



СОФИЙСКИ УНИВЕРСИТЕТ "СВ. КЛИМЕНТ ОХРИДСКИ"
ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА

УЧЕБНА ПРОГРАМА

Утвърдил:

/декан/

ОКС „бакалавър”

Утвърдена с решение на ФС с протокол:

Избираема дисциплина

№ 8 от 25.06.2018

редовна форма на обучение									
Специалност:	(код и наименование)								
Информатика, Информационни системи, Компютърни науки, Математика и информатика, Софтуерно инженерство									

Дисциплина:	(код и наименование)	Ж	7	2	7
Практическа роботика и умни "неща"					
Practical Robotics and Smart Things					

Учебната програма е разработена и предложена за утвърждаване от катедра:	
Информационни технологии	
от:	ас. Траян Илиев

Преподавателските екипи се утвърждават ежегодно от Факултетен съвет.

Заетост и кредити		
Обща заетост:		150
Кредити:		5
Учебна заетост	Форма	Хорариум
Аудиторна заетост	Лекции	30
	Семинарни упражнения	0
	Практически упражнения (хоспитиране)	30
Обща аудиторна заетост:		60
Кредити аудиторна заетост:		2
Извънаудиторна заетост	Подготовка на домашни работи	10
	Контролни работи и подготовка за тях	
	Учебен проект	
	Самосотятелна работа в библиотека или с интернет ресурси	10
	Доклад/Презентация	10
	Курсов учебен проект	50
	Подготовка за финален тестови изпит	10
Обща извънаудиторна заетост:		90
Кредити извънаудиторна заетост:		3

Предвидена форма на оценяване:	КИ
И - изпит, КИ - комбинирано изпитване; ТО - текущо оценяване	

Формиране на оценката по дисциплината		
№	Показател	%
1	Финален курсов проект	40%
2	Финален тестови изпит	40%
3	Домашни работи през семестъра (текущ контрол)	10%
4	Работа в час (задачи по време на упражнения)	10%

Анотация на учебната дисциплина

Интелигентните устройства са навсякъде около нас – автомобилите и домовете стават „умни“, дрехите включват „интелигентни материи“ и незабележимо вградена електроника (wearable electronics), почти всички електроуреди и предмети на бита придобиват способности за комуникация помежду си и с отдалечени услуги „в облака“ (cloud computing). Често взаимодействието става с гласови команди (Amazon Alexa). В крайна сметка цялата тази информация и възможност за контрол стават достъпни за нас през удобен вграден, мобилен или гласов интерфейс от всяко място, по всяко време (consumer application dashboards).

С помощта на множество практически проекти, курсът запознава с бързо-развиващата се област на „Интернет на нещата“ (Internet of Things – IoT), обслужващата и социална роботика (service and social robotics). Акцентът е върху придобиване на реален опит при реализация на вградени и мрежово свързани устройства и малки роботи (Arduino, Raspberry Pi 2/3/Zero, ESP 8266, Lego, сензори, актуатори и др.) и най-вече върху начина, по който тези устройства комуникират помежду си и със заобикалящия ги свят (социална способност).

За да реализираме на практика тези проекти се нуждаем от рамка за моделиране на взаимодействието между устройствата, разглеждани като интелигентни агенти (Intelligent Agents - IA) в рамките на една многоагентна система (Multi-Agent System - MAS). Разглеждат се основните характеристики на интелигентните агенти - автономност, способност за реагиране, проактивност, способност за самообучение (адаптивност), социална способност (езици за комуникация между агенти - ACLs), рационалност, мобилност, както и различни парадигми и архитектури за реализация – йерархична (базирана на планиране), реактивна (Subsumption Architecture), и хибридна. Включено е запознаване с Belief-Desire-Intention (BDI) модел на практически разсъждения като основа за моделиране на света, комуникация и автономно вземане на решения от агентите. Знанията се представят и реферират с помощта на онтологии и Semantic Web W3C стандарти (RDF/RDFS, OWL).

