

```
totalhalt : first stop!  
  
one_way :  
turn : look to one side  
servo_turn : wait for the servo to be finished turning  
g 0, 0!  
totalhalt
```

```
the other way:  
turn : look to another side  
servo_turn : wait for the servo to be finished turning  
g 0, 0!  
totalhalt
```

```
so which is the better way?  
a2 : yes!  
b2 : turn  
  
b2 : return
```

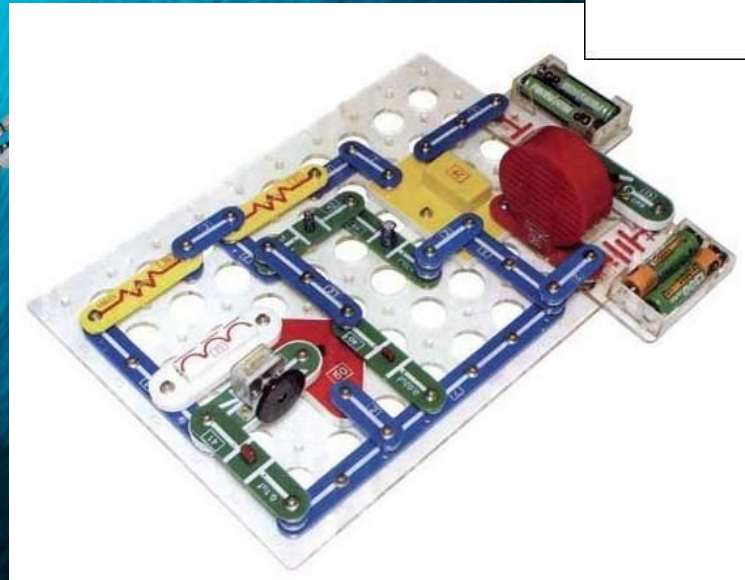
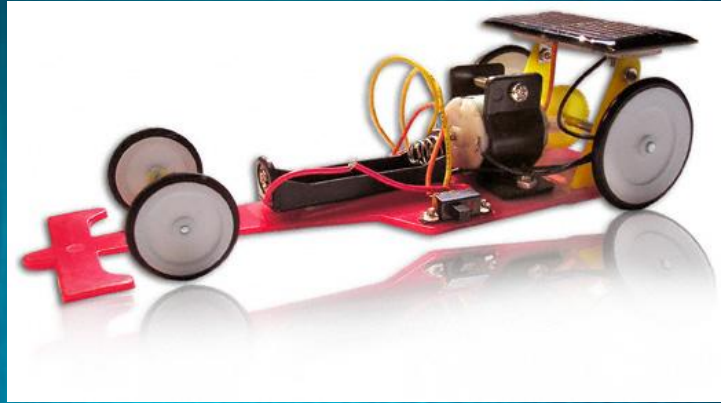
```
turns  
g low 5 : low 7 : high 4  
turn : servo : totalhalt
```

```
light  
g low 0 : low 2 : high 7  
turn : servo : totalhalt
```



Перворобот

Электронный конструктор



totalhalt : finish stop!
back : wait
turn : look to one side
servo : turn & wait for the servo to be finished turning
& go on!
totalhalt :

the other way!
turn : look to another side
servo : turn & wait for the servo to be finished turning
& go on!
totalhalt :

on which is the better way?
do them!
back : turn
back : turn

turn :
- low 5 : low 7 : high 4
turn : servo : totalhalt :

turn :
- low 6 : low 8 : high 7
turn : servo : totalhalt :

```
totalhalt = first.stopt  
brak = 0  
turn = look to one side  
servo.turn + wait for the servo to be finished turning  
g = 0  
totalhalt
```

```
the other way:  
turn = look to another side  
servo.turn + wait for the servo to be finished turning  
g = 0  
totalhalt
```

```
on which is the better way:  
on them  
back to turn  
back return
```

```
turns  
low 5 : low 7 : high 4  
turn + servo totalhalt
```

```
high  
low 6 : low 7 : high 7  
turn + servo totalhalt
```

