

# R.I.T.A.



---

**R**obot **I**nventor  
to **T**each **A**lgorithms

# Agenda

- ¿Qué es RITA?
- Conceptos Básicos
- ¿Qué debo tener en cuenta para escribir una estrategia de combate?
- Desafío
  - Individual
  - Grupal

# ¿Qué es RITA?

- RITA es un programa que te permitirá crear un robot para ser puesto en combate frente a los robots de tus compañeros
- El robot tiene la forma de un **“TANQUE”** de guerra

**El “TANQUE” ganador será el primero que destruya al resto, para esto debe atacar y defenderse**

# Tu tanque en combate...



# Conceptos Básicos

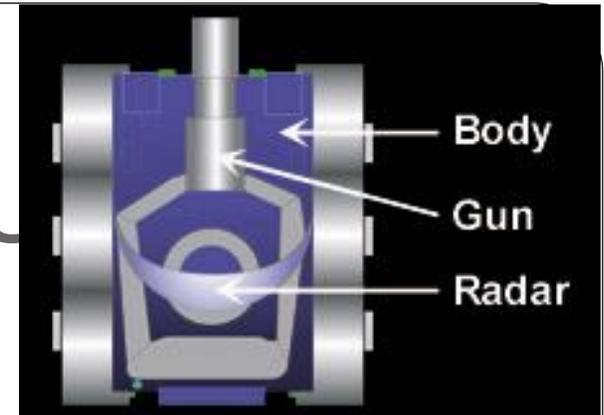
---

Composición de un TANQUE

Acciones y reacciones de un TANQUE

El campo de batalla

# Composición de un TANQUE



- **Body (Cuerpo):** Lleva encima el arma con el radar. Los movimientos que puede hacer el cuerpo son hacia adelante, hacia atrás, hacia la izquierda o derecha.
- **Gun (Arma):** Montada sobre el cuerpo, es usada para disparar balas. Los movimientos que puede hacer son hacia la izquierda o derecha.
- **Radar:** Montado sobre el arma, es usado para “escanear” otros robots mientras se mueve. El movimiento que puede realizar es hacia la izquierda o derecha. Genera “avisos o señales” cuando un robot es detectado.

# ¿Qué puede hacer un Tanque?

- **Se inicializa**
- **Busca** a otros tanques
- **Ataca** a otros tanques
- **Se defiende** de otros tanques

**¡Esto conformará tu estrategia!**  
**¡De esto depende tu supervivencia!**

# Un tanque “reacciona”...

Reacción...	En RITA
Al chocar contra otro tanque	onHitRobot
Al ser alcanzado por una bala enemiga	onHitByBullet
Al chocar contra un muro	onHitWall
Al escanear (encontrar) un tanque	onScannedRobot

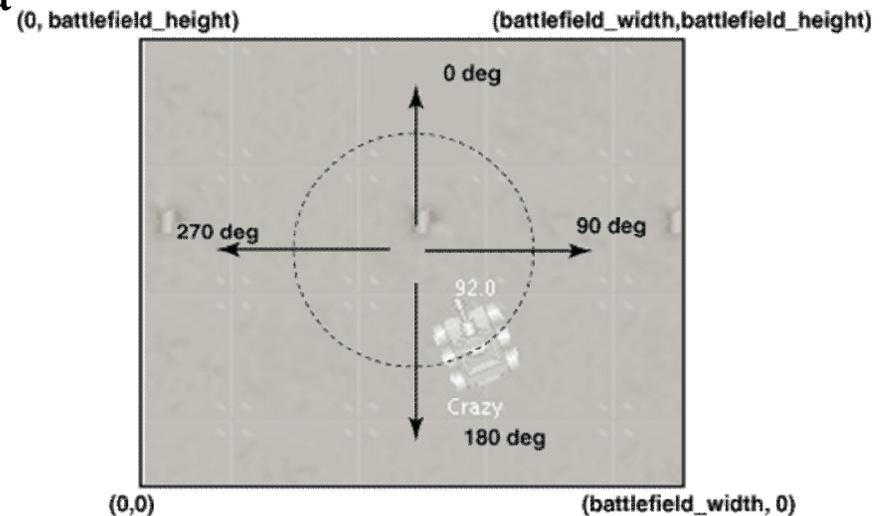
**Ante estas circunstancias nosotros indicaremos que acciones realizará nuestro tanque...**

