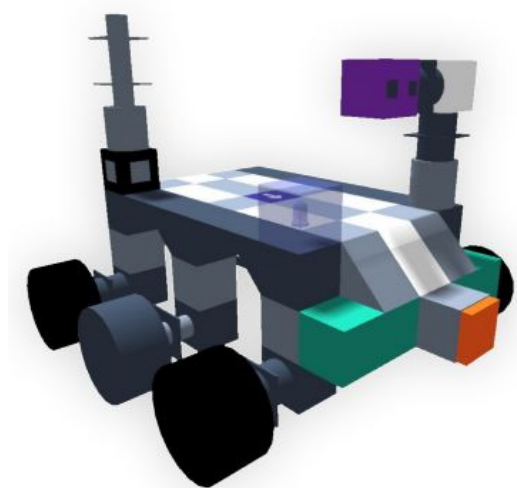




# MODALIDADE PRÁTICA: VIRTUAL RESGATE



## Manual de Regras e Instruções

Simulação / Nacional

Versão 1.0  
Setembro - 2024

Organização



Apoio



Patrocínio



# ÍNDICE

**Caderno Amarelo: A Olimpíada** (Pág. 3)



**Caderno Azul: A Plataforma** (Pág. 8)



**Caderno Verde: O Desafio** (Pág. 15)



**Caderno Vermelho: A competição** (Pág. 45)



**Caderno Cinza: Mudanças** (Pág. 59)



**Caderno Laranja: Conflitos** (Pág. 62)



Qualquer dúvida sobre a Modalidade Prática: Resgate Virtual, procure no FAQ da OBR  
[obr.robocup.org.br/perguntas-frequentes](http://obr.robocup.org.br/perguntas-frequentes)

*Caderno Amarelo*

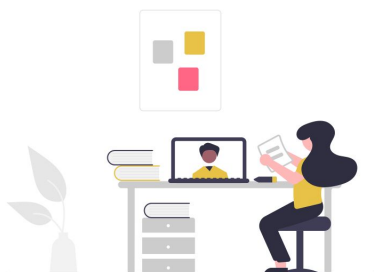
# A Olimpíada



**Descrição dos princípios, objetivos, código de conduta e a missão da Olimpíada Brasileira de Robótica**

# Princípios dos Competidores e Equipes

Alcançar seus objetivos **sem esperar que seu professor os alcance por você.**



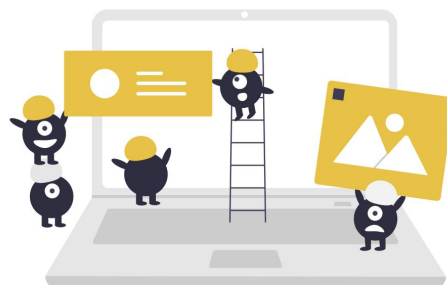
Superar os seus **limites** e os da sua equipe.



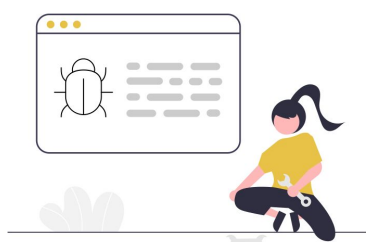
**Ajudar** seus colegas e adversários a superarem seus limites.



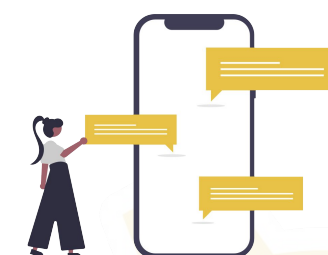
Ser um **bom competidor e amigo** de todos ao mesmo tempo.



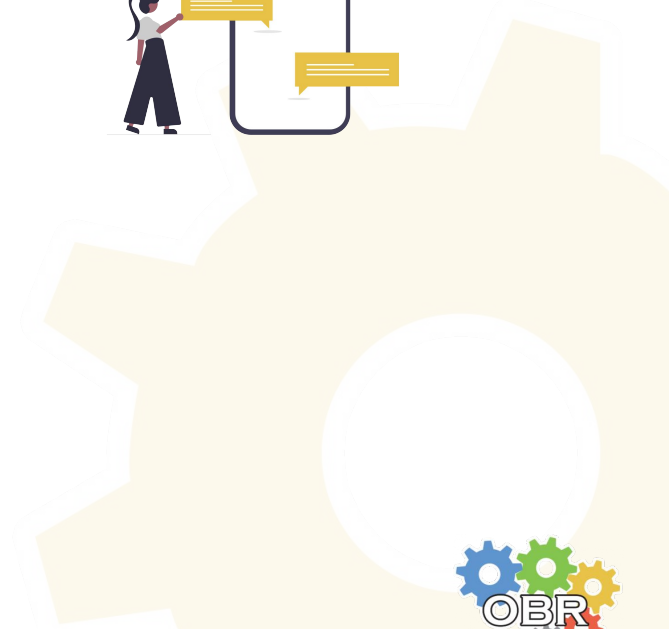
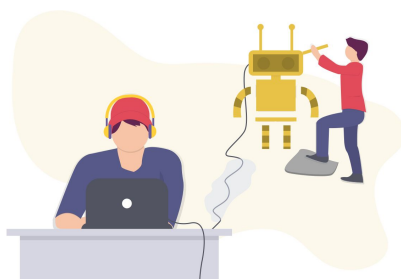
Saber que **mais importante** do que ganhar é conseguir **competir e aprender.**



**Ajudar sempre** a construir uma comunidade OBR maior e melhor.



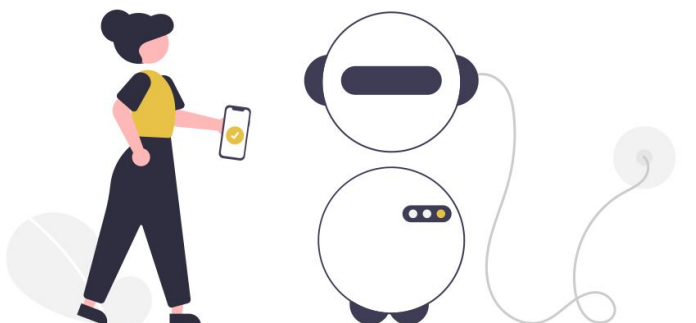
**Aprender** mais sobre a evolução da robótica no Brasil e no mundo.



## Missão da OBR

Espera-se que todos os participantes (estudantes e seus tutores) **respeitem a missão** da competição e da OBR de: *promover, incentivar e disseminar a robótica pelo Brasil.*

*A Robótica é uma área extremamente motivadora e que deve semear o desenvolvimento tecnológico no país e no Mundo, nos próximos anos.*



Proporcionar aos estudantes de hoje um contato com essa tecnologia pode retirá-los da condição de meros usuários de tecnologia e abrir a perspectiva de torná-los **desenvolvedores tecnológicos** nas próximas décadas, além de elevar o país e sua juventude a patamares de grandeza comparáveis aos demais países desenvolvidos do mundo.

Nosso maior desafio e objetivo é tornar nosso país um forte protagonista das transformações tecnológicas do futuro, capacitando nossos estudantes com a robótica desde seus primeiros anos de vida.

A competição e a OBR, portanto, **não devem** ser objeto de **promoção pessoal**, muito menos ser utilizada como mecanismo de **promoção de escolas**. Ela deve ser única e exclusivamente usada para a **promoção dos nossos estudantes a um futuro melhor.**

No final, não importa quem vai ganhar ou perder a competição, pois o Brasil e nossos estudantes já irão ter ganhado muito só por terem participado de forma intensa dessa que é a maior ação de disseminação da robótica em nível Nacional: A **Olimpíada Brasileira de Robótica!**



## Código de Conduta

Participe da competição de forma **limpa, saudável e ética**. Ajude seus colegas e outras equipes a superarem seus limites. **Divirta-se** durante toda a competição **e colabore** para que todos os demais participantes (juízes, plateia, professores, etc) **se divirtam também**.