Пластмассовый конструктор



Металлический конструктор



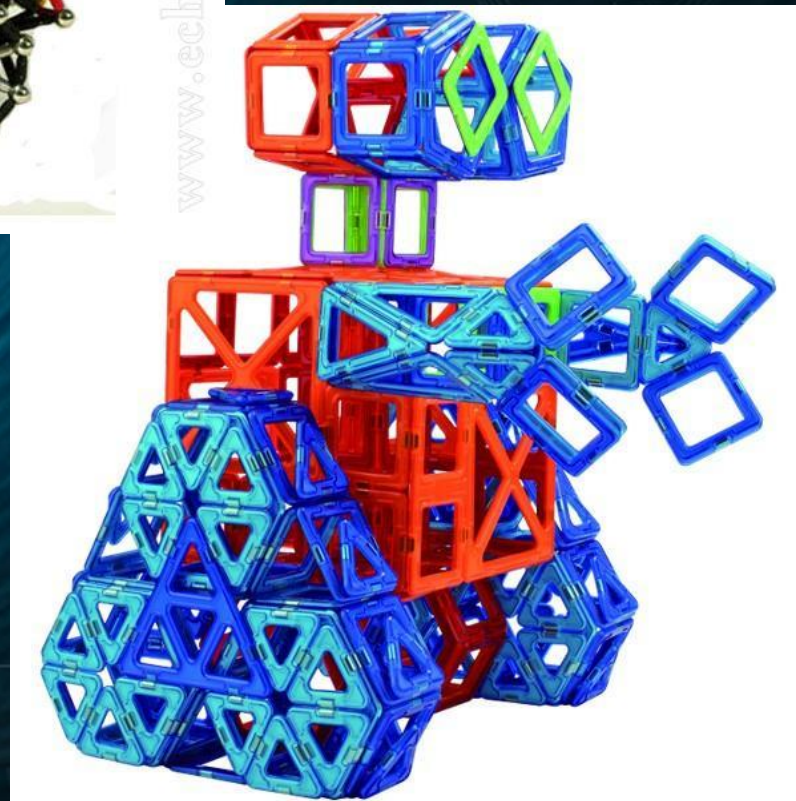
www.karusel-toys.ru



Магнитный конструктор



www.echobot.ru



```
totalhalt = first stop!  
  
low = 1  
high = 2  
servo.turn(1) wait for the servo to be finished turning  
g = 0  
totalhalt!  
  
the other way!  
return = look to another side  
servo.turn(1) wait for the servo to be finished turning  
g = 0  
totalhalt!  
  
as which is the better way?  
g = 0  
servo.turn  
  
low = 1  
high = 2  
turn(1) servo totalhalt!  
  
high = 2  
low = 1  
turn(1) servo totalhalt!
```

Деревянный конструктор

totalHeight = first.steel

base.moveTo

turn() look to one side

servo.turn() wait for the servo to be finished turning

g.by() at

totalHeight

the other way:

turn() look to another side

servo.turn() wait for the servo to be finished turning

g.by() at

totalHeight

as which is the better way:

