

# >> NAO ROBOCUP



**RFT TUKE**

# >> MAINMENU();



## OBSAH PREZENTÁCIE :

Pohyby

Rozpoznávanie obrazu

Stratégia strelenia gólu

Stratégia brankára

Komunikácia medzi hráčmi

# RFT TUKE

# >> PLAYMOTION();



**DIZAJN**

**POHYBOV :**

**.CHOREOGRAPHIE**

**.WEBOTS MOTION EDITOR**

**RFT TUKE**

# >> PLAYMOTION();



## POHYBY VSTÁVANIA :

- .VSTÁVANIE Z POZÍCIE DOLE BRUCHOM
- .VSTÁVANIE Z POZÍCIE NA CHRBTĚ

# >> PLAYMOTION();

**DETEKCIA PÁDU :**

- ULTRASOUND SENSORS
- BACK
- FRONT
- ACCELEROMETER
- TOUCH SENSORS

# >> PLAYMOTION();

## POHYB VPRED, OTÁČANIE A STREĽBA:

- .SHOOT MOTION
- .TURN LEFT / RIGHT
- .FORWARDS SLOW / FAST

# >> GET.IMAGE();

# ROZPOZNÁVANIE OBRAZU:

**.LOPTA**

**.BRÁNA**

[illegible]

# >> GET.IMAGE();

// normalize rgb components to be less dependant of lighting  
**REPREZENTÁCIA OBRAZU A**

## **PREDSPRACOVANIE:**

```
static void rgbnorm(const unsigned char in[3], double out[3]) {
```

```
    double max = in[0];
```

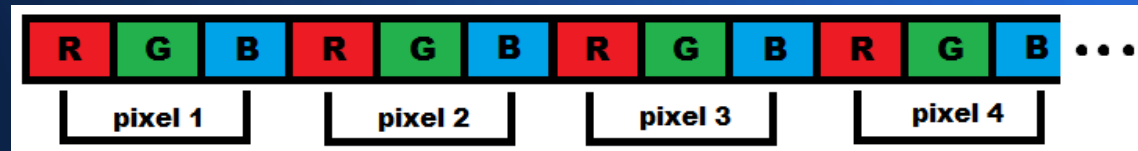
```
    if (in[1] > max) max = in[1];
```

```
    if (in[2] > max) max = in[2];
```

```
    out[0] = in[0] / max;
```

```
    out[1] = in[1] / max;
```

```
    out[2] = in[2] / max;
```





# >> GET.IMAGE();

## DETEKCIA LOPTY:

```
CONST DOUBLE BALL_COLOR[3] = { 1.0, 0.6, 0.2 };
```

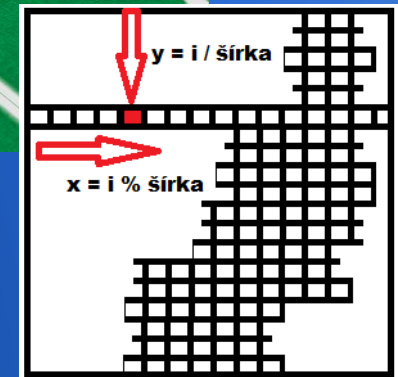
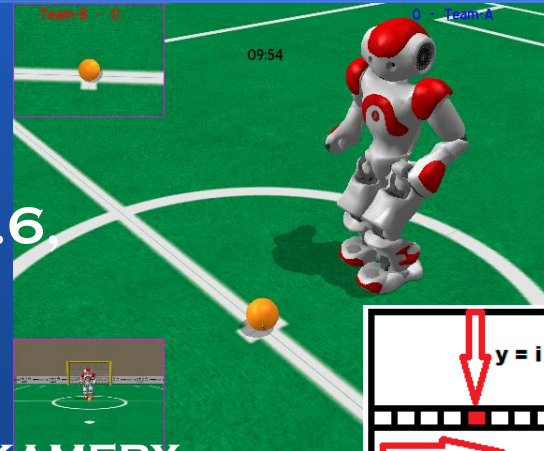
```
.SMER = ( PRIEMER(X) / ŠÍRKA - 0.5) * ZORNÉ POLE KAMERY
```

```
.ELEVÁCIA = - ( PRIEMER(Y) / VÝŠKA - 0.5) * ZORNÉ POLE KAMERY
```

```
.WB_SERVO_SET_POSITION(HEAD_YAW, WB_SERVO_GET_POSITION(HEAD_YAW)-  
OBJECTDIRECTIONANGLE);
```

```
.WB_SERVO_SET_POSITION(HEAD_PITCH, WB_SERVO_GET_POSITION(HEAD_PITCH)-  
OBJECTDIRECTIONANGLE);
```

