

UYGA VAZIFA



Avtomobil 3 soat davomida 80 km/soat tezlik bilan, qolgan 2 soat davomida 90 km/soat tezlik bilan harakatlandi. Avtomobilning o'rtacha tezligini hisoblash dasturini tuzing. (*Eslatma: o'rtacha tezlikni hisoblash uchun umumiy masofani umumiy vaqtga bo'lish kerak*).

2-dars. SCRATCH MUHITIDA TARMOQLANUVCHI BLOKLAR BILAN ISHLASH

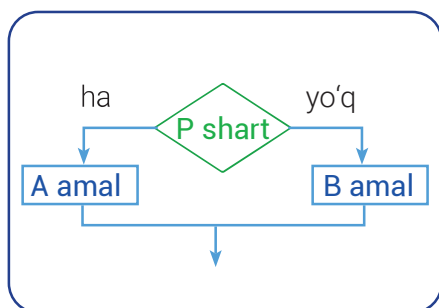
Algoritmning uchta asosiy konstruksiyasi (ko'rinishi) mavjud bo'lib, avvalgi darsda chiziqli algoritm konstruksiyasi bilan tanishdik. Endi esa tarmoqlanuvchi algoritm konstruksiyasi va ularga doir dastur tuzishni ko'rib chiqamiz.

Hayotda shunday jarayonlar ham mavjudki, unda kechadigan xatti-harakatlar muayyan shartlarga muvofiq bajarilishi lozim. Agar tashqarida yomg'ir yog'ayotgan bo'lsa, soyabon olish lozim, aks holda, soyabonga hojat bo'lmaydi. Sizning xatti-harakatingiz mantiqiy shartning bajarilishiga, ya'ni tashqarida yomg'ir yog'ayotganligiga qarab, 2 ta tarmoqqa (xatti-harakatga: soyabon olish va soyabon olmaslik) ajraladi va ulardan bittasigina bajariladi. Yuqoridagi kabi jarayonlar uchun algoritm tuzishda tarmoqlanuvchi strukturaga ega algoritmardan foydalaniladi.

TAYANCH TUSHUNCHALAR

Tarmoqlanuvchi struktura, odatda, qandaydir **mantiqiy shartni tekshirish blokini** o'z ichiga oladi. Mantiqiy shartni tekshirish natijasiga ko'ra, **tarmoq** deb ataluvchi u yoki bu amallar ketma-ketligi bajariladi.

Tarmoqlanuvchi algoritmlar – muayyan shartlarga muvofiq bajariladigan harakatlar ketma ketligi.

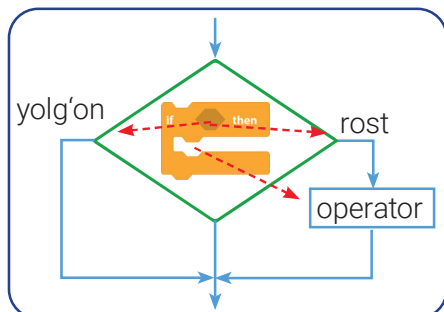


Berilgan shartning bajarilishiga qarab, tarmoqlanuvchi struktura ko'rsatilgan tarmoqdan faqat bitta amalning bajarilishini ta'minlaydi. Berilgan shart – *P* romb ichiga yoziladi. Dastur mantiqiy shartning bajarilish yoki bajarilmasligiga qarab tarmoqlanishi mumkin. Agar dasturda shart bajarilsa, "*ha*" tarmog'i bo'yicha *A amal*, aks holda, ya'ni shart bajarilmasa, "*yo'q*" tarmog'i bo'yicha *B amal* bajariladi.

Muayyan shartlarga muvofiq ayrim buyruqlar to'plamini amalga oshirish kerak bo'lganda, tarmoqlanish bloklaridan foydalaniladi. Aynan tarmoqlanuvchi jarayonlarga doir dasturlashda ma'lumotlarning mantiqiy toifalaridan foydalaniladi. Mantiqiy toifa ikki xil qiymatni qabul qilishi mumkin: "*ha*" (true/rost) va "*yo'q*" (false/yolg'on).

Scratch dasturining "Control" bo'limiga tegishli *if..then* va *if..then..else* bloklari bilan tanishib chiqamiz.

