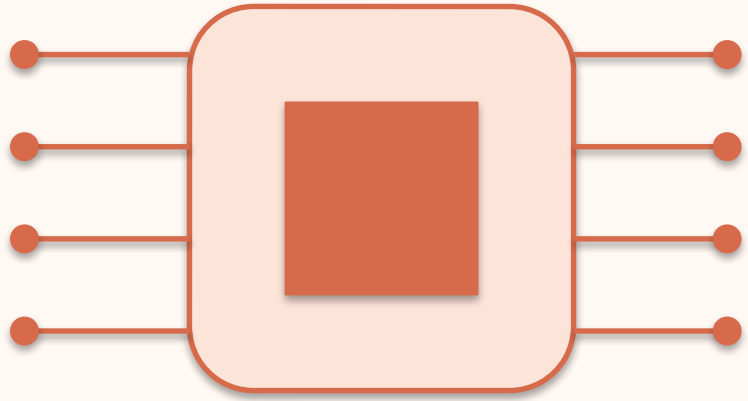
CODE / ALGORITHM

РОБОТТАРДЫ БАҒДАРЛАМАЛАУ НЕГІЗДЕРІ



Алгоритмдер • басқару логикасы • бағдарламалық архитектура

CONTROL • COMPUTE • CONNECT

Балжан Ибрагимқызы
PhD, ассистент-профессор

Дәрістің мақсаты және күтілетін нәтижелер

Роботты бағдарламалау — сенсор деректерін әрекетке айналдыратын басқару алгоритмін құру.

Мақсаты

Робот қозғалысын және сыртқы құрылғыларын бағдарламалық басқару қағидаларын меңгеру.

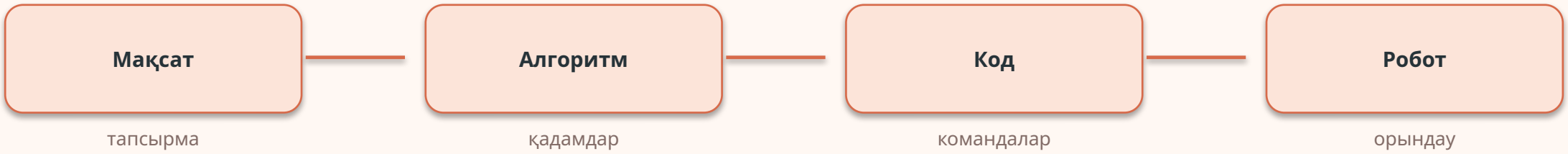
Міндеттері

Алгоритм, цикл, шарт, функция және кері байланыс ұғымдарын қарастыру.

Нәтиже

Қарапайым мобильді роботқа арналған алгоритм мен бағдарлама құрылымын әзірлеу.

Бағдарламалық басқару тізбегі



1. Роботты бағдарламалау дегеніміз не?

Бағдарлама роботтың сенсорлардан ақпарат алуын, шешім қабылдауын және жетектерді басқаруын анықтайды.

Кіріс деректері

Қашықтық, жарық, энкодер, камера және батырма сигналдары.

Өңдеу логикасы

Шарттар, математикалық есептеу, күй машинасы және жоспарлау.

Шығыс командалары

Мотор жылдамдығы, бұрылу, дыбыс, жарық және манипулятор әрекеті.

Бағдарламалық цикл

Оқу — сенсор

Есептеу — шешім

Әрекет — жетек

Қайталау — цикл

2. Алгоритм және оны көрсету тәсілдері

Алгоритм — есепті шешуге арналған нақты, реттелген және аяқталатын командалар тізбегі.

Сызықтық

Командалар бірінен кейін бірі орындалады. Мысал: 20 см алға жүру.

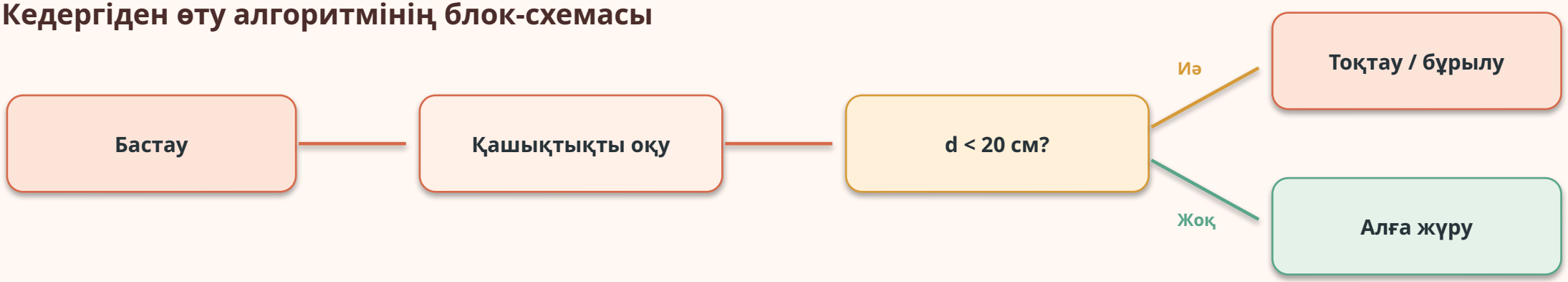
Тармақталу

Шартқа байланысты әртүрлі әрекет орындалады. Мысал: кедергі болса — тоқтау.

