

RTU studiju kurss "Robotu operētājsistēmas (ROS) pamati"**33000 Datorzinātnes, informācijas tehnoloģijas un enerģētikas fakultāte****Vispārējā informācija**

Kods	DE1116
Nosaukums	Robotu operētājsistēmas (ROS) pamati
Studiju kursa statuss programmā	Obligātais/Ierobežotās izvēles
Atbildīgais mācītspēks	Andrejs Zujevs - Doktors, Vadošais pētnieks
Mācītspēks	Maksims Terjohins - Zinātniskais asistents Artūrs Ardavs - Pētnieks
Apjoms daļās un kredītpunktos	1 daļa, 6.0 kredītpunkti
Studiju kursa īstenošanas valodas	LV
Anotācija	Robotu operētājsistēma (Robot Operating System - ROS) ir atklātā pirmkoda programmatūras satvars, kas nodrošina komunikācijas mehānismus, rīkus un programmatūras arhitektūras standartizētas veidnes, lai izstrādātu uzticamas, mērogojamas un uzturamas robotu sistēmas. ROS nav operētājsistēma to klasiskajā nozīmē, bet ir programmatūras bibliotēku kopums, kas darbojas esošo Linux vai Windows operētājsistēmu ietvaros un nodrošina mehānismus robotu vadībai (algoritmi un metodes), savstarpējai komunikācijai, sensoru datu apstrādei un robotu izpildmehānismu vadībai. ROS savas attīstības ceļā kļuva par de-facto standartu dažāda veida robotu, t. sk. robotizēto sistēmu vadības programmatūras izstrādē. Studiju kurss ļaus studentiem brīvi orientēties un pielietot ROS praksē dažādu robotu programmēšanai, vadībai un robotu simulācijai virtuālajā vidē. Tā kā ROS nemitīgi turpina savu attīstības ceļu, mainoties ROS versijām, arī studiju kursa saturs tiks atbilstoši atjaunots un papildināts.
Mērķis un uzdevumi, izteikti kompetencēs un prasmēs	Studiju kursa mērķis ir nodrošināt pamata teorētiskas zināšanas un praktiskas iemaņas darbam ar ROS. Studiju kursa uzdevumi: - veidot izpratni par ROS arhitektūru, darbības mehānismiem un rīkiem, veicināt spēju praktiski analizēt robotu sistēmu programmatūras uzbūvi un komunikācijas principus ROS satvarā; - veidot izpratni par ROS komunikācijas modeļiem (tēmas, servisi, darbības) un attīstīt prasmes datu apmaiņā starp robotu sistēmas komponentēm, nodrošinot drošu un efektīvu mezglu savstarpējo sadarbību; - veidot izpratni par ROS konkurējošo procesu pārvaldību, taimeriem, atbildes procedūrām, un ROS mezglu dzīves cikliem; - iemācīt studējošos veikt robotu modelēšanu un simulāciju ROS satvarā; - veidot izpratni par ROS augsta līmeņa programmatūras bibliotēku (navigācija, kustību plānošana, ārējās vides uztvere) integrāciju ar zema līmeņa aparāturu (motori, sensori, vadības mezgli); - iemācīt studējošiem izmantot ROS programmatūras izstrādes darbplūsmu, veikt savu ROS pakotņu izstrādi, kompilēšanu, datu vizualizāciju, robotu