Shartli operatorning to'liqsiz shakli – if..then bloki



Shart natijasi “rost” qiymatni qabul qilsa, blokning ichida joylashgan operatorlar bajariladi, aks holda, dasturning bajarilishi shartli operator blokidan keyingi operatorlarga uzatiladi.

Bu turdagi bloklarga maxsus turdagi mantiqiy ifodalardan foydalanuvchi qo'shimcha blokni kiritish mumkin. Bunday bloklar yordamida mantiqiy ifodalar, qandaydir (masalan, “Sichqoncha tugmachasini bosdingizmi?”, “Tashqarida

yomg'ir yog'yaptimi?” yoki “ $15 * 2 = 36$ tenglik to'g'rimi?” kabi) savollar “so'raladi”. Berilgan savolga “yo'q, bilmayman” yoki “o'ylashim kerak” kabi javoblar berilishi mumkin emas. Savollarga faqat va faqat “ha” yoki “yo'q” tarzida javob berilishi lozim.

Shuningdek, blok ichida so'roq tarzidagi iboralardan foydalanilmasligi, balki shunchaki “rost” yoki “yolg'on”ligi tasdiqlanadigan ifoda (ibora)lar bo'lishi ham mumkin.

Masalan, “ $15 * 2 = 36$ ”, “ $16 + 3 < 25$ ”.

Mantiqiy ifodada ishtirok etadigan bloklar bitta bo'limga tegishli bo'lmasligi ham mumkin. Ularni “Sensing” va “Operators” bo'limlariga tegishli bloklar orasidan topish mumkin. Biz ushbu bo'limga tegishli ayrim bloklardan avvalgi darsimizda ham foydalangan edik.

Scratchda “Operators” bo'limida joylashgan mantiqiy amallarga oid quyidagi 6 ta blokdan foydalaniladi:



– chapdagi qiymat o'ngdagi qiymatdan kichik;



– chapdagi qiymat o'ngdagi qiymatdan katta;



– ikkalasi teng;



– mantiqiy ko'paytirish (konyunksiya) amali: bir vaqtning o'zida ikkala mantiqiy ifoda yoki o'zgaruvchining qiymati rost;



– mantiqiy qo'shish (dizyunksiya) amali: kamida bitta mantiqiy ifoda yoki o'zgaruvchining qiymati rost (ikkalasi ham rost bo'lishi mumkin);



– mantiqiy inkor (inversiya) amali: mantiqiy ifoda yoki o'zgaruvchining qiymati rost bo'lsa – yolg'on, yolg'on bo'lsa – rost.

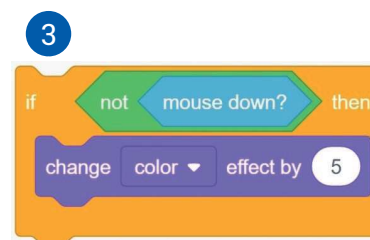
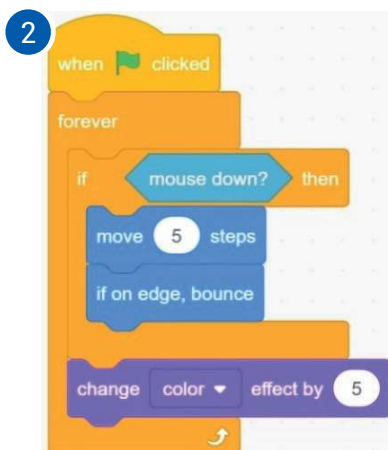
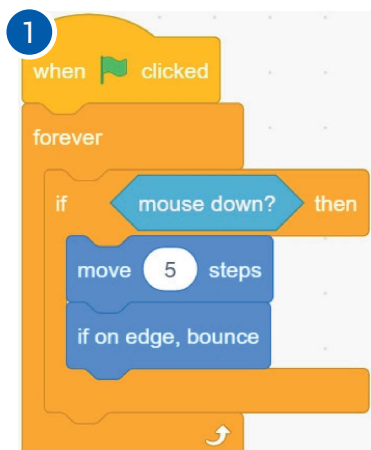
AMALIY MASHG'ULOT



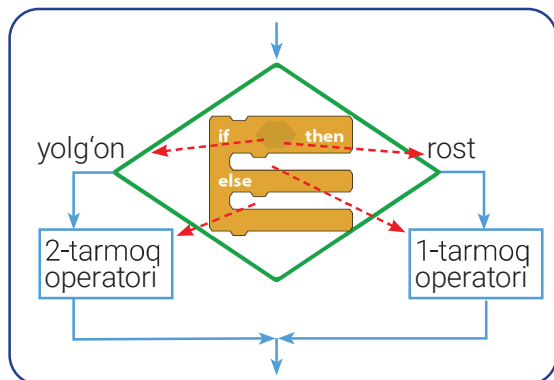
1-mashq. Sichqoncha tugmasi bosib turilsagina mushuk sahna bo'ylab harakatlansin. Agar u sahna chegarasiga urilsa, orqaga burilib, harakatda davom etsin. Mushukning rangi galma-gal o'zgarib turuvchi effekt qo'llanilsin.