*É esperado que todas as equipes estejam motivadas pelo espírito do “fair play” (jogo justo).*

*A organização fará todo o esforço para permitir um ambiente de competição saudável e cooperativo. Em alguns casos, medidas extremas podem ser tomadas:*

*Comportamento dos professores ou dos pais dos alunos de uma equipe que causem desconforto, desrespeito ou que não colaborem para a boa conduta da competição, podem acarretar na desclassificação da equipe.*

Incentivamos o compartilhamento de trabalhos desenvolvidos e aprendizados adquiridos ao longo da jornada, contribuindo para o crescimento coletivo da comunidade. No entanto, é fundamental que as equipes trabalhem de forma independente. O uso de códigos idênticos ou muito semelhantes por equipes diferentes pode comprometer a integridade da competição, e equipes nessa situação estão sujeitas a desclassificação. Colabore para um ambiente criativo e inovador, respeitando sempre os esforços únicos de cada participante.



Espera-se, ainda, que os participantes apresentem os seguintes comportamentos e respeito:

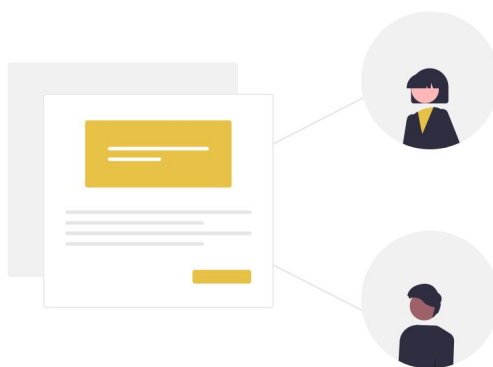
- Tutores **NÃO** devem trabalhar ou auxiliar diretamente na programação dos robôs virtuais.
- Equipes podem ser desclassificadas caso desrespeitem este Código de Conduta.



## Compartilhamento de Conhecimento

---

É o entendimento comum a toda a organização de que todo desenvolvimento pessoal, tecnológico ou curricular deve ser compartilhado entre todos os participantes da competição, durante e depois dela. O melhor ensinamento que pode ser dado à equipe é dotá-la do espírito de cooperação para com os colegas. Qualquer robô ou desenvolvimento feito pelos alunos poderá ser publicado nos websites oficiais, a critério da comissão organizadora.



*Caderno Azul*

# A Plataforma

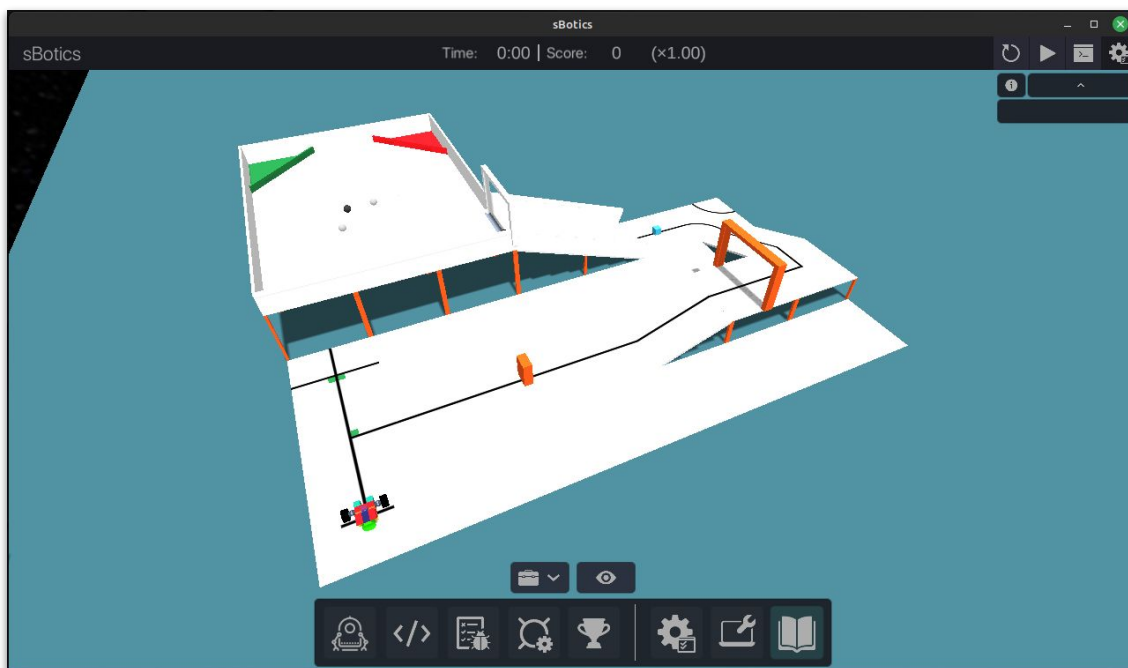


**Como baixar, utilizar e melhor entender a plataforma  
escolhida pela Olimpíada.**



## sBotics

O ambiente utilizado para simulação e programação é o **sBotics**, que é uma plataforma de simulação de *Robótica Educacional*, capaz de gerar virtualmente o ambiente descrito neste manual. Possibilita também a montagem e programação de robôs.



Exemplo do ambiente de simulação com uma arena gerada aleatoriamente



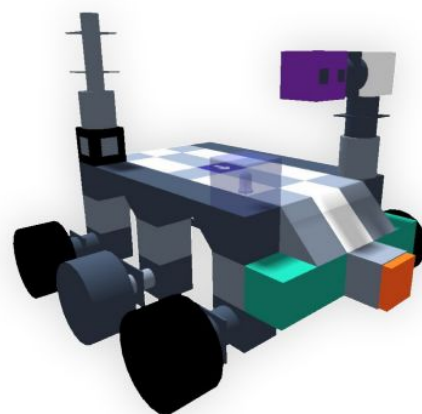
É possível fazer o download do simulador através do site [sbotics.net](http://sbotics.net), com suporte oficial para os sistemas **Windows 8+**, **Linux 64 bits** e **Mac OS 10.14 (Mojave)**.



## Sobre

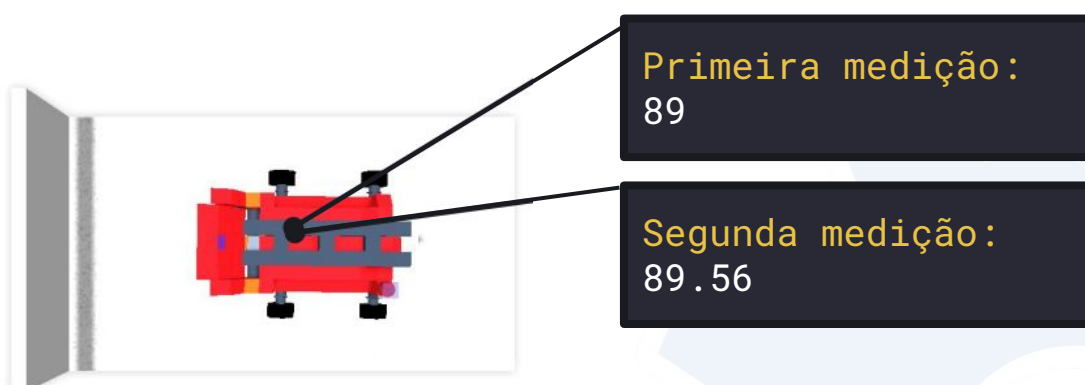
O simulador de Robótica Educacional **sBotics**, projetado para ser uma ferramenta de ensino de **robótica**, vem sendo utilizado pela Olimpíada desde 2020, na Modalidade Resgate Virtual.