**RTF TUKE**



# >> GET.IMAGE();

## DETEKCIA BRÁNKY:

- .BLÍZKA BRÁNA
- .VZDIALENÁ BRÁNA



# RFT TUKE

# >> GET.IMAGE();

## VZDIALENÁ BRÁNA:



- VÝHODA — VEĽMI PRESNÉ NAMERANÉ ÚDAJE

- NEVÝHODA — JE NUTNÉ VIDIEŤ CELÚ BRÁNKU

RTF TUKE

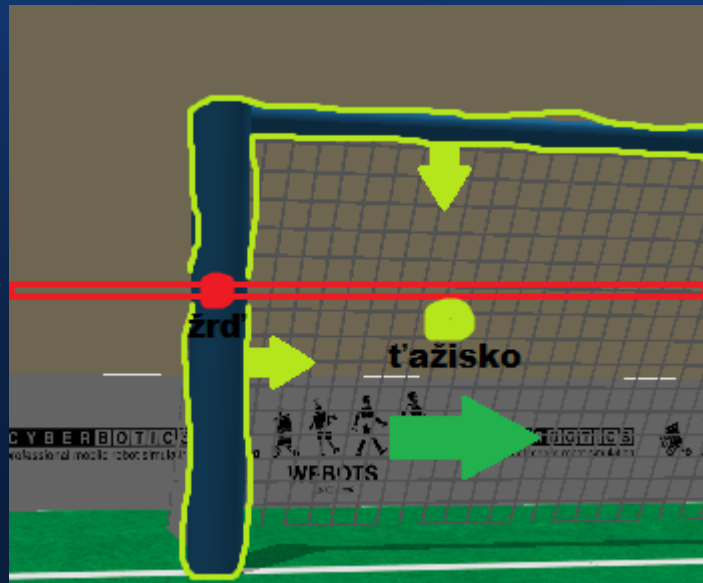
# >> GET.IMAGE();

**BLÍZKA BRÁNA:** .IF (POZICIA\_TAZISKA > POZICIA\_ZRDE)

POZICIA\_BRANY = ( POZICIA\_ZRDE /  
SIRKA\_OBRAZU - 0.5) \* FOV + 0.2;

.ELSE

POZICIA\_BRANY = ( POZICIA\_ZRDE /  
SIRKA\_OBRAZU - 0.5) \* FOV - 0.2;



