

**LAPORAN SOURCE CODE ROBOMIND
PRAKTIKUM DASAR PEMROGRAMAN**



Disusun oleh:
ALFANSYAH PUTRA RAJA DINATA (L0223002)

**PROGRAM STUDI SAINS DATA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
DATA UNIVERSITAS SEBELAS MARET
2023**

SOURCE CODE ROBOMIND

1. LATIHAN 01

```
repeat
{
    if(frontIsWhite)
    {
        east()
        break
    }
    else if(frontIsBeacon)
    {
        eatup()
    }
    else
    {
        east()
    }
}
```

Untuk Pemograman di latihan 1, Saya memakai fungsi *repeat* untuk mempermudah eksekusi program yang saya buat.

Dari kode yang saya lampirkan diatas, bisa terlihat bahwa saya memakai fungsi if untuk kondisi keadaan, dan pemakaian else untuk menambah beberapa aturan yang dibuat.

Karena ada beacon yang menghalangi map di latihan ini, saya memilih untuk memakai fungsi “*eatup()*” karena lebih efisien.

2. LATIHAN 2

```
repeat{
    if(frontisWhite){
        forward
        right
    }
    else if(frontisBeacon){
        break
    }
    else{
        forward
    }
}
```

Disini saya memilih untuk menggerakkan robot maju kedepan karena di map ini tidak ada halangan yang membuat pergerakan robot menjadi sulit dan butuh kode yang rumit.

Karena titik putih menjadi acuan untuk belok, saya memakai fungsi "*frontisWhite*" untuk melakukan aksi belok.

Saat robot sampai tepat di beacon, saya memilih memakai fungsi "*break*" untuk menghentikan program.

3. LATIHAN 3

```
repeat{
    if(frontisWhite){
        forward
    }
    else if(leftisWhite){
        left
        forward
    }
    else if(rightisWhite){
        right
        forward
    }
    else if(frontisBeacon){
        break
    }

    else{
        east
    }
}
```

Karena diharuskan untuk mengikuti garis putih dan garis tersebut berada tepat di arah timur robot, saya menggunakan fungsi “*east*” agar robot secara default tetap ke arah timur.

Disaat robot berjalan, dia akan memindai bagian depan, kiri, dan kanan untuk mengecek apakah terdapat blok warna putih atau tidak, jika ada dia akan mengikuti garis, jika tidak dia akan bergerak terus ke timur.

Karena ada *beacon* sebagai titik akhir, saya memilih fungsi “*break*” untuk menghentikan program saat didepan robot terdapat *beacon*.

4. LATIHAN 4

```
repeat(3){  
  if(leftisBlack){  
    paintWhite  
    forward  
  }  
}  
stopPainting  
forward|
```

Untuk menghindari pemberian cat berlebih, saya memilih membatasi pengulangan sebanyak 3 kali dan mengakhiri nya di blok keempat agar hasil panjang garis putih sesuai dengan panjang garis hitam di samping kiri.

Agar dapat melihat jelas garis yang dibuat robot, saya memajukan robot satu blok.

5. LATIHAN 5

```
repeat(){
  if(frontisWhite){
    forward()
    right()
  }
  else if(frontisBeacon){
    pickUp()
    left(2)
    break
  }
  else{
    forward()
  }
}

repeat(){
  if(frontisWhite){
    forward()
    left()
  }
  else if(frontisObstacle){
    backward()
    putDown()
    break
  }
  else {
    forward()
  }
}
```

Untuk tugas terakhir, deskripsi yang saya sampaikan sama dengan ‘Latihan 2’, dengan tambahan beacon harus di pickup dan ditaruh di posisi selatan titik awal. Saya memilih untuk menambah fungsi kedua setelah pengambilan *beacon* dan memutar balik robot karena jika digabung, akan menimbulkan malfungsi pada program. Untuk menaruh *beacon* di posisi yang ditunjuk, saya harus membuat robot mendeteksi halangan tembok dulu, lalu mundur selangkah dan bisa menaruh *beacon* ke posisi yang dituju.