Eslatma: loyihani amalga oshirish uchun "Control" bo'limida joylashgan "if..then" blokidan foydalaniladi.

1. Dasturni ishga tushiring.
2. Sichqoncha yordamida mushukni sahnaning quyi chap burchagiga olib o'ting va spraytlarni boshqarish panelining "Direction" bo'limidan – "Left/Right" xossasini faollashtiring.
3. Mushuk uchun 1-skriptni tering.
4. Dasturni ishga tushiring, sichqonchaning ixtiyoriy tugmachasini bosib turing va natijani kuzating.
5. Loyihani "Mushuk_1" nomi bilan saqlang.
6. Mushukning rangi galma-gal o'zgarib turishi uchun 1-skriptni 2-skript ko'rinishiga o'zgartiring.
7. Dasturni ishga tushiring va natijani kuzating. O'zgarishni tushuntirib bering.
8. 2-skriptdagi mantiqiy ifodani 3-skriptdagi mantiqiy ifodaga o'zgartiring. Dasturni ishga tushiring va natijani kuzating. O'zgarishni tushuntirib bering.
9. Loyihani "Mushuk_2" nomi bilan saqlang.



Shartli operatorning to'liq shakli – if..then..else bloki



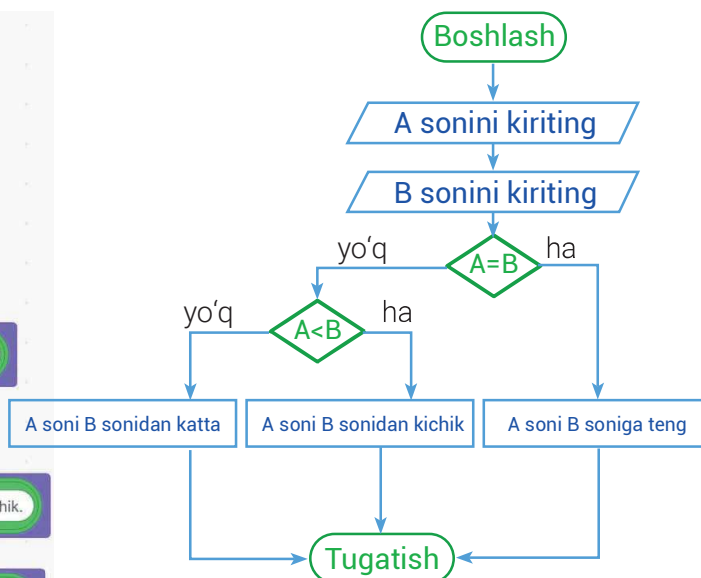
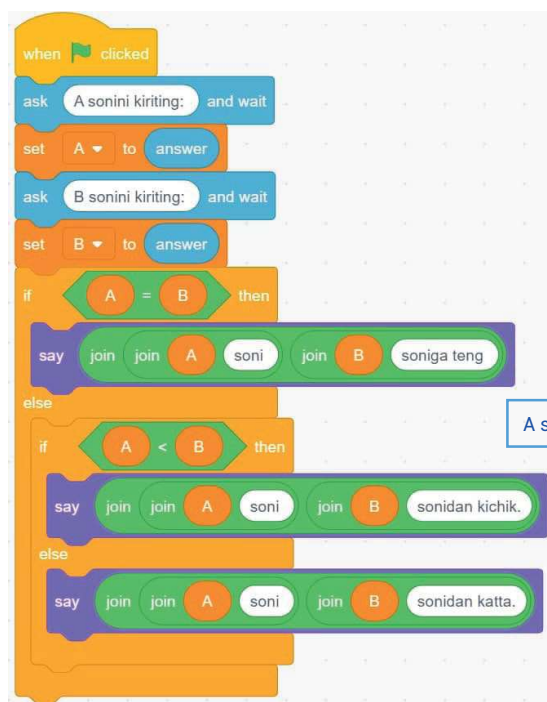
Agar shartning natijasi rost qiymatni qabul qilsa, *if* bo'limi ichidagi buyruqlar, ya'ni 1-tarmoq operatori bajariladi, aks holda, shartning natijasi yolg'on qiymatni qabul qilsa, *else* bo'limi ichidagi buyruqlar, ya'ni 2-tarmoq operatori bajariladi. Scratch dasturida shartli operatorlarni ichma-ich joylashtirish uchun hech qanday cheklov yo'q. Har qanday murakkablikdagi tarmoqlanishni ichma-ich joylashgan shartli operatorlar yordamida hal etish mumkin.