at then

base.turn

base.turn

turns

low 5 : low 7 : high 4

turn() servo totalHeight

turn

low 6 : low 7 : high 7

turn() servo totalHeight

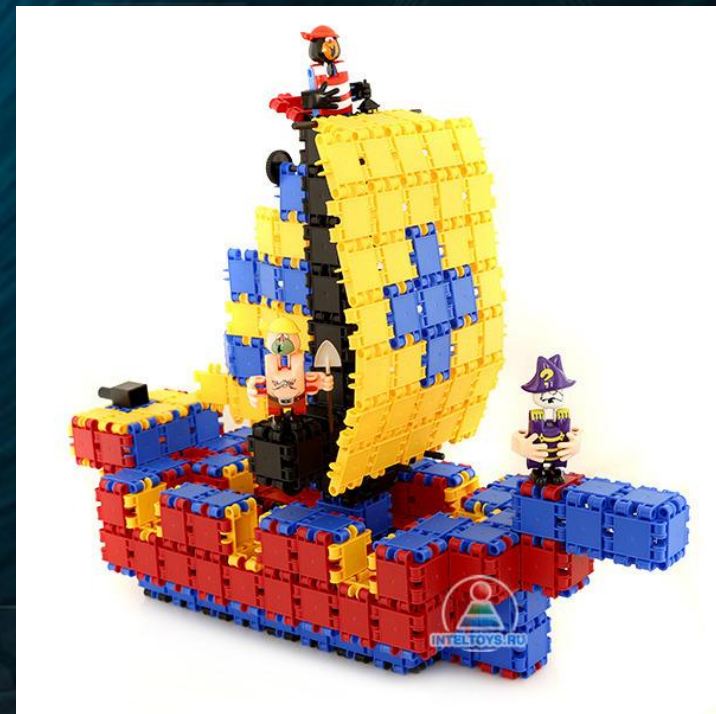


Конструктор из бумаги «Умная бумага»



Конструкторы из Бельгии

Кликс



Конструктор Lego



totalhalt : first stop
back : turn : look to one side
servo.turn : wait for the servo to be finished turning
a : b : c
totalhalt
the other way :
turn : look to another side
servo.turn : wait for the servo to be finished turning
a : b : c
totalhalt
as which is the better way :
a : b : c
totalhalt
back : turn
totalhalt
turn :
a : low 5 : low 7 : high 4
turn : servo : totalhalt