Como uma **plataforma independente**, funciona como um **kit de robótica**, e não um simulador de algum kit específico, fazendo a **experimentação, treino, observação** e outros pontos necessários em kits de robótica tradicionais extremamente importantes para seu uso e domínio.



### Randomização de Ambiente

O propósito da plataforma, tal qual o da Olimpíada, é disseminar o conhecimento, melhor preparando os estudantes para as dificuldades que um robô real pode ter. Por isso, o simulador utiliza-se de **aleatorizações mínimas** para simular variáveis reais como: Folgas nos motores, tensão das baterias variando, entre outros fatores. Isso por sua vez torna necessário programações mais “inteligentes”, menos dependentes em tempo ou movimentações simples e com mais checagens, além de dificultar pré-mapeamento e improvisos na lógica.

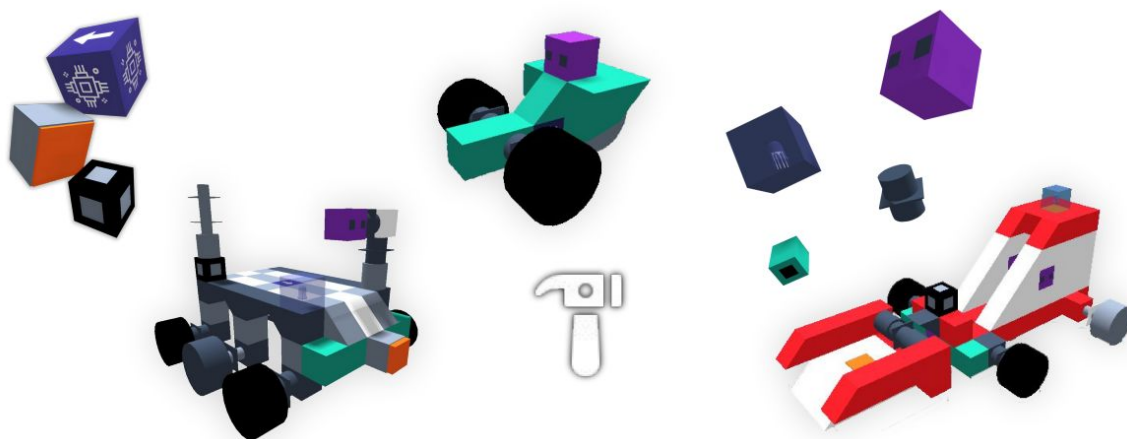


*A mesma programação que move o robô para frente por 1 segundo e mede a distância do retângulo cinza **variou** de 89 para 89.56 em duas execuções que **partiram do mesmo ponto**.*



## Os Robôs

O sBotics dispõe de múltiplos robôs, que devem ser **montados** e **programados** apenas **pelos estudantes** para cumprir **autonomamente** o desafio descrito no **Caderno Verde**.

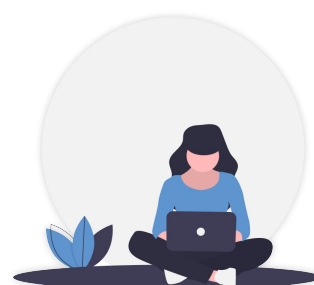


Cada robô possui seu próprio peso (que determina sua agilidade) e número de sensores, que deve ser escolhido ao gosto e estratégia **dos estudantes**, e a própria plataforma trata como mostrar essas informações ao usuário a respeito de cada robô.

## Programação

O simulador dispõe de múltiplas linguagens de programação, que podem ser escolhidas **livremente** pelo usuário para realizar as tarefas dadas. São elas **blockEduc**, **rEduc** e **C#**.

Em uma dada rotina da plataforma, um **único programa** (o mais recentemente compilado) é executado na **arena carregada** (*automaticamente* se realizado pelo componente de competições) e será (re)executado a cada **(re)inicialização do robô** (início da rotina ou após *Falhas de Progresso*).



## Treinos

---

É possível treinar na plataforma utilizando a **geração e configuração de cenário**, possibilitando montar casos descritos por este manual e ir preparando sua programação para a competição.

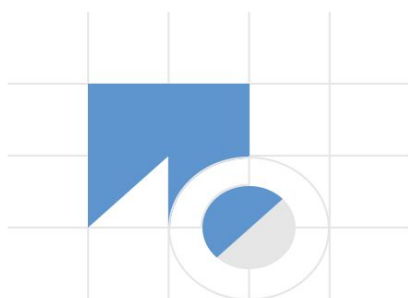
**Note que:** O gerador aleatório de arenas pode *não* prever num mesmo cenário, *todos* os desafios propostos pelo manual, sugere-se usar o Editor de arenas para propor os casos específicos desejados pela equipe em seus treinos.



## Medidas

---

A plataforma **não** possui sistema de medição, e qualquer elemento presente neste manual estará sendo representado na plataforma em suas próprias medidas, com todos os trajetos e elementos testados e possíveis de serem realizados. Por se tratar de uma **única plataforma** sendo utilizada por **todos os usuários**, as condições e situações serão idênticas a todas as equipes dentro de um mesmo Nível.



## Comunidade

---

O sBotics conta com uma comunidade no **Discord** com mais de 2.300 (dois mil e trezentos) integrantes, onde os usuários podem **interagir, tirar dúvidas** e **ajudar** uns aos outros.

É possível acessar o link para a comunidade do *Discord* no site do sBotics! ([discord.sbotics.net](https://discord.sbotics.net)).



## Suporte

---

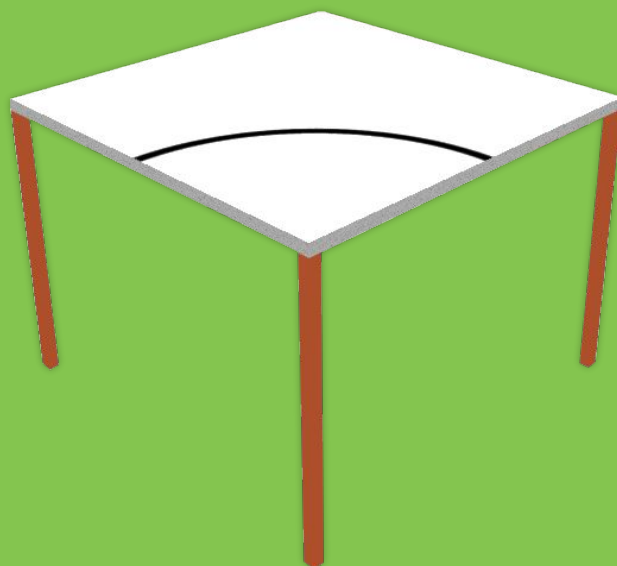
O sBotics conta com um sistema de **suporte** através de um **formulário** (canal oficial de contato), disponível no site ([support.sbotics.net](https://support.sbotics.net)), onde os usuários podem **entrar em contato com uma equipe de especialistas** para **enviar sugestões, tirar dúvidas ou relatar problemas**.

**Atenção:** Relato de problemas relacionados ao funcionamento do sBotics devem ser realizados pelo canal oficial de contato do sBotics!



*Caderno Verde*

# O Desafio

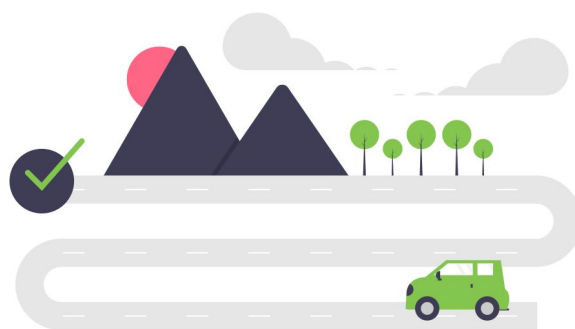


**Regras, descrições e pontuações de cada componente do desafio.**

## Descrição

A missão da **OBR - Modalidade Resgate Virtual Simulação** caracteriza-se por realizar o resgate de vítimas de uma situação de desastre utilizando robôs virtuais em um **ambiente simulado**.