AMALIY MASHG'ULOT



2-mashq. Tarmoqlanuvchi if..then va if..then..else bloklaridan foydalanib, *A* va *B* sonlarini o'zaro taqqoslash dasturini tuzing.

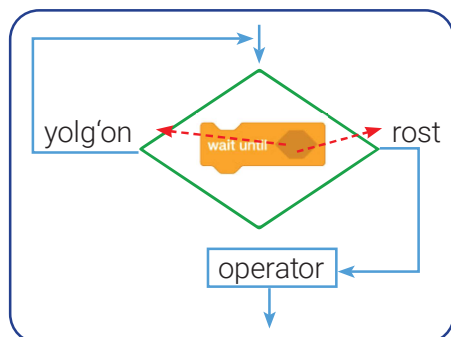
1. Dasturni ishga tushiring va *A* va *B* o'zgaruvchilarni yarating.
2. *A* va *B* sonlardan kattasini topish uchun quyidagi skriptni tering:



3. Dasturni ishga tushiring va natijani kuzating.
4. Loyihani "Taqqoslash" nomi bilan saqlang.



To'xtatish shartli operatori – wait..until bloki



wait..until bloki dasturda operator bajarilishini shartli ravishda to'xtatib turish uchun qo'llaniladi. Dasturning keyingi operatorlari mana shu blokda ko'rsatilgan shart bajarilganidan keyingina ishga tushadi. Bu blokda ham boshqa tarmoqlanish bloklari singari mantiqiy ifoda (ibora)larni kiritish uchun alohida joy mavjud. Bu blok davomiyligi sekundlar bilan emas, balki to'xtash *wait* 3 seconds (pauza) vazifasini bajaruvchi mantiqiy shartning bajarilishi bilan belgilanadi. Blok, uyga kirish uchun eshik qulfining kaliti kerak bo'lgani kabi, belgilangan shart

boshlanishidan oldin skriptning bajarilishini to'xtatib turadi.

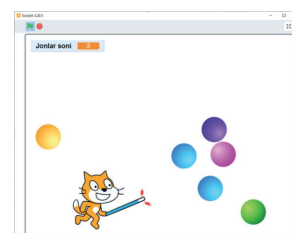
Blok algoritmda quyidagi holatlar mavjud bo'lganda foydalaniladi:

- ma'lum bir hodisani *wait until* touching edge ? kutishda;
- zarur o'zgaruvchi *wait until* baho = 5 yoki ro'yxat *wait until* costume number contains 2 ? qiymatini kutishda.

AMALIY MASHG'ULOT

3-mashq. Quyidagi ssenariyga asoslangan skript tuzing.

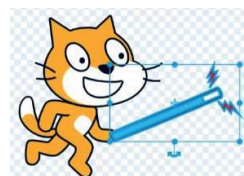
Bir necha rangli shar yuqoridan pastga tushmoqda. Mushuk yo'naltiruvchi tugmachalar (→, ←, ↑, ↓) orqali harakatlanadi. Mushuk jonlar sonini ko'paytirish uchun qo'lidagi qurol bilan sharlarni uradi. Agar sharlar mushukka tegib ketsa, uning jonlari soni kamayadi. Jonlar soni 0 (nol)ga teng bo'lganda, o'yin tugaydi.