simulāciju, konteinerizāciju (Docker vidē), atklādošanu un notikumu reģistrēšanu žurnālos; - veidot izpratni par ROS programmatūras izvietošanu dažādās operētājsistēmās, par tīklu konfigurāciju, kā arī par laika sinhronizāciju daudzrobotu sistēmās; - iemācīt studējošiem orientēties ROS programmatūras veiktspējas analizē un optimizācijā, kā arī interpretēt un uzlabot sistēmas ātrdarbību; - iemācīt studējošiem izmantot ROS satvaru robotu navigācijai, to kustību plānošanai, sensoru integrēšanai, ārējās vides kartēšanai un robotu vadības algoritmu izmantošanai; - veidot izpratni par ROS programmatūras noturību pret kļūdām, tostarp drošu ROS pakotņu palaišanas un izbeigšanas secību, arī attīstīt spēju analizēt kļūdu scenārijus.
Patstāvīgais darbs, tā organizācija un uzdevumi	Patstāvīgo darbu izpilde paredz, ka studējošais savā personīgajā datorā uzstāda un izmanto ROS satvaru. Katrs patstāvīgais darbs ir veltīts konkrētai studiju kursa tēmai un ietver iepriekš sagatavotus programmatūras koda piemērus, kas atvieglo mācību procesu. Studiju kursa noslēgumā studējošais izstrādā gala projektu, kurā apvienotas visas kursa tēmas un demonstrētas studējošā spējas izstrādāt sarežģītus robotikas risinājumus, izmantojot ROS satvaru.
Literatūra	Obligātā. / Obligatory: Hernández Cordero, A. Learning ROS 2, Packt Publishing, 2023. Joseph, L. Robot Operating System (ROS) for Absolute Beginners – ROS 2 Edition, Apress, 2021. Quigley, M., Gerkey, B., & Smart, W. D. Programming Robots with ROS 2, O'Reilly Media, forthcoming edition (ROS 2 update). S. Macenski, T. Foote, B. Gerkey, C. Lalancette, W. Woodall. Robot Operating System 2: Design, architecture, and uses in the wild Science Robotics vol. 7, May 2022 L. Puck et al. Distributed and Synchronized Setup towards Real-Time Robotic Control using ROS2 on Linux 2020 IEEE 16th International Conference on Automation Science and Engineering (CASE), Hong Kong, China, 2020, pp. 1287-1293, doi: 10.1109/CASE48305.2020.9217010 Papildu. / Additional: Joseph, L., & Cacace, J. Mastering ROS 2 (2nd ed.), Packt Publishing, 2022. Gandhinathan, R., & Joseph, L. ROS 2 Robotics Projects (2nd ed.), Packt Publishing, 2022. Vinoski, S. Designing Distributed Systems with DDS, O'Reilly Media, 2018. Joseph, L. ROS Robotics Projects, Packit Publishing, 2017. Ruffin W., Gianluca C., Henrik C., and Agostino C.. Procedurally provisioned access control for robotic systems In IEEE/RSJ International Conference on Intelligent Robots and Systems, 2018
Nepieciešamās priekšzināšanas	Programmēšanas prasmes. Ieteicamas prasmes programmēšanai ar C++/Python un iemaņas darbam ar Linux komandrindu.