Курсът изгражда цялостна картина на необходимите технологии започвайки от хардуерния слой (практически проекти с Arduino, Raspberry Pi и ESP 8266 + най-различни сензори и актуатори: ултразвукови и инфрачервени за дистанция, оптично масиви за следене на линия, камери, енкодери, двигатели и драйвери, серво- механизми за хващане на предмети и движение на камерата, сензорни TFT екрани и други), през обработката на събитийни потоци в реално време с помощта на микро-контролери и серийна USB (UART), I2C, SPI, комуникация, софтуерни библиотеки от ниско ниво (Java: LeJOS, Pi4J), до горните слоеве на приложението за реактивна обработка на събития реализирани със Spring Reactor, Akka, Eclipse IoT (Kura, Paho, Californium) платформи.

Следващите нива в IoT/ Service Robotics архитектурата включват механизми и протоколи за комуникация - Constrained Application Protocol (CoAP), MQ Telemetry Transport (MQTT), HTTP, уеб услуги (REpresentational State Transfer - REST APIs), "облачни технологии" (Docker, Kubernetes, Apache Brooklyn, Ansible), бази от данни в реално време (Prometheus, Graphite, InfluxDB, RethinkDB).

Включени са също така и технологии за изграждане на вградени (Java Swing, JavaFX) и мобилни (Angular + TypeScript + Material Design / Ionic) уеб интерфейси и интерактивни "табла за управление" (dashboards), както и технологии за визуализация и анализ на събития в реално време (Grafana).

Целта е освен практическите проекти реализирани по време на упражненията, участниците да сформират екипи за реализиране на собствени роботи/ IoT много-агентни системи и/или компоненти за тях, които да бъдат демонстрирани в края на курса. Дългосрочната цел е сформирание на екип за участие в национални и международни състезания от типа на RoboCup (<http://www.robocup.org/>), където роботи изпълняват задачи с различна степен на сложност – играят футбол, спасяват хора при бедствия и аварии, кооперират се с хора за изпълнение на задачи у дома, в промишлеността и логистиката.

Информация за някои от роботите, които ще сглобяваме и програмираме по време на курса можете да намерите на адрес: <http://robolearn.org/>.

Кодът е достъпен в GitHub: <https://github.com/iproduct/course-social-robotics>

Предварителни изисквания

Очаква се студентите да могат да боравят свободно с технически английски език. Необходимо е добро познаване на езика Java, желателно и на езика Python. Препоръчителна, но не задължителна, е също базова техническа грамотност за работа с цифрова схемотехника.

Очаквани резултати

По време на курса студентите ще придобият знания и умения за:

- многоагентните системи с разпределен изкуствен интелект, кооперативната, социалната и домашната роботика и автоматизация;
- обектно-ориентирано, актьор-ориентирано и агентно-ориентирно софтуерно архитектурно моделиране, реактивно програмиране (reactive programming) и actor model – **Spring Reactor, RxJava, RxJS, Akka**;
- реализация на вградени интерфейси за IoT устройства и роботи с **Java Swing /Java FX** на ARM Linux платформа;
- реализация на мобилни уеб/ native интерфейси за IoT устройства и роботи с **Angular 5, TypeScript, Material Design / Ionic**;
- практически опит с платформи **Raspberry Pi 3, Lego® Mindstorms EV3** и **Arduino** за изграждане на малки роботи и IoT устройства;
- свързване на сензори (*ултразвукови и инфрачервени за дистанция, оптично масиви за следене на линия, камери, енкодери*), актуатори (*двигатели и драйвери, серво-механизми за хващане на предмети и движение на камерата*), вградени интерфейси (сензорни TFT екрани - *touchscreens*), и комуникационни модули (*WiFi, Bluetooth Low Energy - BLE, ESP 8266 System-On-Chip*);
- **IoT** архитектури и **протоколи за комуникация (MQTT, CoAP, AMQP)** и конфигуриране на устройства (**OMA-DM, LWM2M**);
- практическа реализация на IoT Greenhouse проект с **Eclipse IoT** платформа (*Kura, Paho, Californium* – MQTT и CoAP комуникационни протоколи);
- Визуализация на данни в реално време с **Grafana**;
- Java® програмиране на **Lego®** роботи с използване на библиотеката **LeJOS**;

