywnie oceniony dla zaliczenia przedmiotu.</p>