high :
a : low 6 : low 7 : high 7
turn : servo : totalhalt

Спасение самолета

```
totalhalt : first stop!  
  
back : wait  
turn : look to one side  
servo.turn : wait for the servo to be finished turning  
g : 90 deg  
totalhalt
```

```
the other way:  
turn : look to another side  
servo.turn : wait for the servo to be finished turning  
g : 90 deg  
totalhalt
```

```
so which is the better way?  
a2 : yes!  
back : turn
```

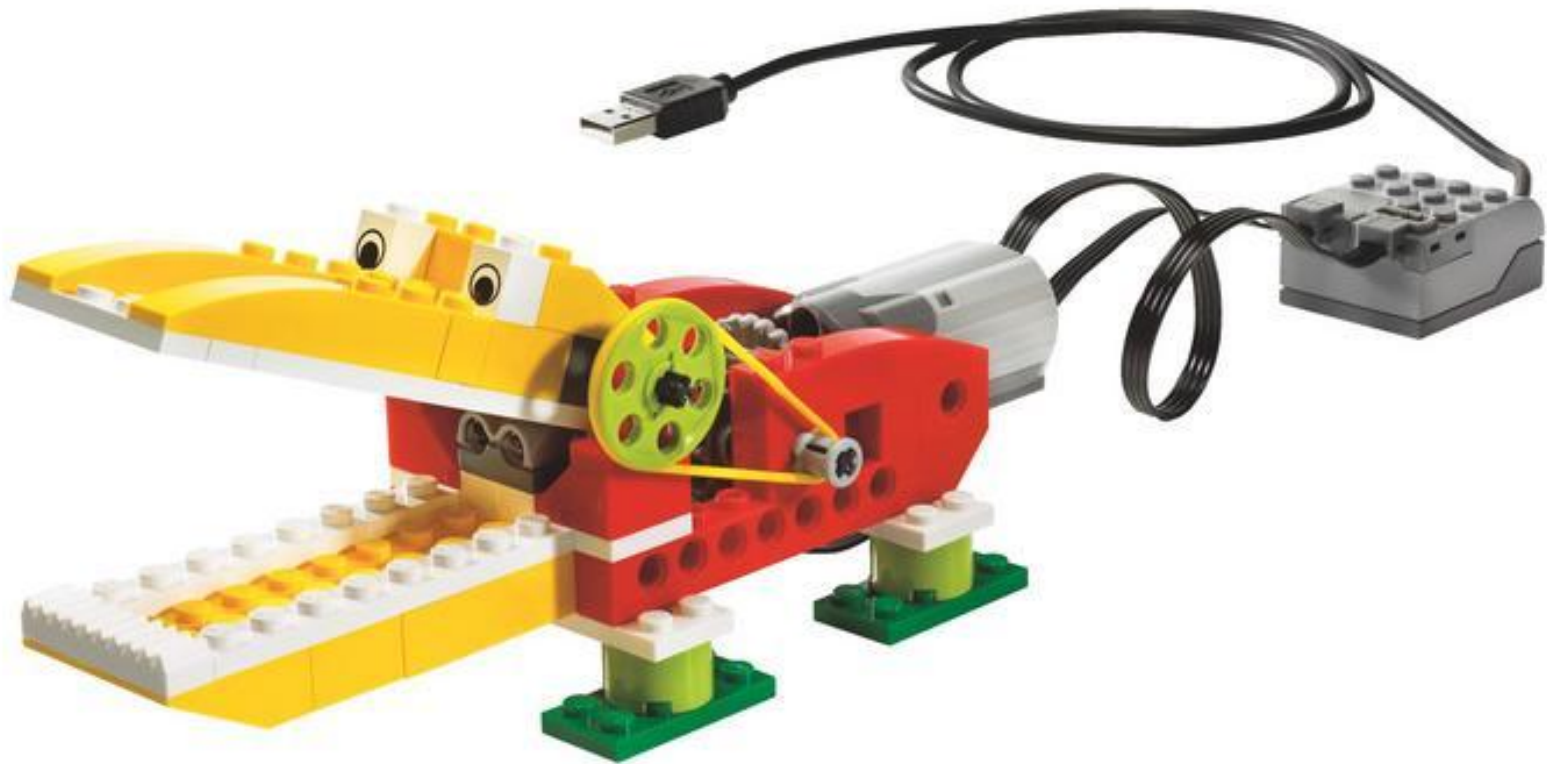
```
back : turn
```

```
turns  
g : low 5 : low 7 : high 4  
turn : servo totalhalt
```

```
light  
g : low 6 : low 4 : high 7  
turn : servo totalhalt
```