• Java® програмиране на Raspberry Pi 3 роботи и мрежово-свързани, вградени устройства с използване на библиотеката Pi4J ;
• управление сензори и актуатори през GPIO интерфейс – протоколи: Serial UART, I2C, SPI ;
• програмиране на Arduino хардуер, сензори, актуатори и комуникационни протоколи с използване на Arduino IDE / Eclipse (C++) ;
• обработка на изображения и компютърно зрение с библиотеката OpenCV ;
• планиране на действия и манипулиране на обекти със STRIPS (<i>Stanford Research Institute Problem Solver</i>);
• Belief–Desire–Intention (BDI) модел на човешките практически разсъждения и езици за комуникация между интелигентни агенти;
• представяне на знания с помощта на онтологии и W3C стандарти за изграждане на Семантична Мрежа (Semantic Web) : <i>Resource Description Framework (RDF)</i> и <i>Web Ontology Language (OWL)</i> .
Студентите ще изградят също умения за целенасочено търсене, анализ и употреба на информация, както и умения за работа в екип по избран от тях практически <i>IoT / Robotics</i> проект.

Учебно съдържание		
№	Тема	Хорариум л. / с.упр. / пр.
1	Представяне на курса, въведение в областта на „Интернет на нещата“ (<i>Internet of Things – IoT</i>), обслужващата и социална роботика (<i>service and social robotics</i>), умни домове (<i>smart homes</i>), автономни транспортни средства (<i>autonomous vehicles</i>), интелигентни материи (<i>smart fabrics & wearable electronics</i>), агрегация и обработка на сложни събития (<i>Complex Event Processing - CEP</i>) в облака (<i>cloud computing</i>). Разпределена обработка на потоци от събития в реално време. Ламбда архитектура: <i>Apache Spark, Storm, Kafka, Apex, Flink, Beam</i> . Многослойна референтна архитектура на <i>IoT. Platform as a Service (PaaS)</i> . Наблюдение и управление на <i>IoT</i> мрежи, устройства и роботи чрез вградени и мобилни уеб/ <i>native</i> интерфейси – <i>application dashboards</i> с <i>Grafana</i> .	4
2	Приложение на многоагентните системи с разпределен изкуствен интелект в кооперативната, социалната и домашната роботика и автоматизация. Обектно-ориентирано, актьор-ориентирано и агентно-ориентирано софтуерно инженерство. Реактивно програмиране (<i>reactive programming</i>) и <i>actor model</i> – <i>Spring Reactor, RxJava, RxJS, Akka</i> . Отличителни характеристики на интелигентните агенти. Видове интелигентни агенти. Интелигентни агенти с хардуерна архитектура – роботика. Колаборативна и социална роботика. Хардуерни платформи за изграждане на малки роботи и <i>IoT</i> устройства – запознаване с <i>Raspberry Pi 2, Lego® Mindstorms EV3</i> и <i>Arduino</i> .	4