*Em um ambiente hostil, muito perigoso para a saúde do ser humano, um robô virtual **completamente autônomo** programado pela equipe de estudantes recebe uma tarefa difícil: simular o resgate de vítimas **sem** a necessidade de **interferência humana**.*



O robô virtual **deve ser** ágil para **superar obstáculos difíceis**, sejam eles caminhos onde o trajeto não pode ser reconhecido, elementos desconhecidos que devem ser desviados, caminhos sem saída que precisam ser superados, e entre outros contratemplos para conseguir **salvar a(s) vítima(s)**, transportando-a(s) para uma **região segura** onde os humanos já poderão assumir os cuidados, para então **sair** com segurança da construção.

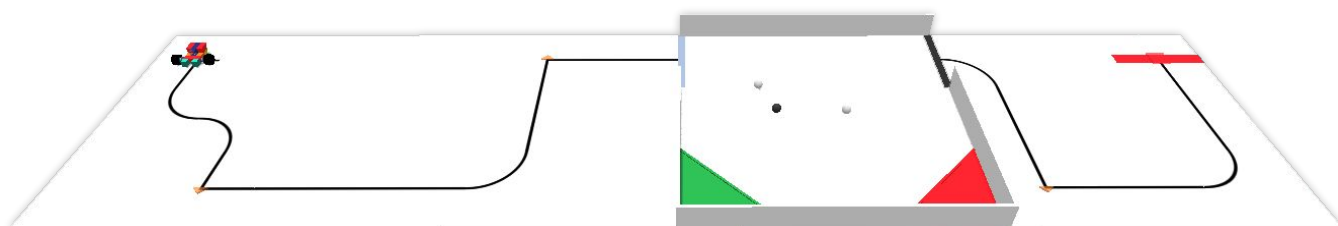
**Gerenciamento de tempo e habilidades técnicas** são **essenciais!** Venha ser a mais bem-sucedida Equipe de Resgate do Brasil!





## Resumo

Um robô **autônomo** deve seguir uma **linha escura** enquanto supera *perigos* diferentes em uma **arena modular** formada por diferentes **ladrilhos**. O piso é **claro** e os ladrilhos podem ter pequenos desníveis entre si e rampas são utilizadas para alterar o nível da arena.

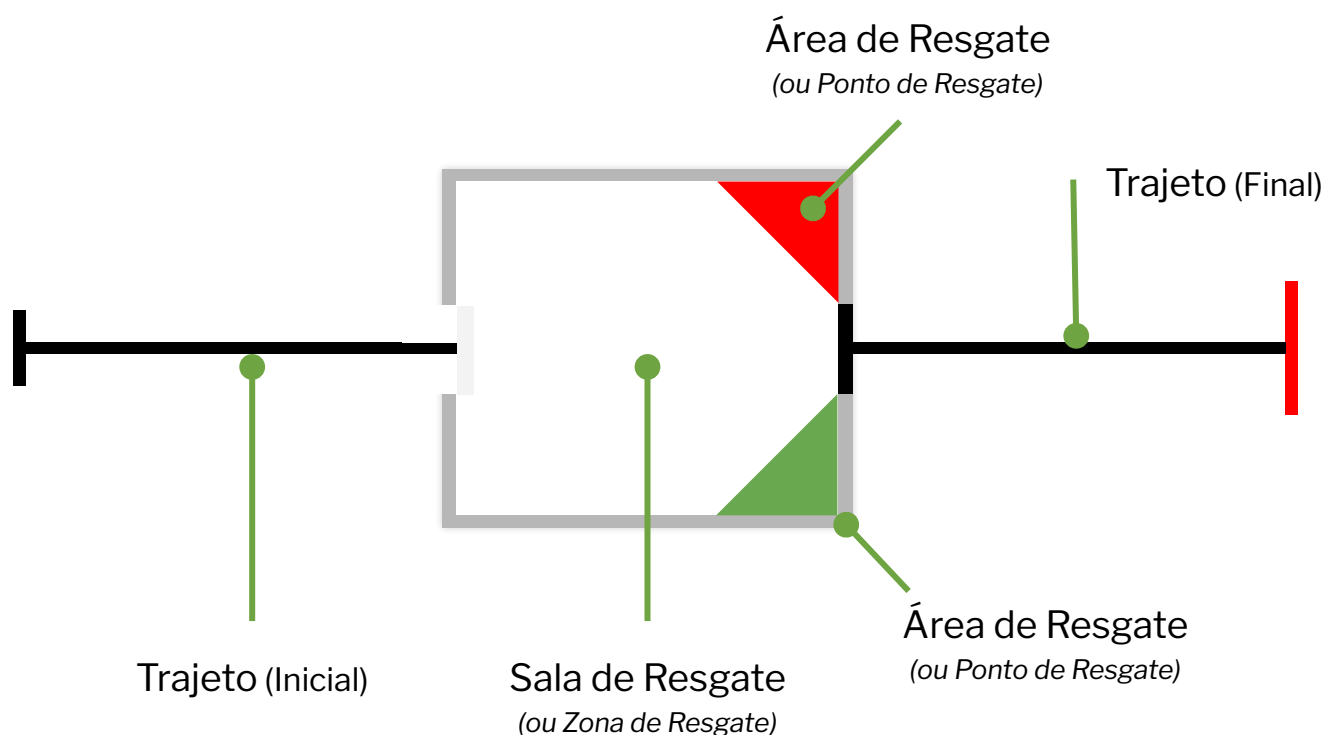


Equipes **não podem** passar informações sobre a arena ao robô, já que ele deve reconhecer a arena *sozinho* através de seus sensores. O robô recebe pontuação **no trajeto** de acordo com a informação seguinte:

|   |           |
|---|-----------|
| Superar Interseção ou Beco sem Saída    | 10 Pontos |
| Superar uma Gangorra                    | 20 Pontos |
| Desviar de um Obstáculo                 | 20 Pontos |
| Superar um Ladrilho Contendo Gap(s)     | 10 Pontos |
| Subir ou Descer uma Rampa               | 10 Pontos |
| Superar um ladrilho contendo Lombada(s) | 10 Pontos |
| Superar uma Passagem                    | 0 Pontos  |
| Ladrilho de Chegada                     | 60 Pontos |

## Resumo (Parte 2)

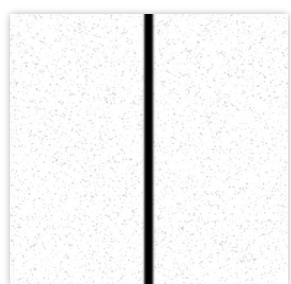
Caso o robô fique preso em algum ponto da arena ele pode ser reiniciado no último **checkpoint**, marcados por pequenos *marcadores* cilíndricos de cor *laranja*. O robô ganhará pontos quando alcançar novos **checkpoints**. Algum lugar no trajeto terá uma sala quadrada ou *retangular*, a **Sala de Resgate**. A entrada para essa sala será demarcada por uma fita de cor prateada no piso e a saída dessa sala será demarcada com uma fita de cor preta no chão.



Dentro da *Sala de Resgate*, o robô deve **localizar e transportar vítimas vivas**, pequenas bolas de cor *prata*, para a **Área de Resgate Verde** e mortas, pequenas bolas *escuras*, para a **Área de Resgate Vermelha**. O robô pode ganhar múltiplos **multiplicadores** por *resgatar de vítimas*. O robô deve então **sair da Sala de Resgate** e seguir a linha até que o **Ladrilho de Chegada com uma Linha Vermelha** seja alcançado.