1 Dasturni ishga tushiring va "Jonlar soni" deb nomlangan o'zgaruvchini yarating.

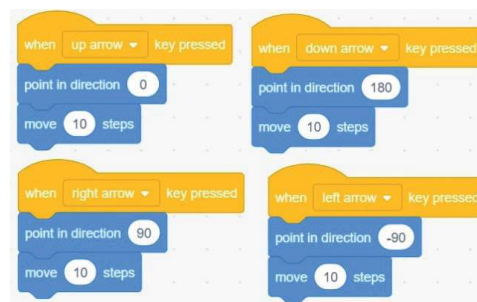
2 Sahnaga spraytlar kutubxonasidan "Magic wand" nomli spraytni joylashtiring. "Costumes" sahifasiga o'ting va sehrli tayoqcha uchlarini qizil rangga bo'yang.

Spraytni to'liq nusxalang (buning uchun dastlab "Ctrl + A", keyin "Ctrl + C"ni bosib) va uni mushuk qo'lga sehrli tayoqcha sifatida joylashtiring ("Ctrl + V"ni bosib) [sehrli tayoqcha nusxasini mushuk kostyumlariga (spraytga emas!) qo'shing]. Dasturda mushukning faqat bitta kostyumi ishlatiladi, shuning uchun sehrli tayoqcha 1-kostyumga o'rnatilsa kifoya.



3

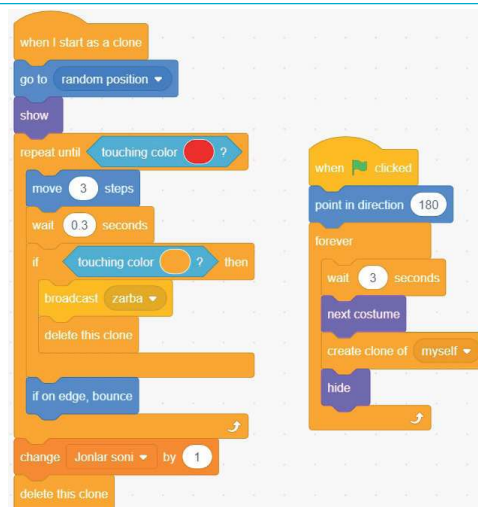
Mushukni yo'naltiruvchi tugmachalar orqali harakatlantirish uchun ushbu skriptlarni tering:



4

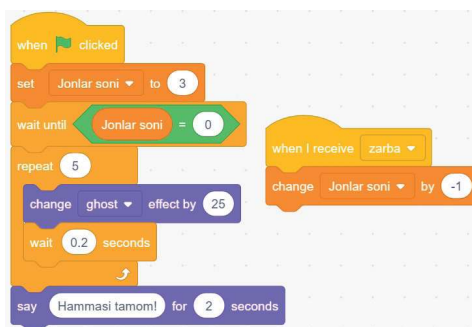
Rangli sharlar sahnada tasodifiy joyda paydo bo'lib, yuqoridan pastga qarab harakatlanadi. Bir nechta bir xil spraytni yaratib o'tirmaslik uchun klonlash qo'llaniladi. Kutubxonada bir nechta rang variantlariga ega "Ball" nomli shar ko'rinishidagi "Sprite" mavjud. Yangi klon yaratishda faqat uning kostyumigina o'zgaradi, xolos. Asl (original) sprayt yashiringan holatda bo'ladi.

Buni amalga oshirish uchun "Ball" sprayti uchun ushbu skriptlarni tering:



5

Mushuk sehrli tayoqchasining qizil uchi bilan sharlarga tegmasa, sharlar sahna chegarasiga urilib, yana ekranda qolaverishi kerak. Mushuk tayoqchasi bilan sharni ursa, jonlarining soni ko'payib boradi. Agar shar mushukka tegib ketsa, "zarba" degan xabar uzatiladi va jonlar soni kamayadi. Bularni amalga oshirish uchun mushuk spraytiga ushbu skriptlarni tering:



6

Dasturni ishga tushiring va natijani kuzating.

7

Loyihani "Sharchalar" nomi bilan saqlang.

SAVOL VA TOPSHIRIQLAR



1. Tarmoqlanish deganda nimani tushunasiz?
2. Tarmoqlanuvchi algoritmi deb nimaga aytiladi?
3. Tarmoqlanuvchi jarayonlarga hayotiy misollar keltiring.
4. Scratch dasturida qancha shartli operator mavjud?
5. Bitta shartli operator ichiga nechta shartli operatorni ichma-ich joylashtirish mumkin?
6. Mantiqiy ifodalarsiz shartli operatorlardan foydalanish mumkinmi? Nima uchun?