3	Рационални агенти – <i>AIMA (Artificial Intelligence: Modern Approach)</i> агенти. Структура на рационални агенти. Агенти базирани на таблица. Агенти с прости рефлексии. Агенти с вътрешно състояние. Агенти базирани на цели. Агенти базирани на полезност. Агентни среди – видове и свойства. Програмна реализация на агенти и агентни среди. Примери.	4
4	Изграждане на роботи с <i>Lego® Mindstorms</i> . Програмиране на <i>Lego®</i> роботи. <i>Java®</i> програмиране на <i>Lego®</i> роботи с използване на <i>LeJOS – Wifi</i> конфигуриране, разработка с <i>LeJOS</i> , билд файлове, използване на сензори, мотори, инструменти, <i>ssh</i> достъп до <i>EV3</i> блока, примерни програми и практически упражнения с <i>LeJaRo</i> робот и <i>LeJOS –</i> избягване на препятствия, хващане на предмети, следване на линия.	4
5	Програмиране на <i>Raspberry Pi 3</i> роботи и IoT устройства с използване на <i>Java</i> библиотеката <i>Pi4J</i> . Управление на сензори и актуатори през <i>GPIO</i> интерфейс – основни протоколи за комуникация: <i>Serial, I2C, SPI</i> . Елементи на схемотехниката – аналогово-цифрови преобразуватели, <i>level shifters</i> и др. Сглобяване на малък робот с <i>Raspberry Pi 3</i> . Управление на драйвера на двигателите през <i>GPIO</i> с <i>Pulse Width Modulation (PWM)</i> . Добавяне на сензори за дистанция и програмиране на прости рефлексии у <i>Raspberry Pi</i> робота.	4
6	Хардуерна и софтуерна платформа с отворен код <i>Arduino</i> – хардуер, сензори и актуатори, комуникационни протоколи, езици за програмиране, среди за разработка <i>Arduino IDE / Eclipse (C++)</i> . Сглобяване и програмиране на примерни <i>Arduino</i> проекти със сензори и актуатори.	4
7	<i>Интернет на нещата (IoT)</i> - практически проекти с <i>ESP 8266 WiFi SOC (System-On-Chip)</i> модули с интегриран <i>TCP/IP</i> стек от протоколи. <i>IoT</i> архитектури и протоколи за комуникация (<i>MQTT, CoAP, AMQP</i>) и конфигуриране на устройства (<i>OMA-DM, LWM2M</i>). <i>Eclipse IoT платформа</i> , проекти: <i>Edje, Paho, Wakaama, Kura, OpenHAB/SmartHome, Californium, Mosquitto, Leshan, Hono, hawkBit, BIRT</i> . Реализация на примерен проект <i>Greenhouse Tutorial</i> с <i>Eclipse Kura, Paho</i> и <i>Californium</i> .	4
8	Взаимодействие и кооперация между хора и интелигентни агенти с хардуерна архитектура (роботи). Проектиране и реализация на графични интерфейси за вградени устройства и роботи с <i>Java Swing / Java FX</i> на <i>ARM Linux</i> платформа.	4
9	Разработка на уеб интерфейси за управление на <i>IoT</i> устройства с <i>JavaScript</i> библиотеките <i>Angular 2, TypeScript, Material Design. Angular CLI. Ionic 2</i> платформа за разработка на <i>native</i> мобилни приложения с <i>Angular</i> и <i>TypeScript</i> .	4

