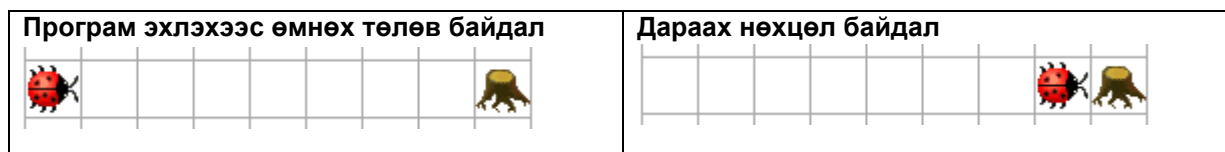


Програмчлалын үндсэн ойлголт: Давталттай ажиллагаа (Давталт)

Зогсох нөхцөлтэй давталт (while-давталт)

Кара модны өмнө очих хүртэл чигээрээ явна:



Нөхцөл биелэсэн бол заавар эсвэл зааврын блокыг while давталтын тусламжтайгаар давтан гүйцэтгэнэ:

```
public void myProgram() {  
    while (!kara.treeFront()) {  
        kara.move();  
    }  
}
```

!kara.treeFront()-ийн анхаарлын тэмдэг бол үгүйсгэл оператор юм. Энэ оператор нь kara.treeFront() хэллэгийг урвуулдаг. Хэрэв Кара-гийн өмнө мод байвал энэхүү илэрхийлэл нь *үнэн*-ний оронд худал болно. Үүнтэй адилаар, хэрэв Кара-гийн өмнө мод байхгүй бол, энэхүү илэрхийлэл нь *худал*-ийн оронд *үнэн* болно.

while давталтын бүтэц болон өгүүлбэр зүй

```
while (давталтын нөхцөл) {  
    заавар1  
    заавар2  
    заавар3  
}
```

Нөхцөл биелэсэн бол, өөр өөр зааврыг дараалуулан боловсруулдаг. Энд эхний зааврыг боловсруулахаас өмнө зааврын блокыг ер нь гүйцэтгэх ёстой эсэхийг үргэлж шалгадаг.

Давталтыг давхарлаж болно: while-давталт нь өөр нэг while-давталтыг агуулна, тэр нь өөр нэг while-давталтыг агуулж болно, гэх мэтчилэн.

Тоонд-суурилсан давталт (for-давталт)

Кара хоосон газарт дараалан байрласан 5 навчийг тавих хэрэгтэй:

Програм эхлэхээс өмнөх төлөв байдал	Дараах нөхцөл байдал
	

for давталтын тусламжтайгаар зааврын блокыг тодорхой тоогоор давтан гүйцэтгэж болно.

```
public void myProgram() {  
    for (inti=1; i<=5; i++) {  
        kara.putLeaf();  
        kara.move();  
    }  
}
```

for давталтын бүтэц болон өгүүлбэр зүй

```
for (Анхны утга олголт; Нөхцөл; Шинэчлэлт) {  
    заавар1  
    заавар2  
    заавар3  
}
```

Анхны утга олголт-**зааврыг** (Дээрх жишээнд `int i=1`) давталт эхлэхээс өмнө нэг удаа гүйцэтгэнэ. Ихэнх тохиолдолд үүнийг тоолуур хувьсагчид анхны утгыг олгохын тулд ашигладаг.

Нөхцөлийг (дээрх жишээнд `i<=5`) яг while давталт шиг эхний заавраас өмнө шалгадаг. Биелэхгүй бол давталтын гүйцэтгэлийг зогсоогоод, програмын үлдсэн үйл ажиллагааг үргэлжлүүлнэ.

Давталтын үйлдлийн төгсгөлд – нөхцөлийг шинээр шалгахаас өмнө – шинэчлэлт-**зааврыг** (дээрх жишээнд `i++`) гүйцэтгэнэ. Ихэнхдээ тоолуур хувьсагчийг өсгөхийн тулд ашигладаг.

Санамж: Анхны утга олголтын `int` энд хэрэгтэй, ингэснээр Java бидний тоолуур хувьсагч `i` нь бүхэл тоог илэрхийлж буйг мэднэ.

Шинэчлэлт-заавар `i++` нь тоолуур хувьсагчийг нэгээр нэмэгдүүлнэ. Өөрөөр бас `i=i+1` гэж бичиж болно. Үүнтэй адилаар `i--` гийн оронд `i=i-1` гэж бичиж болно.

JavaKara:

Давталтын даалгавар

1,2,3 –р даалгаврыг дагаж хийх.