Голодный аллигатор



Рычащий лев



Порхающая птица



Вратарь



Танцующие ПТИЦЫ

```
totalHalt = 1; first servo;
brn;
turn; // look to one side
servo; turn; // wait for the servo to be finished turning
g; // go
totalHalt;
```

```
the other way:
turn; // look to another side
servo; turn; // wait for the servo to be finished turning
g; // go
totalHalt;
```

```
or which is the better way:
g; // go
servo; turn;
totalHalt;
```

```
turn;
g; low 5; // low 7; high 4
turn; servo; totalHalt;
```

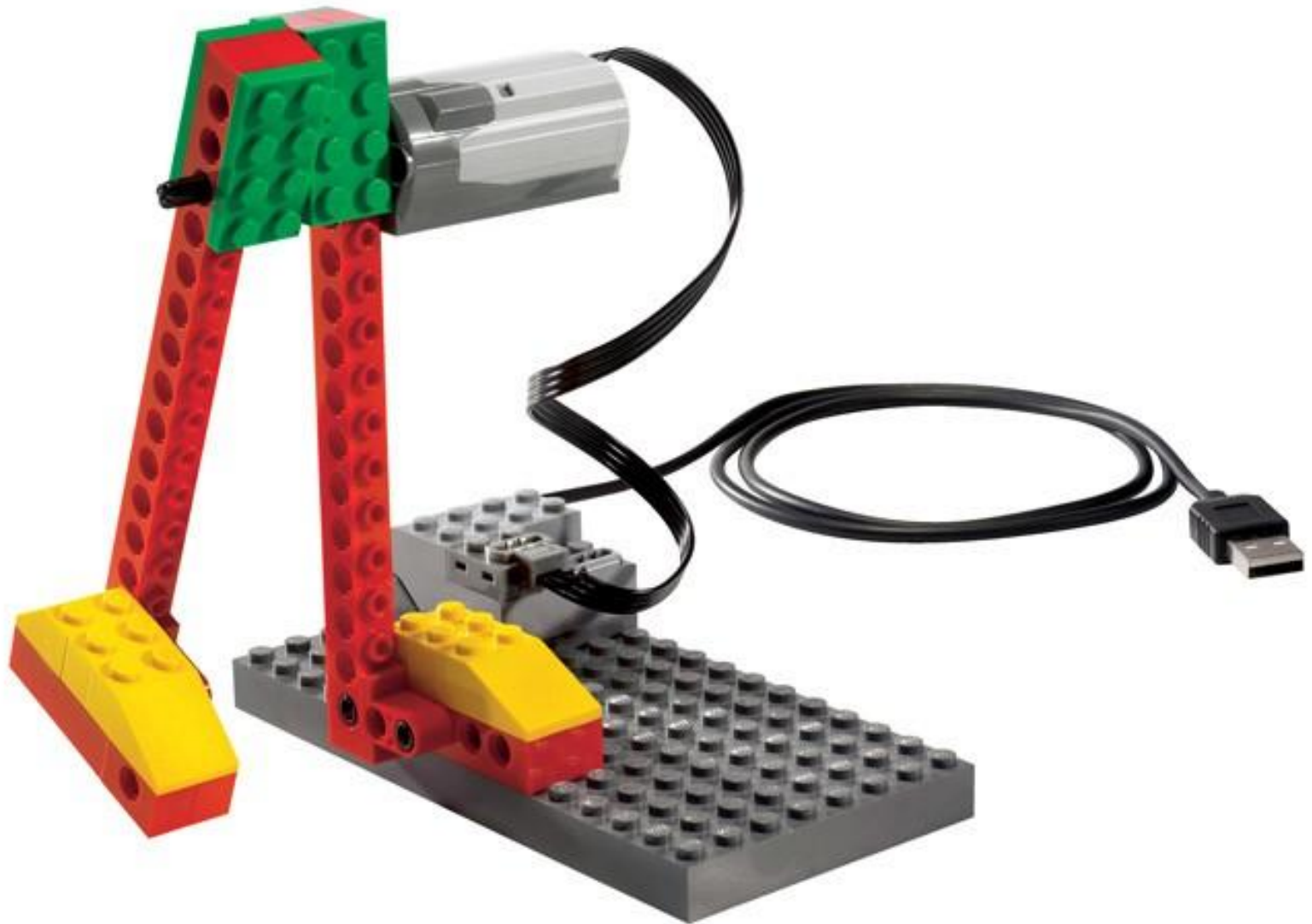
```
turn;
g; low 6; // low 4; high 7
turn; servo; totalHalt;
```