10	<p><i>Akka toolkit</i> за разработка на масивно конкурентни, разпределени и устойчиви към грешки събитийно-ориентирани <i>актьор (actor) системи</i> – референции, състояние, поведение, наследници, супервайзинг стратегия, прекъсване. Наблюдение и контрол – стратегии <i>One-For-One</i> и <i>All-For-One</i>. Делегиране на задачи на наследници. Референции, пътища и адреси на актьори. Селекция по път. Обработка на изключения и грешки. Жизнен цикъл на актьор – <i>DeathWatch</i>. Изпращане и получаване на съобщения – <i>tell, ask, reply</i>. Спиране на актьори. Динамично добавяне/замяна на поведения на актьори – <i>become</i> and <i>unbecome</i> (шаблон <i>State</i> Актьори със <i>stash</i>. Диспечери на съобщения. <i>Mailboxes</i> – типове конфигурация и приоритети. Маршрутизация на съобщения и стратегии за маршрутизация. Реализация на машини на състояние (крайни автомати) с помощта на актьори.</p>	4
11	<p>Компютърно зрение и разпознаване на обекти – запознаване с библиотеките <i>OpenCV-Java</i>, <i>OpenCV-Python</i> и <i>NumPi</i>. Практически примери за обработка на изображения и компютърно зрение с <i>Pi Camera Module v2</i></p>	4
12	<p>Планиране на действия и манипулиране на обекти - декомпозиция на проблема и алгоритми за планиране и извод. Представяне на състоянията целите и действията. Търсене на път в пространство на състоянията - <i>forward u backward chaining</i>. Евристично търсене. Граф на планиране (<i>planning graph</i>). Използване на съждителна логика. <i>STRIPS (Stanford Research Institute Problem Solver)</i> планиране. Практическа реализация на езика <i>Java</i>.</p>	4
13	<p>Самообучение и адаптивно поведение при интелигентни агенти. Проектиране на самообучаващи се системи. Индуктивно логическо самообучение - представяне на хипотези, научаването на понятие като търсене в пространство на хипотезите, пространство на версиите (<i>version space</i>), <i>inductive bias</i>. Използване на <i>дърво на решенията (decision tree)</i>. Използване на <i>невронни мрежи (neural networks)</i>. <i>Deep learning</i>.</p>	4
14	<p><i>FIPA</i> стандарти на за изграждане на многоагентни системи. <i>Belief-Desire-Intention (BDI)</i> модел на човешките практически разсъждения и езици за комуникация между интелигентни агенти (<i>Agent Communication Languages – ACLs</i>). Представяне на знания с помощта на онтологии и <i>W3C</i> стандарти за изграждане на <i>Семантична Мрежа (Semantic Web)</i>: <i>Resource Description Framework (RDF)</i> и <i>Web Ontology Language (OWL)</i>. Практическа реализация на онтология с <i>Protege</i>. Достъп до онтологията с <i>Java - Apache Jena Ontology API</i>.</p>	4
15	<p>Финален тест и обсъждане на курсовите проекти. Допълнителни въпроси. Перспективи пред <i>IoT</i> и социалната роботика.</p>	4