4-өөс 9-р даалгавараас дурын 5 даалгаврыг гүйцэтгэж шалгуулах

1. Хонгил хайгч Кара I



Кара шулуун хонгилын (Нүд 2a) орох үүдийг хайна. Хонгилын доторх эхний нүдэнд түүнийг зогсоох програмыг бичнэ. Анхаарах зүйл: Зарим хонгил эхлээд нэг талдаа ханатай, заримдаа баруун талд, заримдаа зүүн талд.

2. Хонгил хайгч Кара II



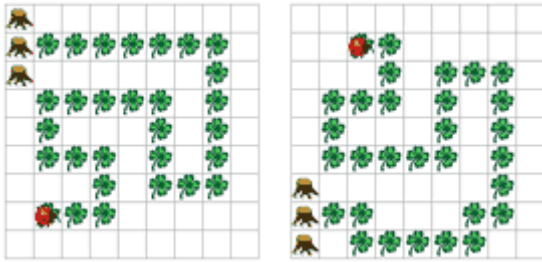
Кара хонгилын (Нүд 2b) гарах үүдийг хайж олно. Эхлээд хонгилоор явах ёстой. Хонгил дуусах эхний нүдэнд түүнийг зогсоох програмыг бич!

3. Ойд навч хайх II



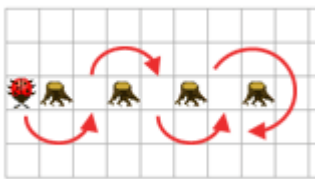
„Ойд навч хайх I“-ийг зэрэгцэн байрласан олон модыг тойрон гарч чаддаг болгож өөрчил!

Санамж: Энэ даалгаварыг шийдэхдээ давхар давталтыг хэрэглэх ёстой. Нэг давталтыг доод талын мөрд навчийг хайхад, өөр нэгийг дээд талын мөрд хайлтыг хийгээд, доод талын мөрд буцахад.



Кара навчин жимийг “иддэг” програмыг бич! Програм Кара модны өмнөх навч дээр зогсоход зогсоно. Эхлээд навч дээр эсвэл өмнө нь зогсож байж болно.

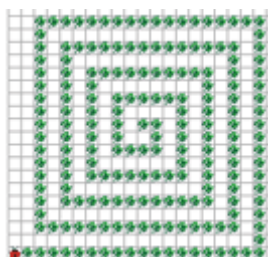
Санамж: Энэ даалгаварыг гүйцэтгэхэд амар биш. Програмчилж эхлэхээс өмнө шийдэх арга зам ямар байх талаар эхлээд нарийн сайн бодох хэрэгтэй!



Шүлшилтийг Кара хязгааргүй давтан хийхээр програмчил.

Санамж: Кара-гийн хөдөлгөөнийг мод тойрох „Дөрөвний нэг эргэлт“-ээр хуваавал хялбар болно.

6. Спирал зурах



Кара дээр харуулсан навчин спиралийг зурдаг програмыг бич. Дотроосоо гадагш спиралийн өнцөг бүр өмнөхөөсөө нэгээр урт байна.

7. Зургийг урвуулах

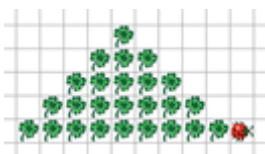


Кара модны доторх навчин зургийн “Эсрэг дүр”-ийг үүсгэдэг програмыг бич. Навч байвал авна, байхгүй бол тавина. Эхлэхдээ дандаа зүүн дээд өнцөгт баруун зүгт харсан байна.

Санамж: Хялбараар шийдэхийн тулд (`world.isLeaf(x,y)` болон `world.setLeaf(x,y,true)` өөрөөр `world.setLeaf(x,y,false)`) зааврыг хэрэглээрэй.

Санамж: Эхлээд даалгаварыг газрын зөвхөн нэг мөрийн хувьд шийдээрэй, жишээ нь $y=1$ (Навчтай эхний мөр). Энд нэг давталт хэрэгтэй. Дараа нь програмаа 2 дах давталтаар өргөтгөн, ингэснээр бүх мөрийг боловсруулна.

8. Гурвалжин зурах



Кара-г гурвалжин зурдаг болгож програмчил!

Санамж: Хялбараар шийдэхийн тулд (`world.isLeaf(x,y)` болон `world.setLeaf(x,y,true)` өөрөөр `world.setLeaf(x,y,false)`) зааврыг хэрэглээрэй.

Санамж: Тэгш өнцөгтийн шийдэлтэй төстэй. Гэхдээ ялгаа нь юу болохыг сайн бодоорой!

9. Төөрдөг байшин (хүнд)



Кара-г төөрдөг байшингаас гаргаж навч руу очуул.

Кара навчийг олоод, газраас авдаг програмыг бич. Энд Кара гарах үүдийн хажуугаар түүнийг ашиглахгүйгээр хэзээ ч өнгөрч явж болохгүй! Эхлэхдээ Кара дандаа баруун тийш харсан байна.