Обезьянка-барабанщица

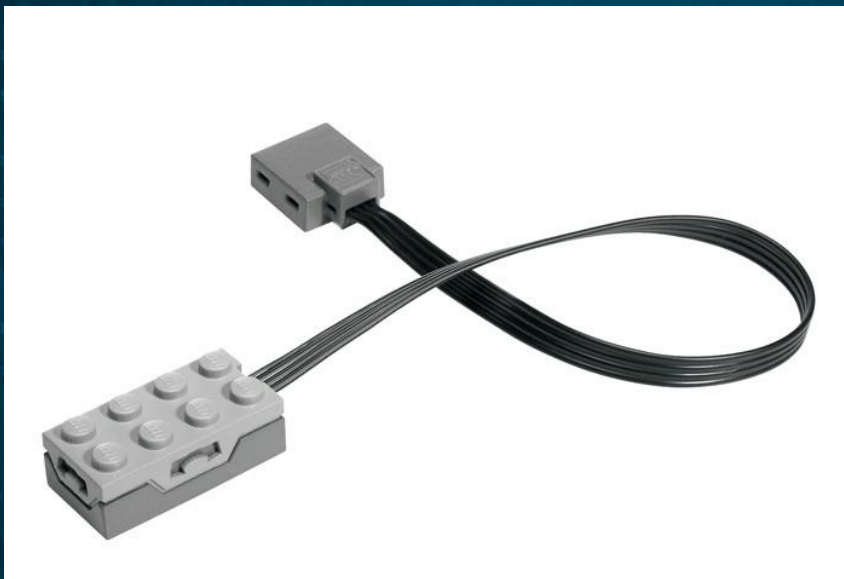


Нападающий



Конструктор Lego Wedo

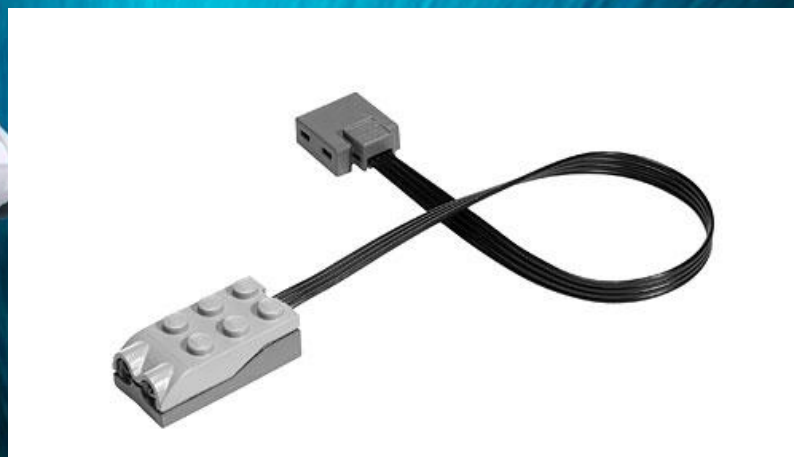




Датчик наклона



USB коммутатор



Датчик расстояния



Мотор



Конструктор NXT Mindstorm



totalhalt = 1; first stop;

```
break;
turn = 1; look to one side
servo.turn = 1; wait for the servo to be finished turning
g = 0; do
totalhalt;
```

```
the other way:
turn = 1; look to another side
servo.turn = 1; wait for the servo to be finished turning
g = 0; do
totalhalt;
```

```
do which is the better way:
do turn;
break; turn;
break; return;
```

```
turns
1 : low 5 : low 7 : high 4
turn = 1; servo; totalhalt;
```

```
high
1 : low 6 : 1; low 2 : high 7
turn = 1; servo; totalhalt;
```



```
totalhdlt : first stool
```

```
brk : hlt  
turn : look to one side  
servo_turn : wait for the servo to be finished turning  
g : 90  
totalhdlt
```

```
the other way:  
turn : look to another side  
servo_turn : wait for the servo to be finished turning  
g : 90  
totalhdlt
```

```
de : which is the better way?  
ac : hlt  
brk : turn  
brk : return
```

```
turns  
g : low 5 : low 7 : high 4  
turn : servo : totalhdlt
```

```
light  
g : low 6 : low 7 : high 7  
turn : servo : totalhdlt
```



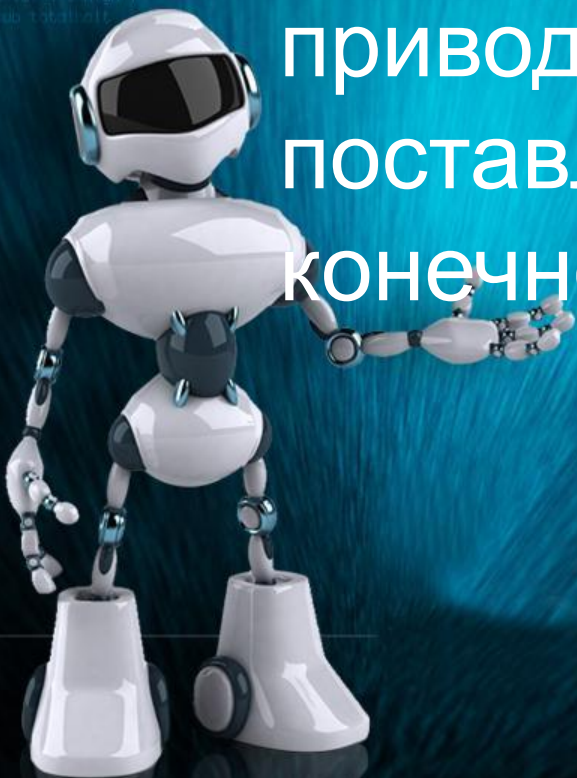
```
totalhalt = first_stop
loop:
  turn: look to one side
  servo_turn: wait for the servo to be finished turning
  g 0, 0
totalhalt
```

```
the other way:
turn: look to another side
servo_turn: wait for the servo to be finished turning
  g 0, 0
totalhalt
```

```
as which is the better way:
do: turn
servo_turn
end: return
```

```
turns:
  low 5 : low 7 : high 4
turn: servo totalhalt
```

```
turn:
  low 6 : low 4 : high 7
turn: servo totalhalt
```



Алгоритм

- это описание последовательности действий, строгое исполнение которых приводит к решению поставленной задачи за конечное число шагов.