## Arena

**Todo** o percurso é *representado* por **ladrilhos** modulares, peças que ocupam um espaço **quadrado** nas quais linhas, paredes, obstáculos e **outros tipos** de *perigos* são montados, *independente* se os mesmos possuem pontuação ou não. A combinação de ladrilhos gera um **número muito grande de percursos** para serem concluídos pelo robô. Ladrilhos podem ter qualquer espessura e uma dada arena terá um **mínimo de 8 ladrilhos**, excluindo ladrilhos especiais como os de *Partida* e *Chegada*.



*Ladrilho simples com uma linha reta visto de cima.*



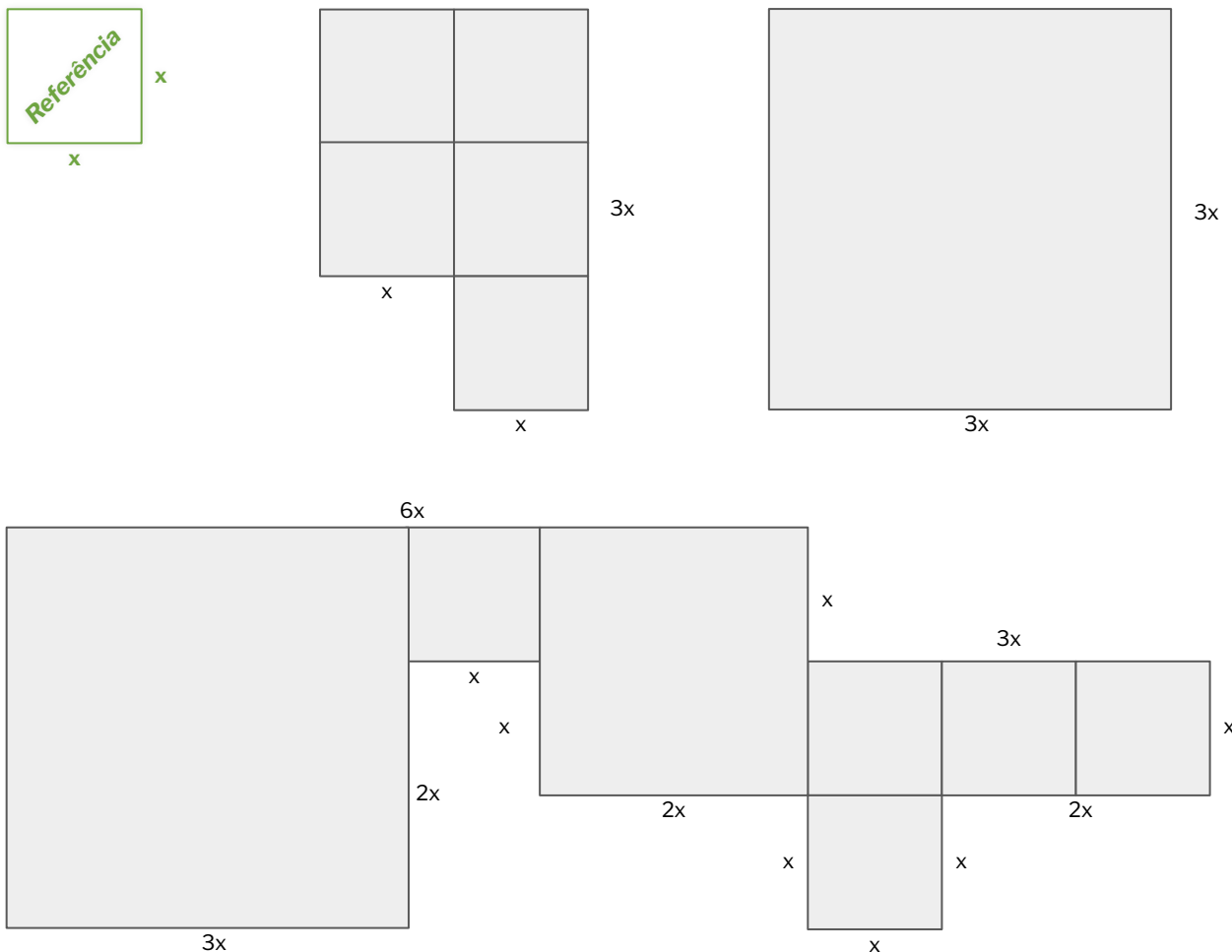
*Ladrilhos podem ser elevados, acessados por **ladrilhos de rampa**.*

É possível que haja ladrilhos que possuem um tamanho maior que outros, porém se mantendo na **mesma proporção** e possuindo um **tamanho possível de ser composto pelos ladrilhos de tamanho “padrão”**. A seguir serão apresentados exemplos válidos e inválidos de diferentes tamanhos de ladrilhos

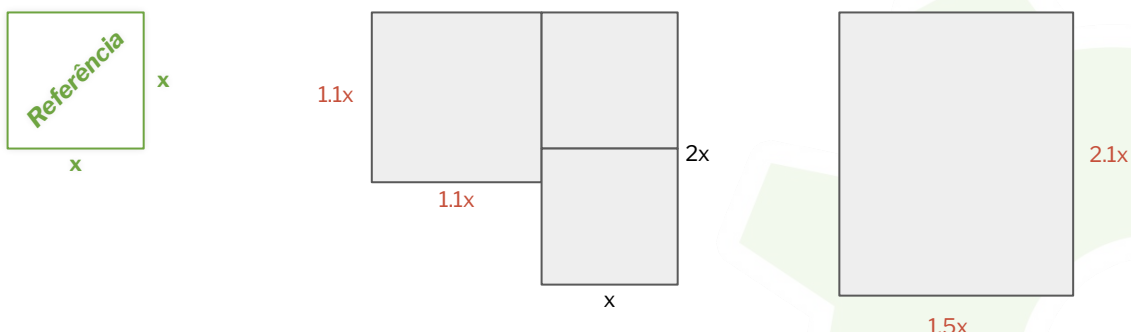
# Construção da Arena

O piso da plataforma simulada segue alguns padrões de dimensão, visíveis nos exemplos abaixo:

## Exemplos válidos de junção de ladrilhos:

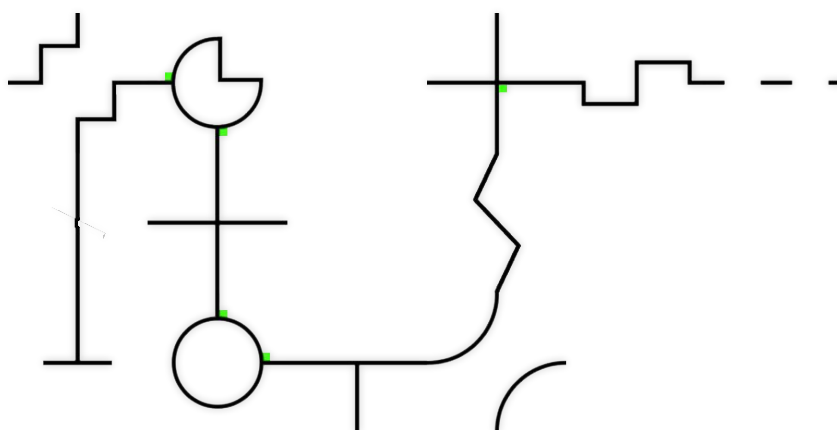


## Exemplos inválidos de junção de ladrilhos:



## Piso

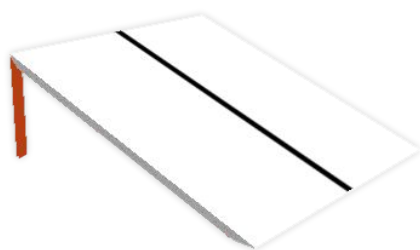
O Piso da **arena**, ladrilhos do trajeto, possui um **fundo claro** e uma **linha escura**, representando a área do desastre e o caminho a ser seguido, respectivamente. Esse caminho não é um trajeto perfeitamente reto, e pode realizar curvas de qualquer angulação, podendo haver *perigos*, que serão explicados posteriormente neste caderno.



Exemplo **ilustrativo** de um **fragmento** de um percurso.