# Acciones “que puede hacer” un Tanque

Acción	En RITA
Colorearse	setColor(colorCuerpo, colorArma, colorRadar)
Disparar con mayor o menor fuerza	fire(poderDeFuego)
Ir hacia adelante una distancia indicada e ir rotando a derecha los grados	turnAheadRight(distancia, grados)
Gira el arma una determinada cantidad de grados	turnGunTo(grados)

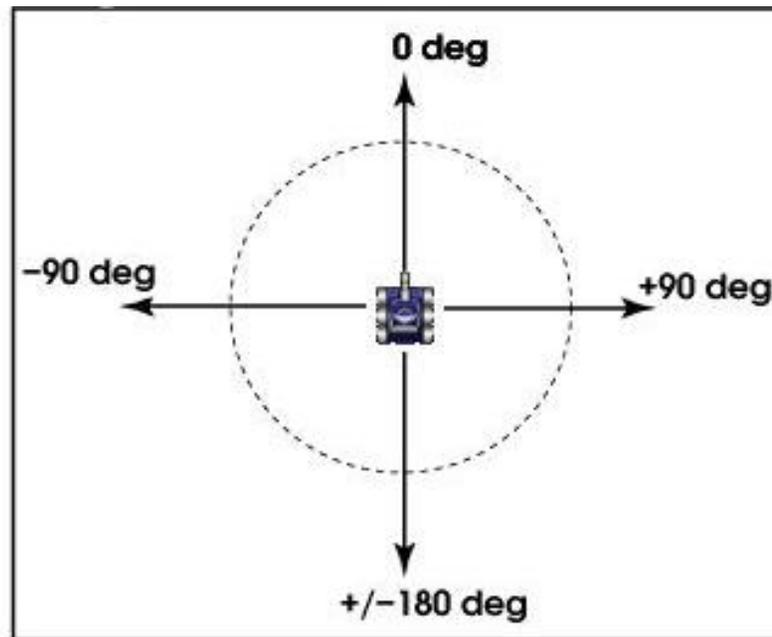
# Coordenadas y convenciones de dirección

- El campo de batalla usa el sistema de coordenadas cartesianas
- Dirección según las agujas del reloj:
  - 0 / 360 grados → hacia el "Norte"
  - 90 grados → hacia al "Este"
  - 180 grados → hacia el "Sur"
  - 270 grados → hacia el "Oeste"



# Bearing (ángulo relativo al cuerpo del robot)

- Rango de grado, va desde los -180 a 180 grados
- Es relativa a la posición en grados a la orientación de nuestro robot



# Ejemplo

Considerando la siguiente imagen ...



- ¿Qué se espera ante la ejecución de **turnGunTo(45)**?
- ¿Qué se espera ante la ejecución de **bearGunTo(45)**?

# ¿Qué debo tener en cuenta para escribir una estrategia de combate?

---

Energía

Balas y Colisiones

Algunas ideas para la estrategia

# Energía de un tanque

- Nuestro tanque empieza con una cierta cantidad de energía (100)
- Las acciones que realiza hacen que pierda o gane energía

**¡Si el tanque se queda sin energía, queda fuera de combate!**

# Pérdida de energía

Nuestro tanque **pierde energía** cuando...

- Realiza un disparo
- Choca contra un robot
- Choca contra los muros
- Recibe un impacto:
- Utiliza intensivamente el radar
- No hace nada (tiempo de inactividad)



¡Esta es la  
mayor  
penalización  
que ocurre!