Циклдік

Командалар қайталанады. Мысал: жол бойы сенсорды үнемі тексеру.

Кедергіден өту алгоритмінің блок-схемасы



3. Бағдарламаның негізгі құрылымдары

Көптеген роботтық бағдарламаларда айнымалылар, шартты операторлар, циклдер және функциялар қолданылады.

Айнымалы

Сенсор мәнін немесе жылдамдықты сақтау

Шарт

if / else арқылы шешім қабылдау

Цикл

while / for арқылы қайталау

Функция

Кодты шағын бөліктерге бөлу

Мысал: шартты басқару логикасы

Егер қашықтық < 20 см болса → моторды тоқтату; әйтпесе → алға қозғалу.

4. Роботты басқару бағдарламасының архитектурасы

Бағдарламаны модульдерге бөлу кодты түсінуді, сынауды және қайта пайдалануды жеңілдетеді.

Сенсор модулі — өлшеу

Шешім модулі — логика

Қозғалыс модулі — моторлар

Қауіпсіздік — шектеулер

Артықшылықтары

Әр модульдің міндеті анық болады; қате табу және өзгеріс енгізу жеңілдейді.

Күй машинасы

Роботтың күйін (күту, жүру, бұрылу, тоқтау) айқын басқарып отырады.

5. Нақты уақыт және кері байланыс

Роботтың басқару бағдарламасы сенсор деректеріне уақытында жауап беруі тиіс.

Ашық контур

Команда беріледі, бірақ нәтиже қайта өлшенбейді. Қарапайым уақытпен басқару мысалы.

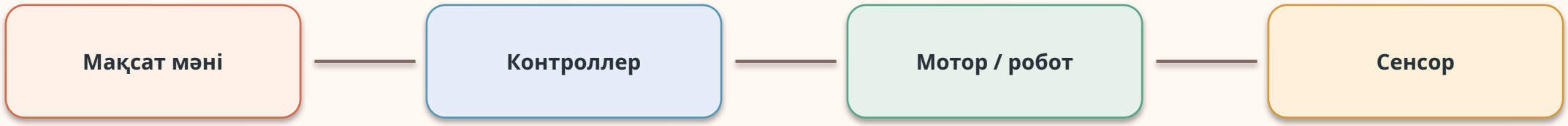
Жабық контур

Нақты нәтиже өлшеніп, мақсатпен салыстырылады; қате бойынша түзету жасалады.

Нақты уақыт

Өлшеу мен басқару белгіленген уақыт аралығында орындалуы керек.

Кері байланыс контуры



Сенсор өлшемі контроллерге қайтып келіп, келесі команданы түзетеді.

6. Роботты бағдарламалау ортасы мен тілдері

Платформа мен бағдарламалау тілі роботтың аппараттық мүмкіндігіне және жобаның күрделілігіне қарай таңдалады.

Блоктық орта

LEGO SPIKE App сияқты орталарда алгоритм блоктар арқылы құрастырылады.

Arduino IDE / C++

Микроконтроллерлерді басқаруға, сенсорлар мен моторларға код жазуға қолайлы.

Python / ROS 2

Жоғары деңгейлі логика, компьютерлік көру, навигация және роботтық жүйелерді біріктіру.

Құралды таңдау өлшемдері

Оқу жеңілдігі	Аппараттық қолдау	Масштабталу	Қауымдастық
алғашқы жобалар	плата / сенсор / мотор	жоба күрделілігі	кітапхана / құжаттама

7. Бағдарламаны тестілеу және жөндеу

Роботтық кодты алдымен жеке модульдерде, кейін толық жүйеде тексеру ұсынылады.

Модульдік тест

Функция немесе сенсор драйвері жеке тексеріледі.

Интеграциялық тест

Сенсор, контроллер және мотордың бірге жұмыс істеуі тексеріледі.

Жолдық сынақ

Робот нақты ортада қозғалып, шеткі жағдайлар бағаланады.

Тестілеу циклі

Жоспарлау — күтілетін нәтиже

Іске қосу — сынақ деректері

Бақылау — лог / өлшем

Түзету — қайта тексеру

Маңызды: қозғалыс аймағын босатып, жылдамдық шегін және апаттық тоқтатуды алдын ала тексеру.

Қорытынды және бақылау сұрақтары

Сапалы роботтық бағдарлама — дұрыс алгоритм, модульдік құрылым, кері байланыс және жүйелі тестілеудің нәтижесі.

Негізгі қорытынды

Алгоритм роботтың әрекетін сипаттайды; код оны контроллерде орындайды; сенсорлар нәтижені кері байланыстырады.

Инженерлік қағида

Кодты модульдерге бөліңіз, қауіпсіздік шектерін енгізіңіз және әр өзгерістен кейін сынаңыз.

Бақылау сұрақтары

1. Алгоритмнің негізгі қасиеттері қандай?
2. Шартты оператор мен цикл не үшін қолданылады?
3. Ашық және жабық контурдың айырмашылығы неде?
4. Неліктен бағдарламаны модульдерге бөледі?
5. Роботтық бағдарламаны тестілеудің кезеңдерін атаңыз.

Әдебиет: J. Craig, Introduction to Robotics; M. Quigley et al., Programming Robots with ROS.