Studiju kursa saturs

Saturs	Pilna un nepilna laika klātienes studijas		Nepilna laika neklātienes studijas	
	Kontakt stundas	Patstāv. darbs	Kontakt stundas	Patstāv. darbs
ROS satvars arhitektūra un datu izplatīšanas serviss.	6	8	0	0
Komunikācijas modeļi.	8	8	0	0
Konkurējošo programmatūras procesu pārvaldība.	6	8	0	0
Robotu modelēšana un simulācija.	8	8	0	0
Augsta līmeņa algoritmu (navigācija, plānošana) integrācija ar zema līmeņa aparāturu (motori, sensori, vadības mezgli).	8	8	0	0
ROS programmatūras izstrādes darbplūsma.	10	8	0	0
ROS programmatūras izvietošana produkcijā.	6	8	0	0
Robotu navigācija un to kustību plānošana.	8	8	0	0
ROS programmatūras efektivitāte un noturība pret kļūdām.	6	8	0	0
Gala projekta izstrāde un aizstāvēšana (individuāli katram studējošam).	2	28	0	0
Kopā:	68	100	0	0

Sasniedzamie studiju rezultāti un to vērtēšana

Sasniedzamie studiju rezultāti	Rezultātu vērtēšanas metodes
Izprot ROS satvara arhitektūru un datu izplatīšanas servisu.	Metodes: 1. praktiskais darbs, kursa gala projekts. Kritēriji: studējošais demonstrē savu ROS programmu (kompilēšana un palaišana notiek bez kļūdām) un tā darbojas atbilstoši nodarbības tēmai.
Spēj darboties ar dažādiem ROS komunikācijas modeļiem.	Metodes: 2. praktiskais darbs, kursa gala projekts. Kritēriji: studējošais demonstrē savu ROS programmu (kompilēšana un palaišana notiek bez kļūdām) un tā darbojas atbilstoši nodarbības tēmai.
Spēj pārvaldīt konkurējošo programmatūras procesus.	Metodes: 3. praktiskais darbs, kursa gala projekts. Kritēriji: studējošais demonstrē savu ROS programmu (kompilēšana un palaišana notiek bez kļūdām) un tā darbojas atbilstoši nodarbības tēmai.
Spēj veikt robotu modelēšanu un simulāciju.	Metodes: 4. praktiskais darbs, kursa gala projekts. Kritēriji: studējošais demonstrē savu ROS programmu (kompilēšana un palaišana notiek bez kļūdām) un tā darbojas atbilstoši nodarbības tēmai.
Spēj integrēt augsta līmeņa algoritmus ar zema līmeņa aparāturu.	Metodes: 5. praktiskais darbs, kursa gala projekts. Kritēriji: studējošais demonstrē savu ROS programmu (kompilēšana un palaišana notiek bez kļūdām) un tā darbojas atbilstoši nodarbības tēmai.
Spēj strādāt ROS programmatūras izstrādes darbplūsmu.	Metodes: 6. praktiskais darbs, kursa gala projekts. Kritēriji: studējošais demonstrē savu ROS programmu (kompilēšana un palaišana notiek bez kļūdām) un tā darbojas atbilstoši nodarbības tēmai.
Spēj izvietot ROS programmatūru produkcijas vidē.	Metodes: 7. praktiskais darbs, kursa gala projekts. Kritēriji: studējošais demonstrē savu ROS programmu (kompilēšana un palaišana notiek bez kļūdām) un tā darbojas atbilstoši nodarbības tēmai.
Spēj veikt robotu navigāciju un kustību plānošanu.	Metodes: 8. praktiskais darbs, kursa gala projekts. Kritēriji: studējošais demonstrē savu ROS programmu (kompilēšana un palaišana notiek bez kļūdām) un tā darbojas atbilstoši nodarbības tēmai.
Spēj nodrošināt ROS programmatūras efektivitāti un noturību pret kļūdām.	Metodes: 9. praktiskais darbs, kursa gala projekts. Kritēriji: studējošais demonstrē savu ROS programmu (kompilēšana un palaišana notiek bez kļūdām) un tā darbojas atbilstoši nodarbības tēmai.

Studējošais izprot ROS satvara arhitektūru un datu izplatīšanas servisu, ROS komunikācijas modeļus un konkurējošo procesu pārvaldību, spēj modelēt un simulēt robotus, integrēt augsta līmeņa algoritmus ar zema līmeņa aparāturu, izstrādāt ROS programmatūru atbilstoši izstrādes darbplūsmai, realizēt robotu navigāciju un kustību plānošanu, kā arī nodrošināt ROS sistēmu efektivitāti un noturību pret kļūdām.	Kursa gala projekta aizstāvēšana: a) demonstrācija, kur gala projekta kompilēšana un palaišana notiek bez kļūdām; b) projektā vienoti ir integrētas visas kursa tēmas tā, lai demonstrētu studējoša reāla/simulēta robota vai robotizētas sistēmas darbību, izmantojot ROS satvaru; c) kritēriju saraksts par gala projektu tiks nodrošināts atsevišķi.
--	---

Studiju rezultātu vērtēšanas kritēriji

Kritērijs	% no kopējā vērtējuma
Praktiskie darbi	50
Kursa gala projekta aizstāvēšana	50
Kopā:	100

Studiju kursa plānojums

Daļa	KP	Stundas			Pārbaudījumi		
		Lekcijas	Prakt d.	Laborat	Ieskaite	Eksām.	Darbs
1.	6.0	24.0	44.0	0.0		*	