# Incremento de energía

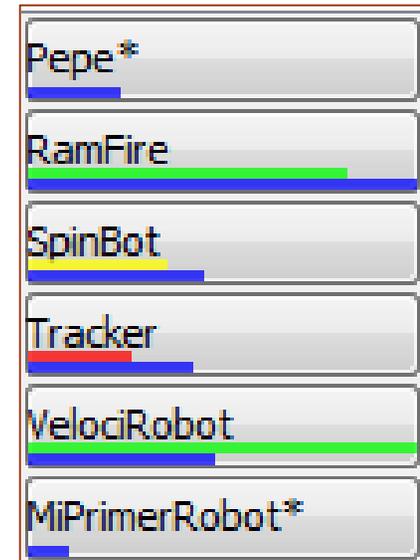
Nuestro robot aumenta su energía cuando...

- Acierta con sus balas sobre un tanque enemigo.

**¡Los tanques que se dedican sólo a disparar, quedan en estado “agotado”, por ende son deshabilitados!**

# ¿Como me va en la batalla?

- 2 barras por robot:
  - La primera indica **la energía que me queda**:
    - **>50**
    - **20-50**
    - **<20**
  - La segunda, en **azul**, indica el puntaje que va ganando en el round



# Balas y Colisiones

- Disparar genera **calor** en el arma. Un arma caliente no dispara, hay que esperar que enfríe.
- Si nuestro tanque **colisiona** (choca) a otro, será detenido momentáneamente. Por el contrario, si se estaba alejando del punto donde ocurrió la colisión, no será detenido.

# Algunas ideas para la estrategia

- El movimiento del tanque debería ser un tanto “errático” o cambiar entre distintos tipos:
  - Lineal: hacia adelante/atrás
  - Circular: avanzar y girar algunos grados siempre en la misma dirección (derecha o izquierda)
  - Oscilatorio: avanzar y girar turnando a izquierda y a derecha
- Alejar nuestro tanque de las zonas donde detectamos enemigos
- Mantener girando el radar de modo de escanear la mayor cantidad de tanques. El radar gira cuando el arma gira directa o indirectamente.

# Algunas ideas para la estrategia

- Al detectar un tanque girar el arma manteniendo el cuerpo del tanque en posición de escape y no enfrente del enemigo
- Cambiar de dirección cuando ocurre una colisión, ya sea porque ocurrió el evento de “chocar contra algo” o porque se detecta que la velocidad de nuestro robot es 0

# ¿Qué nos brinda RITA para crear nuestro tanque?

Revisión e interacción con la herramienta



# R.I.T.A. - la Aplicación

Barra de menú

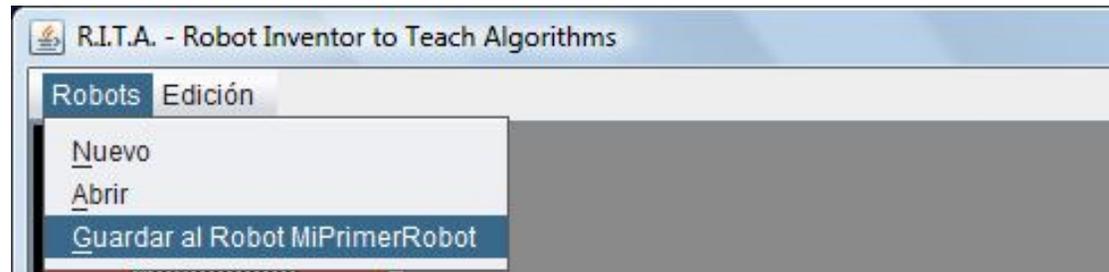
The screenshot shows the R.I.T.A. application interface. At the top is a menu bar. On the left is a vertical palette of blocks categorized into: Bloques Robocode (A Codificar, Movimiento, Acciones, Información, Colores), Sentencias Básicas (Texto, Lógica, Matemática), and Nuevas Definiciones (Método, Variable). The main workspace, titled 'junior: MIPrimerRobot', contains a sequence of code blocks: a 'run' block containing a 'setColors' block (with sub-blocks for bodyColor: orange, gunColor: blue, radarColor: white, bulletColor: yellow, scanAreaColor: black), a 'condition' block set to 'true', a 'do' block containing 'ahead' (steps: 100), 'turnGunRight' (degrees: 360), 'back' (steps: 100), and another 'turnGunRight' (degrees: 360), followed by an 'onScannedRobot' block (fire power: 1), an 'onHitByBullet' block (back steps: 10), and an 'onHitWall' block (back steps: 20). On the right side, there is a 'Minimapa' (minimap) and a 'Tacho' (trash can) icon. A 'Ver Código' button is also visible. The interface is annotated with several callouts: 'Barra de menú' points to the top menu bar; 'Bloques disponibles' points to the left block palette; 'Área de trabajo' points to the main workspace; 'Minimapa' points to the minimap; 'Probar nuestro tanque' points to the 'Ver Código' button; and 'Tacho' points to the trash can icon.