Конспект за изпит	
№	Въпрос

1	„Интернет на нещата“ (<i>Internet of Things – IoT</i>), обслужващата и социална роботика (<i>service and social robotics</i>), умни домове (<i>smart homes</i>), автономни транспортни средства (<i>autonomous vehicles</i>), интелигентни материи (<i>smart fabrics & wearable electronics</i>), агрегация и обработка на сложни събития (<i>Complex Event Processing - CEP</i>) в облака (<i>cloud computing</i>). Разпределена обработка на потоци от събития в реално време. Ламбда архитектура: <i>Apache Spark, Storm, Kafka, Apex, Flink, Beam</i> . Многослойна референтна архитектура на <i>IoT. Platform as a Service (PaaS)</i> . Наблюдение и управление на <i>IoT</i> мрежи, устройства и работи чрез вградени и мобилни уеб/ <i>native</i> интерфейси – <i>application dashboards</i> с <i>Grafana</i> .
2	Приложение на многоагентните системи с разпределен изкуствен интелект в кооперативната, социалната и домашната роботика и автоматизация. Обектно-ориентирано, актьор-ориентирано и агентно-ориентирано софтуерно инженерство. Реактивно програмиране (<i>reactive programming</i>) и <i>actor model</i> – <i>Spring Reactor, RxJava, RxJS, Akka</i> . Отличителни характеристики на интелигентните агенти. Видове интелигентни агенти. Интелигентни агенти с хардуерна архитектура – роботика. Колаборативна и социална роботика. Платформи за изграждане на малки работи и <i>IoT</i> устройства – запознаване с <i>Raspberry Pi 2, Lego® Mindstorms EV3</i> и <i>Arduino</i> .
3	Рационални агенти – <i>AIMA (Artificial Intelligence: Modern Approach)</i> агенти. Структура на рационални агенти. Агенти базирани на таблица. Агенти с прости рефлексии. Агенти с вътрешно състояние. Агенти базирани на цели. Агенти базирани на полезност. Агентни среди – видове и свойства. Програмна реализация на агенти и агентни среди. Примери.
4	Изграждане на работи с <i>Lego® Mindstorms</i> . Програмиране на <i>Lego®</i> работи. <i>Java®</i> програмиране на <i>Lego®</i> работи с използване на <i>LeJOS – Wifi</i> конфигуриране, разработка с <i>LeJOS</i> , билд файлове, използване на сензори, мотори, инструменти, <i>ssh</i> достъп до <i>EV3</i> блока, примерни <i>LeJOS</i> програми – избягване на препятствия, хващане на предмети, следване на линия.
5	Програмиране на <i>Raspberry Pi 3</i> работи и <i>IoT</i> устройства с използване на <i>Java</i> библиотеката <i>Pi4J</i> . Управление на сензори и актуатори през <i>GPIO</i> интерфейс – основни протоколи за комуникация: <i>Serial, I2C, SPI</i> . Елементи на схемотехниката – аналогово-цифрови преобразуватели, <i>level shifters</i> и др. Сглобяване на малък робот с <i>Raspberry Pi 3</i> . Управление на драйвера на двигателите през <i>GPIO</i> с <i>Pulse Width Modulation (PWM)</i> . Сензори за дистанция и програмиране на прости рефлексии у <i>Raspberry Pi</i> робота.
6	Хардуерна и софтуерна платформа с отворен код <i>Arduino</i> – хардуер, сензори и актуатори, комуникационни протоколи, езици за програмиране, среди за разработка <i>Arduino IDE / Eclipse (C++)</i> . Програмиране на <i>Arduino</i> проекти със сензори и актуатори.
7	<i>Интернет на нещата (IoT)</i> - практически проекти с <i>ESP 8266 WiFi SOC (System-On-Chip)</i> модули с интегриран <i>TCP/IP</i> стек от протоколи. <i>IoT</i> архитектури и протоколи за комуникация (<i>MQTT, CoAP, AMQP</i>) и конфигуриране на устройства (<i>OMA-DM, LWM2M</i>). <i>Eclipse IoT платформа</i> , проекти: <i>Edje, Paho, Wakaama, Kura, OpenHAB/SmartHome, Californium, Paho, Mosquitto, Leshan, Hono, hawkBit, BIRT</i> .
8	Взаимодействие и кооперация между хора и интелигентни агенти с хардуерна архитектура (роботи). Проектиране и реализация на графични интерфейси за вградени устройства и работи с <i>Java Swing / Java FX</i> на <i>ARM Linux</i> платформа.
9	Разработка на уеб интерфейси за управление на <i>IoT</i> устройства с <i>JavaScript</i> библиотеките <i>Angular 2, TypeScript, Material Design. Angular CLI. Ionic 2</i> платформа за разработка на <i>native</i> мобилни приложения с <i>Angular</i> и <i>TypeScript</i> .

