

<b>Principali informazioni sull'insegnamento</b>	<b>A.A. 2017-2018</b>
Titolo insegnamento	Reti di Calcolatori
Corso di studio	Informatica
Crediti formativi	9
Denominazione inglese	Computer Networks
Obbligo di frequenza	No
Lingua di erogazione	Italiano

<b>Docente responsabile</b>	Nome Cognome	Indirizzo Mail
	Filippo Lanubile	filippo.lanubile@uniba.it
Luogo ed Orario di Ricevimento	Dip. Informatica 6° Piano	Venerdì dalle 12:00 alle 14:00

<b>Dettaglio credi formativi</b>	Ambito disciplinare	SSD	Crediti
	Informatico	ING-INF/05	9

<b>Modalità di erogazione</b>	
Periodo di erogazione	Primo Semestre
Anno di corso	Terzo Anno
Modalità di erogazione	Lezioni frontali Esercitazioni in aula e laboratorio

<b>Organizzazione della didattica</b>	
Ore totali	225
Ore di corso	86 (56 lezioni frontali e 30 esercitazioni/laboratorio)
Ore di studio individuale	139 (119 lezioni frontali e 20 esercitazioni/laboratorio)

<b>Calendario</b>	
Inizio attività didattiche	25 settembre 2017
Fine attività didattiche	12 gennaio 2018

<b>Syllabus</b>	
Prerequisiti	<p>Deve essere stato colmato l'eventuale debito formativo secondo quanto previsto dal Regolamento Didattico del Corso di Studi.</p> <p>Devono essere stati superati con successo gli esami dei seguenti insegnamenti: Programmazione., Architettura degli Elaboratori., Laboratorio di Informatica e Linguaggi di Programmazione.</p>
Risultati di apprendimento previsti (declinare rispetto ai Descrittori di Dublino) (si raccomanda che siano coerenti con i risultati di apprendimento del CdS, compreso i risultati di apprendimento trasversali)	<ul style="list-style-type: none"> <li><i>Conoscenza e capacità di comprensione</i> Conoscere i concetti fondamentali delle reti di calcolatori. Conoscere i principali servizi e protocolli per applicazioni di rete. Comprendere i meccanismi di indirizzamento, instradamento e trasporto in Internet Conoscere i concetti fondamentali della sicurezza di rete.</li> <li><i>Conoscenza e capacità di comprensione applicate</i></li> </ul>

	<p>Acquisire familiarità con la programmazione in rete basata su interfaccia socket.</p> <ul style="list-style-type: none"> <li>• <i>Autonomia di giudizio</i> Mostrare di aver acquisito autonomia di giudizio sulle scelte relative al funzionamento delle reti di calcolatori e dei protocolli di Internet.</li> <li>• <i>Abilità comunicative</i> Mostrare di essere in grado di comunicare in modo appropriato le caratteristiche tecniche delle reti di calcolatori e dei protocolli di Internet</li> <li>• <i>Capacità di apprendere</i> Mostrare di aver sviluppato capacità di intraprendere in autonomia ulteriori approfondimenti su argomenti attinenti le reti di calcolatori e Internet.</li> </ul>
<p>Contenuti di insegnamento</p>	<ul style="list-style-type: none"> <li>• Fondamenti di reti di calcolatori <ul style="list-style-type: none"> <li>- Introduzione alle reti di calcolatori. Definizioni base. Collegamenti diretti e indiretti. Strategie di multiplazione. Commutazione di circuito e di pacchetto. Caratterizzazione delle reti in base alla dimensione. Interconnessione di reti. Misure di prestazione.</li> <li>- Architetture a livelli Stratificazione. Protocolli di rete. Messaggi in una pila di protocolli. Modello di riferimento ISO/OSI. Architettura di Internet.</li> </ul> </li> <li>• Servizi e protocolli per applicazioni di rete <ul style="list-style-type: none"> <li>- Introduzione alle applicazioni di rete Definizioni base. Modelli client-server e peer-to-peer. Socket API. Scelta del tipo di servizio. Indirizzamento dei processi. Identificazione di un servizio. Concorrenza nei server. Interazione client/server orientata alla connessione e senza connessione. Collaudo di applicazioni di rete</li> <li>- Web Introduzione. Caratteristiche fondamentali. Identificatori di risorsa: URI, URN e URL. Architettura di un browser. Architettura di un server. HTTP: formato di una richiesta e di una risposta, metodi. MIME. Trasmissione di un form. Informazioni di stato: cookies.</li> <li>- DNS Nomi simbolici. Gerarchia dei nomi. Gerarchia dei server DNS. Risoluzione dei nomi. Caching e abbreviazioni. Descrittori di risorsa.</li> <li>- Programmazione delle socket Programmazione delle socket in C: interazione TCP client-server con le socket; costruire messaggi; funzioni DNS; usare Socket UDP.</li> </ul> </li> <li>• Interconnessione delle reti <ul style="list-style-type: none"> <li>- Trasporto in Internet</li> </ul> </li> </ul>