Todo esto conforma la estrategia de mi tanque

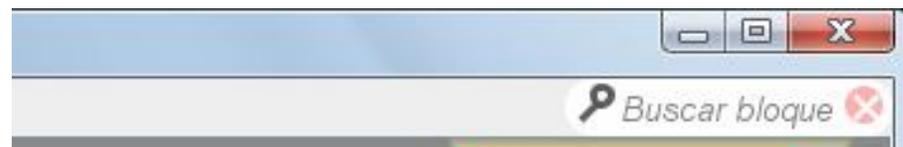
¡Las formas de los bloques indican si pueden o no encastrarse!

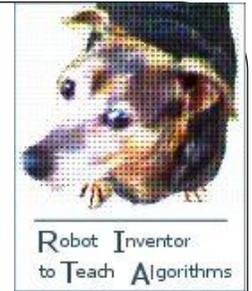
# Barra de menú

- A la izquierda un menú con opciones básicas para creación de robots y guardarlos
  - Permite crear un nuevo robot, guardarlo o abrir uno ya existente



- A la derecha la opción de Búsqueda





# Barra de menú

- Desde el menú Inicio de Windows, **ingrese a la aplicación R.I.T.A.**
- La aplicación le solicitará ingresar un nombre para su robot. **Ingrese un nombre.**
- Revisando el menú superior
  - Robots
  - Edición
  - Información
- En el submenú **Preferencias**, indique el nivel de su adversario como “**Medio**” y luego haga click en “**Guardar**”.

# Bloques disponibles



Información del tanque y de **qué puede hacer**.

Operaciones básicas como **cálculos o verificación** de que se cumplen o no determinadas condiciones

Un “método” será un **conjunto de acciones** que puedo realizar reiteradamente  
Las Variables me permitirán por ejemplo **guardar resultados de cálculos**, si es necesario



# Bloques disponibles - Robocod

- Características y acciones propias del robot.
  - **A Codificar:** representan las reacciones de nuestro robot. Un bloque del grupo “A Codificar” podrá incluir otros bloques de otros grupos.
    - ¿Cuántos bloques de este tipo encuentra en el Área de Trabajo?
    - ¿Qué contiene “onHitByBullet”?
  - **Movimiento:** representa como se desplaza y mueve el robot
    - ¿Qué hace **turnBackLeft**?
    - ¿Qué hace **turnRight**?



# Bloques disponibles - Robocode

- **Acciones:** representa qué es lo que puede hacer este robot
  - ¿Qué hace **turnGunLeft**?
  - ¿Qué hace **fire**?
- **Información:** datos de éste robot, otros robots y el campo de batalla
  - ¿A qué se refiere **energy**?
  - ¿A qué se refiere **hitByBulletAngle**?
  - ¿A qué se refiere **scannedAngle**?
- **Colores:** permiten indicar los colores de su robot (más o menos colores)

# Bloques disponibles: sentencia básicas



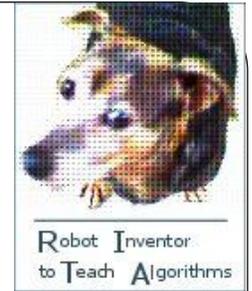
- Evaluación de condiciones, repetición de acciones, operaciones aritméticas
- ¿Cómo representaría: **“si el arma está lista entonces disparar”**?

