



Утвърдил:

/ Декан ФМИ /

Утвърден от Факултетен съвет

с протокол № /

СОФИЙСКИ УНИВЕРСИТЕТ "СВ. КЛИМЕНТ ОХРИДСКИ"

Факултет по Математика и Информатика

Специалност: **Информатика, Компютърни науки, Софтуерно инженерство**

--	--	--	--	--	--	--	--	--	--

Курс: **първи**

Учебна година: **2016/2017**

Семестър: **летен**

УЧЕБНА ПРОГРАМА

Дисциплина: **Обектно-ориентирано програмиране**

Object-Oriented Programming (на англ. език)

--	--	--	--

Тип: Задължителна дисциплината

Преподавател:

Асистент:

Учебна заетост	Форма	Хорариум
Аудиторна заетост	Лекции	45
	Семинарни упражнения	30
	Практически упражнения (хоспетиране)	0
Обща аудиторна заетост		75
Извънаудиторна заетост	Подготовка на домашни работи	40
	Учебен проект	40
	Самостоятелна работа в библиотека или с интернет ресурси	30
	Подготовка за изпит	10
Обща извънаудиторна заетост		120
ОБЩА ЗАЕТОСТ		195
Кредити аудиторна заетост		2,5
Кредити извънаудиторна заетост		3,5
ОБЩО ЕСТК		6,5

№	Формиране на оценката по дисциплината ¹	% от оценката
1.	Учебен проект	25%
2.	Изпит – практика	40%
3.	Изпит – теория	35%

Анотация на учебната дисциплина:

Курсът въвежда студентите в теорията и практиката на обектно-ориентираното програмиране на базата на езика C++. Покриват се основните теми, като например: Класове и обекти, конструктори и деструктори, приятели на класове, наследяване, виртуални функции, абстрактни класове, полиморфизъм и др. Освен тези теми, в рамките на курса се разглежда и работата с файлове, разделната компилация, работата с препроцесора в C++, библиотеката STL и шаблоните за дизайн (design patterns)

В рамките на курса всеки студент ще има възможност да работи по курсов проект, който покрива разглежданите в курса теми.

Предварителни изисквания:

1. Успешно да бъде овладян материалът по дисциплината „Увод в програмирането“ и да са постигнати нейните очаквани резултати.
2. Способност за работа с учебни материали на английски език.

Очаквани резултати:

След успешно завършване на курса се очаква студентите:

1. Да могат да проектират, създават, дебъгват и тестват обектно-ориентирани програми на C++, по зададени спецификации, като архитектурата на програмите отговаря на добрите практики на обектно-ориентирания стил;
2. Да могат да проектират, оформят и организират своите програми съгласно съвременните добри практики;
3. Да имат добро разбиране за процесите, които протичат в една обектно-ориентирана програма на C++;
4. Да познават библиотеката STL и да могат да използват основните класове в нея;
5. Да могат да работят с текстови и двоични файлове;

¹ В зависимост от спецификата на учебната дисциплина и изискванията на преподавателя е възможно да се добавят необходимите форми, или да се премахнат ненужните.

6. Да могат да работят с шаблони (templates) в C++.
7. Да знаят какво представляват шаблоните за дизайн (design patterns), да познават и да могат да имплементират основните шаблони за дизайн от „Gang of Four”

Учебно съдържание

№	Тема:	Хорариум	
		Л	У
1.	Препроцесор	3	2
2.	Разделна компилация	2	2
3.	Работа с команден интерпретатор. Получаване на аргументи от командния ред.	2	2
4.	Работа с файлове	3	3
5.	Структури	3	2
6.	Класове.	5	5
7.	Конструктори и деструктори	3	3
8.	Предефиниране на оператори	3	3
9.	Шаблони (templates)	3	3
10.	STL	3	2
11.	Приятелите на класове	1	1
12.	Наследяване	3	3
13.	Виртуални функции. Абстрактни класове. Полиморфизъм.	2	2
14.	Шаблони за дизайн (design patterns)	3	3

Конспект за изпит

№	Въпрос
1.	Препроцесор. Основни директиви.
2.	Разделна компилация. Добри практики.
3.	Работа с команден интерпретатор. Получаване на аргументи от командния ред.
4.	Работа с файлове. Текстови и двоични файлове. Класове за работа с файлове в стандартната библиотека.
5.	Структури.
6.	Класове. Основни понятия.

7.	Конструктори и деструктори. Голяма четворка.
8.	Предефиниране на оператори
9.	Шаблони. Особенности при изнасянето на шаблон в отделен файл.
10.	STL
11.	Приятели на класове
12.	Наследяване. Множествено наследяване. Особенности.
13.	Виртуални функции. Абстрактни класове. Полиморфизъм.
14.	Шаблони за дизайн (design patterns)

Библиография

1. "Learn C++," [Online]. Available: <http://www.learncpp.com/>.
2. "CProgramming.com," [Online]. Available: <http://www.cprogramming.com/>.
3. "C++ Reference," [Online]. Available: <http://www.cplusplus.com/reference/>.
4. "C++ Language Tutorial," [Online]. Available: <http://www.cplusplus.com/doc/tutorial/>.
5. B. Stroustrup, The C++ Programming Language, 3 ed., Addison-Wesley, 1997.
6. B. Stroustrup, Principles and Practice Using C++, Addison-Wesley, 2008.
7. М. Тодорова, П. Армянов and К. Николов, Сборник от задачи по програмиране на C++: част втора, Обектно-ориентирано програмиране, Технологиика, 2008.
8. E. Gamma, R. Helm, R. Johnson and J. Vlissides, Design Patterns: Elements of Reusable Object-oriented Software, Addison-Wesley, 1995.
9. Е. Гама, Р. Хелм, Р. Джонсън и Д. Влсидес, Шаблони за дизайн: Елементи на обектно-ориентирания софтуер за многократно използване, София: СофтПрес, 2005.
10. S. Meyers, Effective C++, 3 ed., Addison-Wesley, 2005.
11. S. Meyers, More Effective C++, Addison-Wesley, 1995.
12. B. Klemens, 21st Century C: C Tips from the New School, 2nd ed. O'Reilly Media, 2014.
13. B. Overland, C++ Without Fear: A Beginner's Guide That Makes You Feel Smart (2nd Edition), 2011

Дата: февруари 2017 г.

Съставил: доц. Калин Георгиев
доц. Атанас Семерджиев
доц. Трифон Трифонов
гл. ас. Петър Армянов

Прието на заседание на катедра „Компютърна Информатика” –

протокол № ___ от __. __. 2017 г.