O espaço entre os ladrilhos deve ser **mínimo**, e *não perfeito*, sendo previsível pequenos **desencaixes** que podem servir de empecilho para o robô, sempre dentro da missão e descrição do desafio.

Robôs devem estar preparados para andar por ladrilhos que formam **pontes sobre outros ladrilhos**. Ladrilhos colocados acima de outros terão pilares para sustentação, e o espaço mínimo para a passagem inferior é de *2 ladrilhos de altura*.



Ladrilho inclinado de linha reta.

Ladrilhos de linha reta (incluindo aqueles que possuem perigos pontuáveis) podem ser **inclinados** para **elevar ou abaixar o nível do piso** da arena, possuindo uma angulação entre *15 a 25 graus*, onde a *plataforma simulada* por padrão utiliza valores **próximos a 20°**.



## Linha

A linha escura presente no percurso, forma um caminho no piso que deve ser seguido pelo robô.

Com exceção de alguns *Perigos*, não há uma limitação específica de como a linha de um ladrilho deve ser desenhada, ficando a critério do **elaborador do desenho da arena** escolher **quaisquer** das opções disponíveis na *plataforma* de simulação sBotics.



*Ladrilho onde a linha é em sua extremidade*

*A disposição das linhas e o percurso no geral pode mudar entre rodadas da competição.*

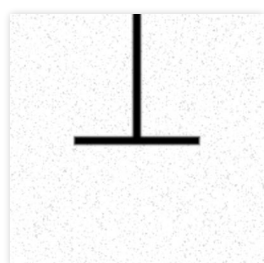
No link a seguir, você encontrará modelos de ladrilhos.

**Importante:** são apenas modelos, você poderá encontrar ladrilhos diferentes e/ou modificados em relação aos disponibilizados.

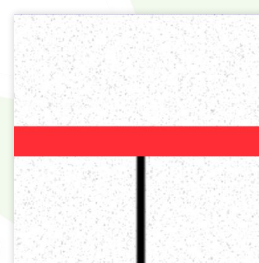
<https://obr.robotcup.org.br/documentos-e-manuais/>

## Disposição do Trajeto

O **Trajeto Inicial** inicia-se com o *ladrilho de partida*, onde será a **posição inicial** do robô. Ele se estende com diversos “*perigos*” até a **faixa prateada**, entrada da **Sala de Resgate**. Já o **Trajeto Final** inicia com a **faixa preta**, saída da **Sala de Resgate**, e estende-se até o *ladrilho de chegada*, fim do percurso. Durante **todo o trajeto** é possível encontrar **Perigos**.



*Ladrilho de Partida.*



*Ladrilho de Chegada.*



## Perigos do Trajeto

No trajeto podem haver diversos *perigos*, tarefas que devem ser **completadas** para o robô continuar (ou *terminar*) o percurso e receber **pontos** como recompensa, sem causar **Falhas de Progresso** (explicadas em outra seção deste caderno). Todos os perigos descritos podem aparecer em qualquer parte da arena.

Cada perigo será pontuado **apenas uma vez por cada direção** no percurso. Pontos não serão dados para tentativas subsequentes dentro do percurso.

### Modelo de detalhamento de perigos presentes neste caderno:

Imagem do  
Componente

Descrição sobre o componente.

- ❖ Especificações técnicas serão apresentadas em **verde**.
- Informações adicionais utilizadas para especificar melhor **diferenças entre níveis** e outras informações.

Imagem adicional  
do Componente

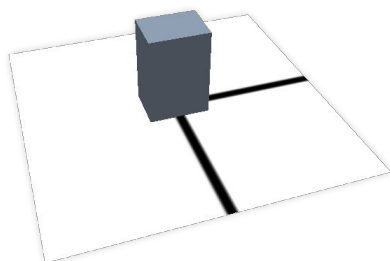
Imagem adicional  
do Componente

Imagem adicional  
do Componente

## Perigos do Trajeto

### Obstáculo

20 Pontos



*Ladrilho com um obstáculo.*

Obstáculos são barreiras dentro de um determinado ladrilho que podem **variar de cor e tamanho**, forçando robô a **desviar** saindo da linha durante alguns instantes. Caso o robô desvie do obstáculo **com sucesso** será atribuído **20 pontos**. Espera-se que um robô supere vários obstáculos.

- ❖ Obstáculos podem representar tijolos, blocos, **pesos** e outros objetos **grandes** e pesados. Obstáculos **não** devem ocupar **mais de uma linha ou ladrilho** e sua altura é de no **máximo ½ ladrilho**.
- **Não** será permitido ao robô, após um desvio, seguir por outra linha da arena, e sim obrigatoriamente, a sua linha subsequente após o obstáculo. Caso o robô não consiga retornar à linha no tempo máximo de **50 segundos**, será considerada uma **Falha de Progresso**.
- Se o robô empurrar ou deslocar algum obstáculo (que podem ou não ser  *muito pesados* ou *fixos ao ladrilho*), será considerado **Falha de Progresso**. Será considerada a pontuação de um obstáculo caso **metade do robô** visite o **ladrilho seguinte** quando visto de cima.

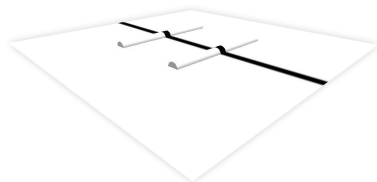




## Perigos do Trajeto

### Lombada (Redutor de Velocidade)

10 Pontos



*Ladrilho com múltiplas lombadas.*

Lombadas podem ser colocadas em **qualquer lugar** em um determinado ladrilho, pontuando **10 pontos** cada ladrilho quando superadas pelo robô, **independente de quantas lombadas tenha neste ladrilho.**

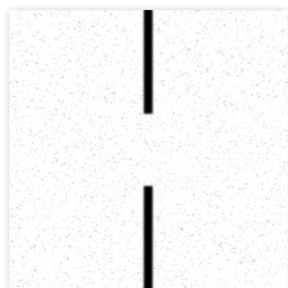
- ❖ **Não** há especificações para lombadas, que podem **variar** de **tamanho** e **angulação**, servindo como “detrito” no percurso do robô.
- Será considerada a pontuação de lombada(s) caso **metade do robô** visite o **ladrilho seguinte** quando visto de cima.



## Perigos do Trajeto

### Gap (Lacuna)

10 Pontos



Gap vista de cima.

Gaps (ou lacunas) são “falhas” na linha do percurso e podem existir uma ou mais dentro de um **mesmo ladrilho**. Pontua-se 10 pontos para cada ladrilho superado pelo robô, **independentemente do número de gaps presentes no ladrilho**.

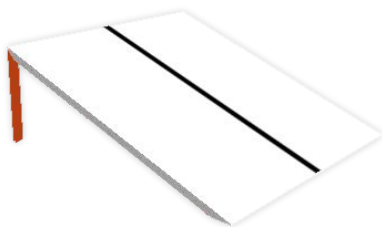
- ❖ Gaps são colocados em trechos de **linha reta** do percurso, tendo como possibilidades de ladrilhos os presentes na *plataforma de simulação sBotics*.
- Será considerada a pontuação de uma gap caso **metade do robô** alcance o **ladrilho seguinte** quando visto de cima.



## Perigos do Trajeto

### Rampa

10 Pontos



Ladrilho inclinado de linha reta.

Ladrilhos de linha reta (podendo incluir uma ou mais *Gaps* e/ou *Lombadas*) podem ser **inclinados** para elevar ou abaixar a altura do piso da arena, atribuindo **10 pontos**, fora aqueles atribuídos se houverem outros *perigos* na rampa, caso o robô consiga subir ou descer o ladrilho, a rampa poderá ter o tamanho de vários ladrilhos, sendo considerado apenas uma rampa.