# Bloques disponibles: Nuevas Definiciones



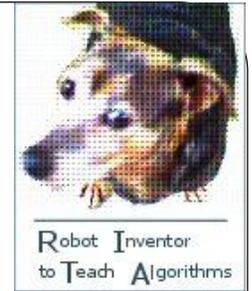
- **VARIABLE:** Sirve para definir variables para que almacene valores temporalmente.
  - Queremos representar la variable **pasos**
    - arrastre el bloque “**nombreDeVariable**”
    - arrastre el bloque que represente el valor “**1**”.
    - **Encastre** ambos bloques.
    - Haciendo **doble click** sobre “**nombreDeVariable**” podrá modificar el nombre de la variable.

# Bloques disponibles: Nuevas Definiciones



- Queremos incrementar en **10** a la variable **pasos**
  - Haga click con el botón derecho del mouse sobre la variable “**pasos**” definida antes, y seleccione “Cambiar valor de la variable”. Esta acción creará un bloque “**set pasos**”.
  - Entre los bloques de la izquierda, busque el bloque que corresponde a la operación matemática de la suma “+” y arrástrelo de modo que se encastre con el bloque “**set pasos**”.
  - **En este punto nos falta indicar cuáles son los valores a sumar.**
  - **Primer valor de la suma.** Haga click con el botón derecho del mouse nuevamente sobre la variable “pasos” y seleccione: “Tomar el valor de la variable”. Esta acción le crea un bloque el cual deberá ser arrastrado como primer valor a sumar.
  - **Segundo valor de la suma.** Busque entre los bloques disponibles de la izquierda el bloque que corresponda al valor “1”, arrástrelo y colóquelo como segundo valor de la suma. Haciendo doble click sobre el mismo puede darle otro valor, como 10.

# Bloques disponibles: Nuevas Definiciones



- **MÉTODO:** Permite agrupar bloques bajo un nombre común de modo de ejecutar ese grupo de bloques varias veces y desde distintos lugares en el Área de Trabajo.
- **Vamos a definir un método que realice primero un cálculo y luego una acción**
  - Arrastre hacia el área de trabajo el bloque “**miMetodo**”. Cambie el nombre haciendo doble click sobre el mismo. El método se llamará “**huir**”
  - El cuerpo del método deberá hacer lo siguiente:
    - Defina una variable que contenga el valor resultante de la siguiente cuenta: último ángulo escaneado – 45
    - A continuación realizar la siguiente acción: Ir hacia adelante 100 pasos y girar tanto grados como nos dio la cuenta anterior. Con esto, trataremos de ir en dirección perpendicular al

# Bloques disponibles: Nuevas Definiciones



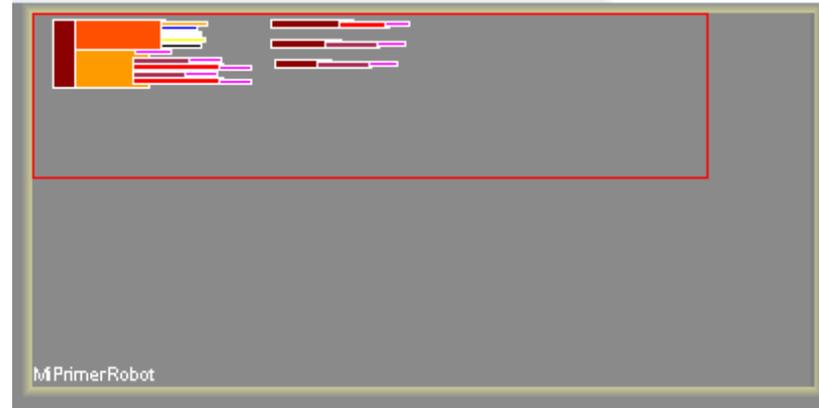
- Invocar a este método desde **“onScannedRobot”**. Es decir, cuando se escanee un robot enemigo realizaremos la acción de **“huir”**.