	<p>Il servizio di trasporto in Internet. UDP. Numeri di porta. TCP. Meccanismo di ritrasmissione. Controllo del flusso. Apertura e chiusura di una connessione. Controllo di congestione.</p> <ul style="list-style-type: none"> <li>- Instradamento in Internet Obiettivo. La funzione di instradamento (routing). Costruzione della tabella di routing. Algoritmi di routing adattivi: distance vector routing e link-state routing. Routing gerarchico.</li> <li>- Indirizzamento in Internet IPv4 datagram. Classi di indirizzi IP. Notazione decimale a punti. Indirizzi speciali e privati. Assegnazione degli indirizzi IP. Subnetting e supernetting. CIDR. Inoltro dei datagrammi IP. DHCP. IPv6. NAT.</li> <li>• Sicurezza di rete <ul style="list-style-type: none"> <li>- Attacchi alla sicurezza</li> <li>- Firewall</li> <li>- Sistemi di rilevamento delle intrusioni</li> <li>- Principi di crittografia Crittografia a chiave simmetrica. Crittografia a chiave pubblica</li> <li>- Integrità dei messaggi Funzioni hash crittografiche. Codice di autenticazione dei messaggi. Firme digitali</li> <li>- Autenticazione end-to-end</li> <li>- Sicurezza a livello di applicazione: email</li> <li>- Sicurezza a livello di trasporto: SSL</li> <li>- Sicurezza a livello di rete: IPsec</li> </ul> </li> </ul>
--	--

<b>Programma</b>	
Testi di riferimento	<p>Teoria: J.F. Kurose &amp; K.W. Ross, Reti di calcolatori e Internet - Un approccio top-down, Pearson</p> <p>Esercitazioni e Laboratori: M. J. Donahoo, K. L. Calvert, TCP/IP Sockets in C: Practical Guide for Programmers. Morgan Kaufman Publishers.</p>
Note ai testi di riferimento	<p>Altri libri consigliati:</p> <p>Teoria: L. L. Peterson &amp; B.S. Davie, Reti di calcolatori, Apogeo, 2008 B. A. Forouzan, F. Mosharraf, Reti di Calcolatori - Un approccio top-down, McGraw-Hill, 2013</p> <p>Esercitazioni e Laboratori: B. Hall, Beej's Guide to Network Programming using Internet Sockets, Version 2.3.23, Nov. 5, 2005. <a href="http://beej.us/guide/bgnet/">http://beej.us/guide/bgnet/</a></p> <p>I libri di testo sono integrati con gli appunti presi a lezione e con le slide del docente disponibili sul sito web del corso.</p>
Metodi didattici	<p>Lezioni frontali supportate da slide, esercitazioni in aula, assegnazioni di esercizi di programmazione di rete con verifica in laboratorio.</p>

<p>Metodi di valutazione (indicare almeno la tipologia scritto, orale, altro)</p>	<p>Prova di laboratorio e prova scritta.          La prova scritta consiste nel rispondere a un questionario contenente domande a risposta chiusa o aperta.          La prova di laboratorio consiste nel completare un esercizio di programmazione in C con le socket.          Per sostenere la prova scritta occorre aver superato la prova di laboratorio.          Per gli studenti che hanno almeno il 70% di presenze del numero di ore di lezioni effettuate nel periodo precedente l'interruzione delle lezioni sono previste prove di laboratorio e scritte intermedie.</p>
<p>Criteri di valutazione (per ogni risultato di apprendimento atteso su indicato, descrivere cosa ci si aspetta lo studente conosca o sia in grado di fare e a quale livello al fine di dimostrare che un risultato di apprendimento è stato raggiunto e a quale livello)</p>	<p>Saranno valutati i risultati di apprendimento previsti. La prova di laboratorio è valutata in termini di idoneità a sostenere la prova scritta. La prova scritta è valutata in trentesimi.</p>
<p>Altro</p>	