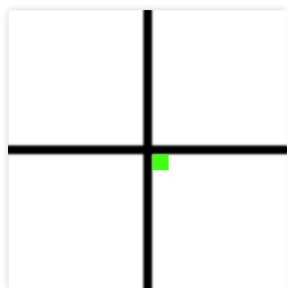
- ❖ A angulação da rampa deve ficar entre **15 a 25 graus**, onde a *plataforma simulada* por padrão utiliza valores **próximos a 20°**.
- Será considerada a pontuação de uma rampa caso **metade do robô** visite o **ladrilho seguinte** quando visto de cima (subindo ou descendo).



## Perigos do Trajeto

### Interseção com Marcações

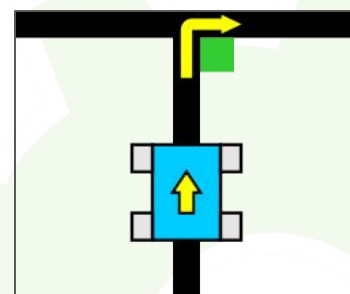
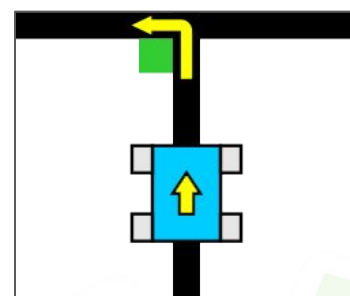
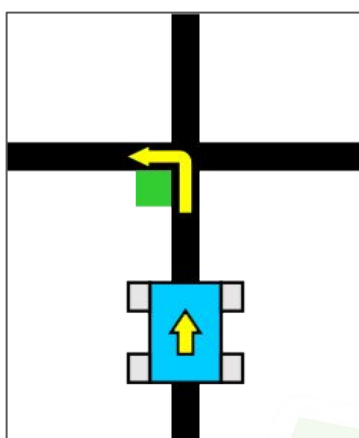
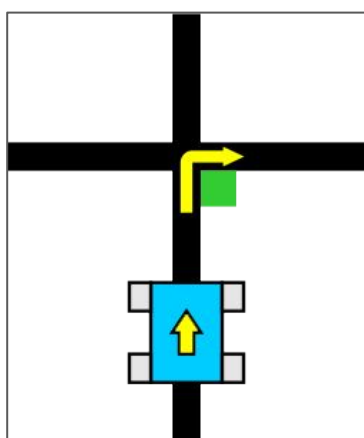
10 Pontos



Interseções com Marcações são caminhos de 3 ou 4 ramificações presentes no percurso no qual o robô deve seguir o sentido indicado pelo marcador de cor verde diretamente anterior à interseção para alcançar **10 pontos**.

*Interseção com  
marcação vista de cima*

- ❖ As dimensões da marcação verde estão definidas **pela plataforma**.
- A interseção será considerada superada (e sua pontuação dada) quando o robô seguir o **caminho indicado pela marcação verde** e caso **metade do robô** visite o **ladrilho seguinte** quando visto de cima. Será considerada **Falha de Progresso** caso o robô **não** execute corretamente uma interseção, seguindo o caminho errado.

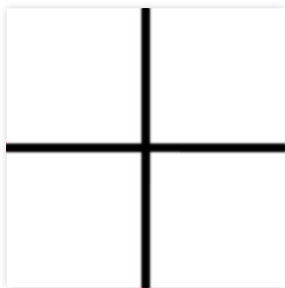


*Exemplos de Interseções  
marcadas.*

# Perigos do Trajeto

## Interseção sem Marcações

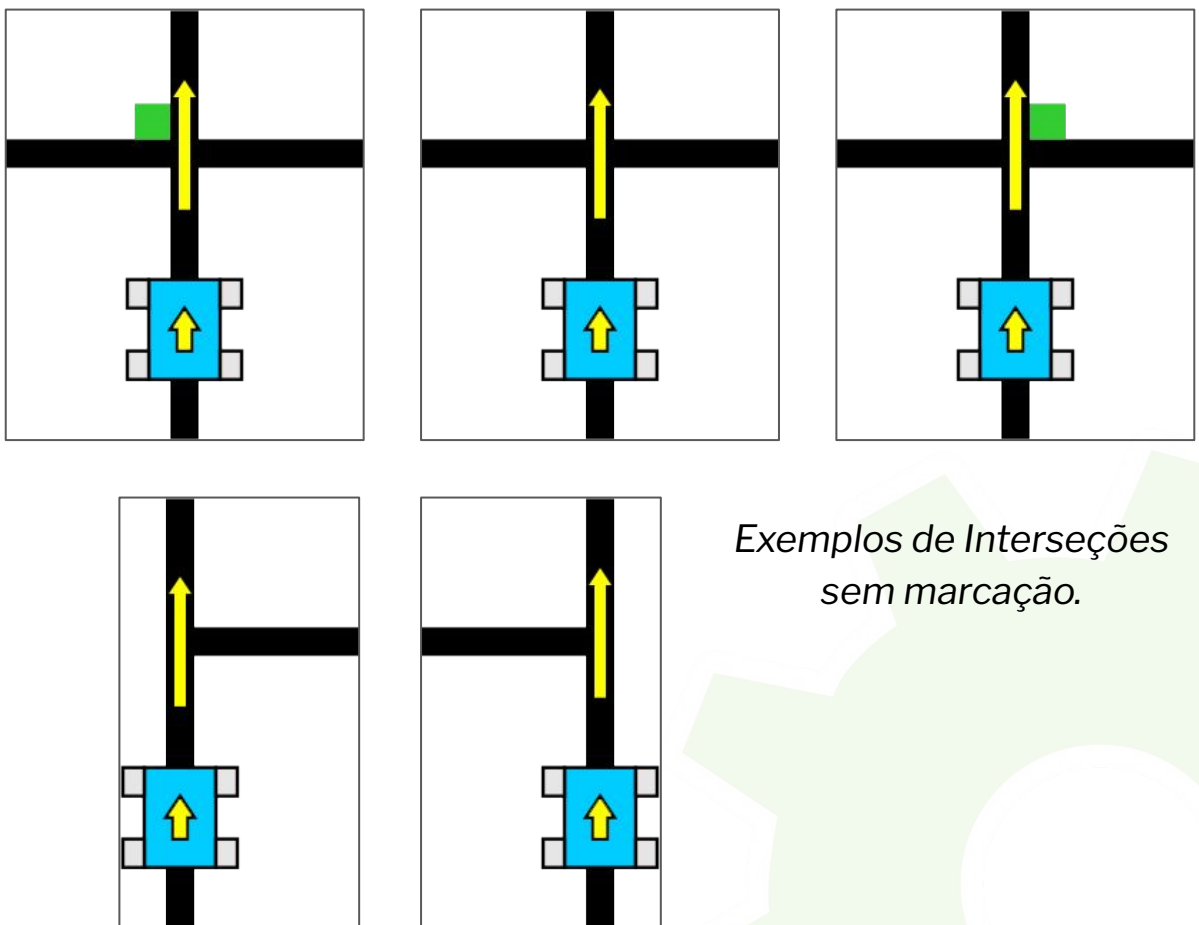
10 Pontos



Interseções *sem* Marcações são caminhos de 3 ou 4 ramificações presentes no percurso no qual o robô deve seguir **reto** para alcançar **10 pontos**, **mesmo que** haja um marcador de cor verde **diretamente após a interseção**.

*Interseção sem  
marcação vista de cima*

- A interseção será considerada superada, e sua pontuação dada, quando o robô seguir **reto**. Será considerada **Falha de Progresso** caso o robô **não** execute corretamente uma interseção, seguindo o caminho errado.