Para realizarlo:

- Haga click con el botón derecho del mouse y seleccione: **“Llamar a éste método”**. Esta acción creará un bloque, arrástrelo hasta la sección **“onScannedRobot”** y encástrelo al final de las acciones ya existentes.

# Complementos

- **Minimapa:**  
Visualizador de todo el código, permite acceso directo a un sector en particular
- **Tacho**
- Posee además una serie de alertas (para bloques y código)



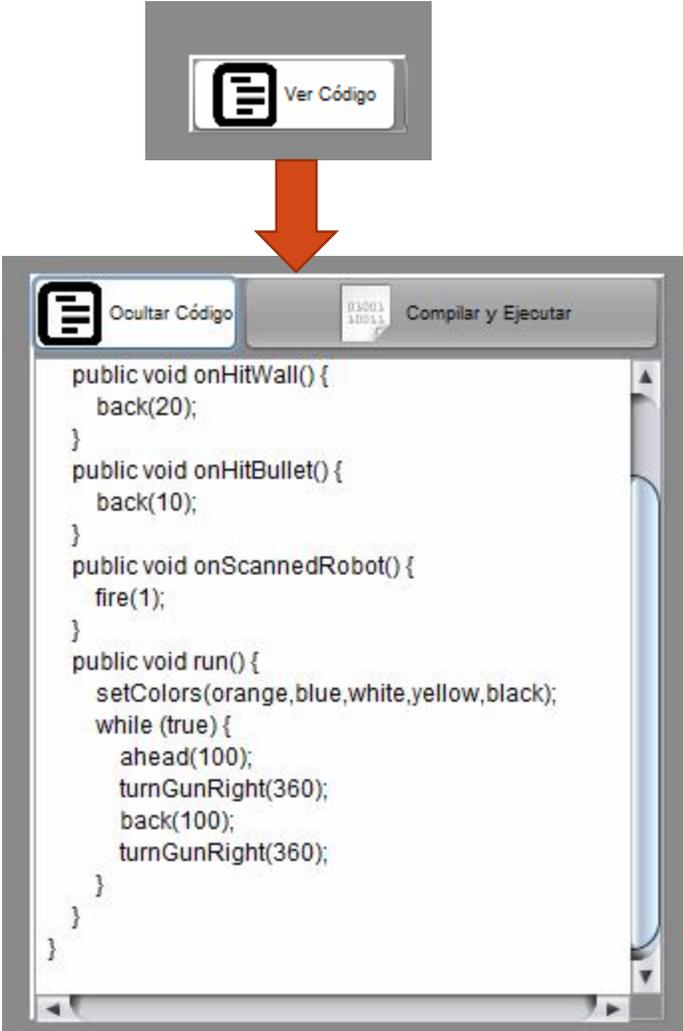


# Complementos

- Elimine del Área de Trabajo **SÓLO** la variable “pasos”. ¿Qué ocurrió?
- Elimine también los bloques “**rotos**”.

# Poner a prueba mi tanque

- La opción “Compilar y Ejecutar” pone a nuestro tanque en combate
- Además, el panel muestra nuestra estrategia escrita en código Java



```
public void onHitWall() {
    back(20);
}
public void onHitBullet() {
    back(10);
}
public void onScannedRobot() {
    fire(1);
}
public void run() {
    setColors(orange,blue,white,yellow,black);
    while (true) {
        ahead(100);
        turnGunRight(360);
        back(100);
        turnGunRight(360);
    }
}
```



# Poner a prueba mi tanque

- El Área de Trabajo contiene un botón "**Ver Código**". Para ejecutar nuestro robot hacer click en la opción "Compilar y ejecutar".
- Guarde su robot

# Desafío Grupal

---

Crear un robot para que se enfrente a los robots creados por el resto de sus compañeros