10	<i>Akka toolkit</i> за разработка на масивно конкурентни, разпределени и устойчиви към грешки събитийно-ориентирани <i>актьор (actor) системи</i> – референции, състояние, поведение, наследници, супервайзинг стратегия, прекъсване. Наблюдение и контрол – стратегии <i>One-For-One</i> и <i>All-For-One</i> . Делегиране на задачи на наследници. Референции, пътища и адреси на актьори. Селекция по път. Обработка на изключения и грешки. Жизнен цикъл на актьор – <i>DeathWatch</i> . Изпращане и получаване на съобщения – <i>tell, ask, reply</i> . Спиране на актьори. Динамично добавяне/замяна на поведения на актьори – <i>become</i> and <i>unbecome</i> (шаблон <i>State</i>). Актьори със <i>stash</i> . Диспечери на съобщения. <i>Mailboxes</i> – типове конфигурация и приоритети. Маршрутизация на съобщения и стратегии за маршрутизация. Реализация на машини на състояние (крайни автомати) с помощта на актьори.
11	Компютърно зрение и разпознаване на обекти – запознаване с библиотеките <i>OpenCV-Java</i> , <i>OpenCV-Python</i> и <i>NumPi</i> . Практически примери за обработка на изображения и компютърно зрение с <i>Pi Camera Module v2</i> .
12	Планиране на действия и манипулиране на обекти - декомпозиция на проблема и алгоритми за планиране и извод. Представяне на състоянията целите и действията. Търсене на път в пространство на състоянията - <i>forward</i> и <i>backward chaining</i> . Евристично търсене. Граф на планиране (<i>planning graph</i>). Използване на съждителна логика. <i>STRIPS (Stanford Research Institute Problem Solver)</i> планиране. Практическа реализация на езика <i>Java</i> .
13	Самообучение и адаптивно поведение при интелигентни агенти. Проектиране на самообучаващи се системи. Индуктивно логическо самообучение - представяне на хипотези, научаването на понятие като търсене в пространство на хипотезите, пространство на версиите (<i>version space</i>), <i>inductive bias</i> . Използване на <i>дърво на решенията (decision tree)</i> . Използване на <i>невронни мрежи (neural networks)</i> . <i>Deep learning</i> .
14	<i>FIPA</i> стандарти на за изграждане на многоагентни системи. <i>Belief-Desire-Intention (BDI)</i> модел на човешките практически разсъждения и езици за комуникация между интелигентни агенти (<i>Agent Communication Languages – ACLs</i>). Представяне на знания с помощта на онтологии и W3C стандарти за изграждане на <i>Семантична Мрежа (Semantic Web): Resource Description Framework (RDF)</i> и <i>Web Ontology Language (OWL)</i> . Практическа реализация на онтология с <i>Protege</i> . Достъп до онтологията с <i>Java - Apache Jena Ontology API</i> .

Библиография	
Основна	
1.	GitHub ресурси за роботика и IoT - https://github.com/iproduct/course-social-robotics/wiki
2.	Информация за роботите LeJaRo и IPTPI – http://robolearn.org/
3.	Сайт на W3C за семантична мрежа – http://www.w3.org/standards/semanticweb/
4.	Официален уеб сайт на Lego® Mindstorms – http://www.lego.com/en-us/mindstorms/?domainredir=mindstorms.lego.com
5.	Официален уеб сайт на LeJOS – http://sourceforge.net/projects/lejos/
6.	Сайт на проекта с отворен код Arduino – http://arduino.cc/
7.	Ресурси на български езика за платформата Arduino – http://playground.arduino.cc/Bulgarian/Nachalo
8.	Akka toolkit for concurrent & distributed applications – http://akka.io/
9.	Сайт на <i>Foundation for Intelligent Physical Agents (FIPA)</i> – http://www.fipa.org/
10.	Сайт на Spring Reactor библиотека за реактивно програмиране с Java – https://projectreactor.io/
11.	Сайт на RxJS библиотека за реактивно програмиране с JavaScript – https://github.com/ReactiveX/rxjs

12. Сайт на Akka платформа за разработка на actor-oriented приложения с Java/Scala – http://akka.io/
13. Сайт на Protege редактор за създаване на онтологии и изграждане на интелигентни системи – http://protege.stanford.edu/
14. Сайт на Apache Jena библиотека за достъп до онтологии и изграждане на приложения за "Семантична мрежа" с Java – https://jena.apache.org/
15. Сайт на ESP 8266 System-On-Chip (SOC) – http://www.esp8266.com/wiki
16. Murch R., Johnson T., Intelligent Software Agents, Prentice Hall, 1999
17. Russel S., Norvig P., Artificial Intelligence: a Modern Approach (3rd Ed.), Prentice Hall, 2009

Дата: 20.04.2018 г

Съставил: _____ ас. Траян Илиев

Програмата е приета на заседание на КС – протокол от

СПРАВКА ПРЕПОДАВАТЕЛСКИ ЕКИПИ

Информатика, Информационни системи, Компютърни науки, Математика и информатика, Софтуерно инженерство;

Практическа роботика и умни "неща" ; Ж727

[illegible]