**• NEVÝHODA — NEDOKÁŽE  
PRESNE URČIT'  
VZDIALENOSŤ  
RFT TUKE**

# >> STATE.ATTACKING();



## STRATÉGIA STRELENIA GÓLU:

**1.** DETEKCIA LOPTY A CESTA K  
LOPTE

**2.** DETEKCIA SÚPEROVEJ  
BRÁNKY

**3.** PRÍPRAVA NA STREĽBU

**4.** SAMOTNÁ STREĽBA  
RTF TUKE

**>> STATE.ATTACKING();**



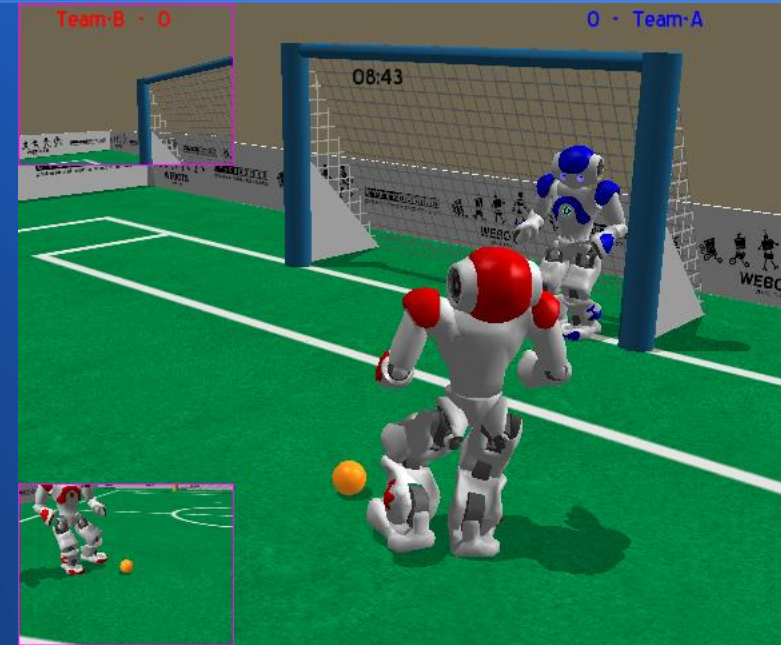
## **DETEKCIA LOPTY A CESTA K LOPTE:**

- .TURN LEFT**
- .TURN RIGHT**
- .FORWARDS SLOW / FAST**

# >> STATE.ATTACKING();

## DETEKCIA SÚPEROVEJ BRÁNKY:

- BLÍZKA BRÁNA
- VZDIALENÁ BRÁNA



## SLEDOVANIE BRÁNY + OTÁČANIE

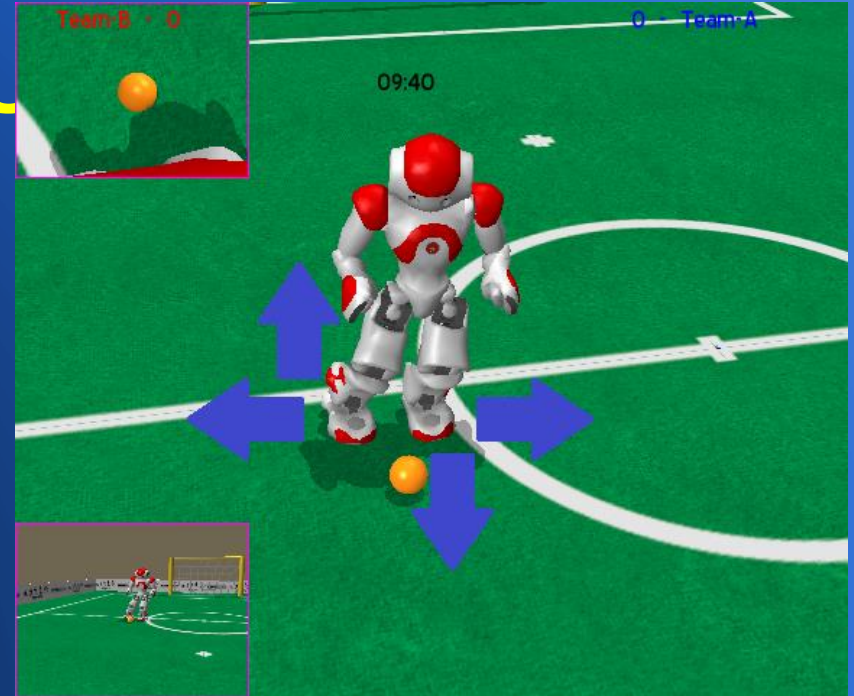
# RTF TUKE



# >> STATE.ATTACKING();

## PRÍPRAVA NA STREĽBU

- SLEDOVANIE LOPTY
- “STEPOVANIE”



ROBOT → LOPTA → BRÁNKA

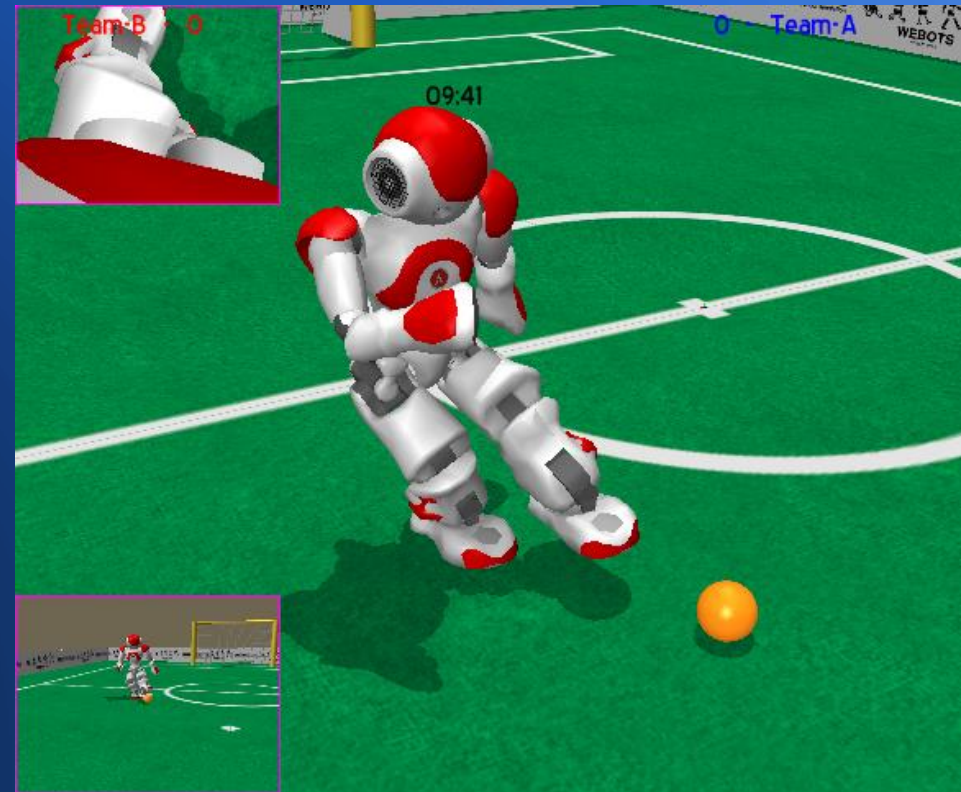
RFT TUKE



# >> STATE.ATTACKING();

## STREĽBA:

- ROBUSTNÝ POHYB
- PRESNOSŤ PRÍPRAVY
- DÁVKA ŠŤASTIA :)

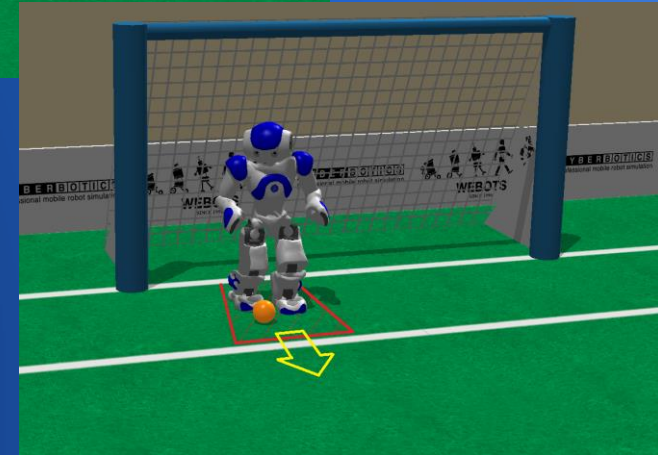
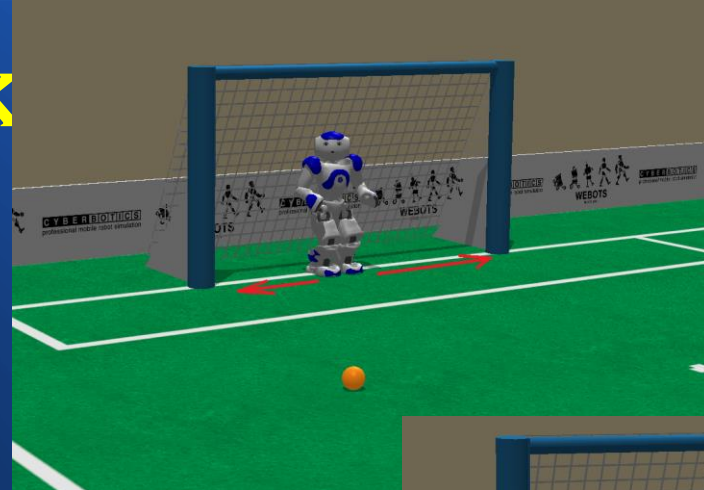


# RFT TUKE

# >> STATE.DEFENDING();

## STRATÉGIA BRANK

- POHYB NA ČIARE
- REAKCIA NA LOPTU
- ODKOPNUTIE LOPTY



# >> SEND.INFOMESSAGE();



## KOMUNIKÁCIA:

•EMITTER

•RECEIVER

•PAKETY

•TEAMOVÁ  
KOMUNIKÁCIA

•INFO O ZÁPASE

```
typedef struct {  
  
    char header  
    [VEĽKOSŤ_HLAVIČKY];  
  
    double ball_dist;  
  
    int attacking;  
  
    // etc  
} INFOMESSAGE;
```

RTF TUKE

**>> SYSTEM.EXIT();**

**THANK  
YOU.**



**RFT TUKE**