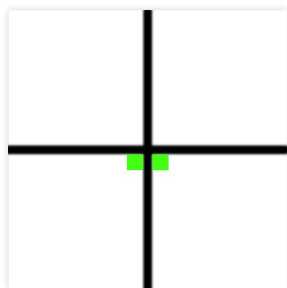


*Exemplos de Interseções  
sem marcação.*

## Perigos do Trajeto

### Beco Sem Saída

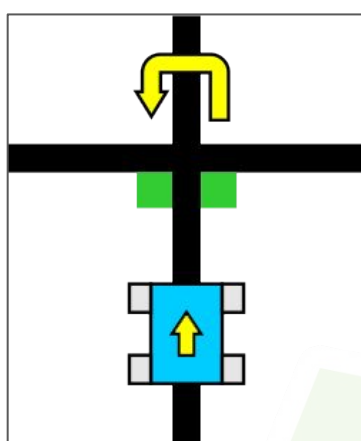
10 Pontos



*Beco sem saída visto de cima.*

Interseções com Marcações que possuem **duas marcações de cor verde** no trajeto do robô são consideradas “Becos sem Saída”, onde o robô deve seguir o sentido **oposto** pelos marcadores de cor verde diretamente anterior à interseção para alcançar **10 pontos**.

- ❖ Para especificações da *marcação verde*, ver o perigo: **Interseção com Marcação**.
- O Beco sem Saída será considerado superado (e sua pontuação dada) quando o robô seguir o caminho **oposto** indicado pelas duas marcações verdes e **metade do robô** visite o **ladrilho seguinte** quando visto de cima. Será considerada **Falha de Progresso** caso o robô **não** execute o comando corretamente no perigo em questão, seguindo o caminho errado.

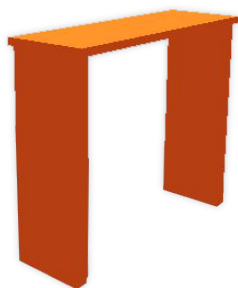


*Exemplo de Beco sem Saída.*

## Perigos do Trajeto

### Passagem

0 Pontos



*Passagem.*

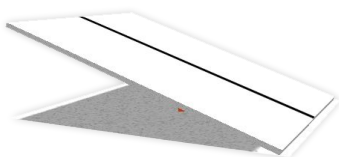
Passagem é um perigo formado por 3 pedaços sólidos que pode ser colocado em linhas retas, que precisam ser superadas, porém não atribuem pontuação.

- ❖ Passagens são **fixas** ao piso e ocupam um espaço horizontal menor que um ladrilho.
- Caso o robô não consiga ultrapassar a passagem, será considerada **Falha de Progresso**.

## Perigos do Trajeto

### Gangorra

20 Pontos



Ladrilhos de linha reta sem elementos pontuáveis inclinados em um pivô atuam como *gangorras*, que quando superadas pontuam **20** pontos.

*Gangorra vista  
de lado*

- ❖ Gangorras possuem a mesma angulação de uma **rampa** (ver seção específica), e seu pivô é fixo ao piso.
- A pontuação **só será dada** caso o robô as complete com sucesso, atravessando de uma ponta a outra da mesma e **metade do robô** visite o **ladrilho seguinte** quando visto de cima.
- Quando ocorre uma **Falha de Progresso**, e a gangorra já tiver sido movimentada pelo robô, ela voltará para a sua posição inicial.

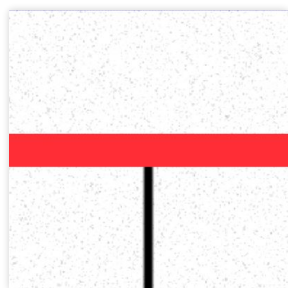




## Perigos do Trajeto

### Ladrilho de Chegada

**Max. 60 Pontos**



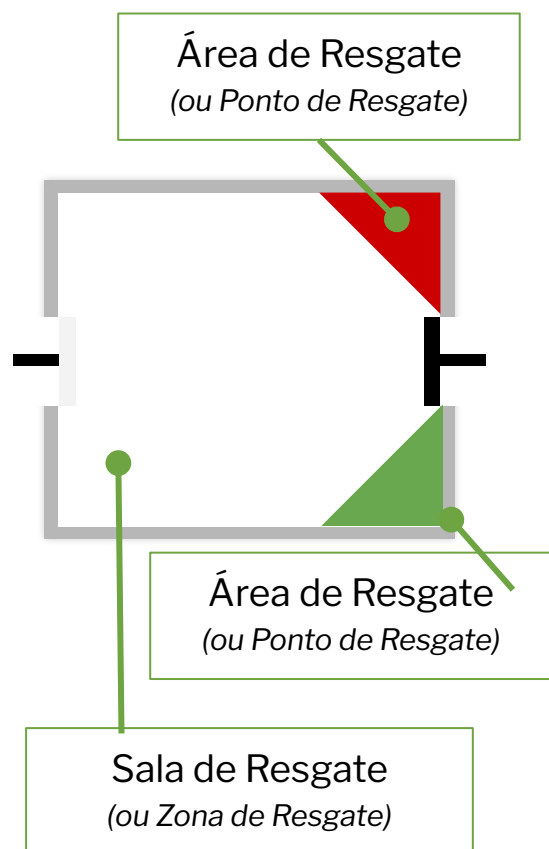
*Ladrilho visto de cima*

O ladrilho de chegada é o último ladrilho do percurso, pontos serão obtidos quando o robô atingir o ladrilho e ficar **completamente parado** por **ao menos 5 segundos**. A pontuação dada corresponde a no **Máximo 60 pontos**, **diminuindo 5** a cada **Falha de Progresso/Nova Tentativa** realizada no percurso pelo robô.

- ❖ A faixa vermelha inicia-se no centro do ladrilho e estende-se horizontalmente de uma ponta a outra no mesmo, possuindo **no mínimo** a espessura da linha escura em si.
- Caso o robô não pare diretamente na faixa vermelha ou passe direto, será considerada uma **Falha de Progresso**.
- A pontuação do ladrilho de chegada não pode ser negativa e é dado por:

$$\text{Ladrilho de chegada} = 60 - 5 \times (\text{quantidade de falhas de progresso})$$

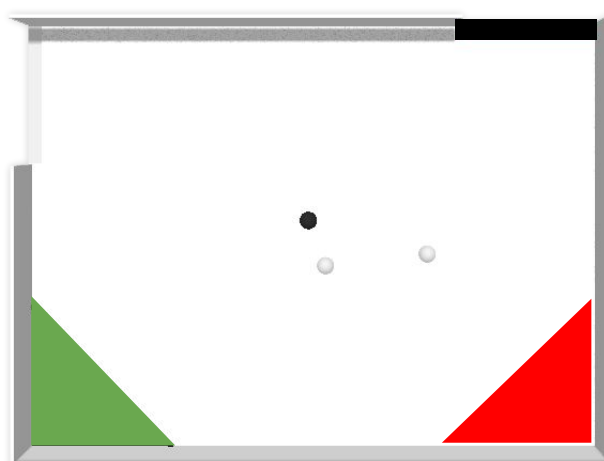
## Resgate (Sala de Resgate)



Resgate é o ápice da tarefa dada ao robô, onde o mesmo deve **resgatar as vítimas** (explicado a seguir neste caderno) em uma **área segura triangular** (a Área de Resgate).

A linha escura **termina** na entrada da *Sala de Resgate* (demarcada por uma fita de cor *prata*), **iniciando** novamente após a saída da mesma (demarcada por uma fita de cor *preta*).

A *Sala de Resgate* possui como tamanho, **3x3** ou **3x4 ladrilhos**, como no exemplo abaixo:



Exemplo de uma sala de resgate  
3 x 4 ladrilhos

## Componentes de Resgate

---

Assim como no trajeto demarcado com a linha, o *resgate* dispõe de várias tarefas que devem ser realizadas pelo robô para alcançar **multiplicadores**, esses serão melhor explicados na seção de **Multiplicadores de Resgate** deste caderno.

**Modelo de detalhamento dos componentes de resgate presentes neste caderno:**

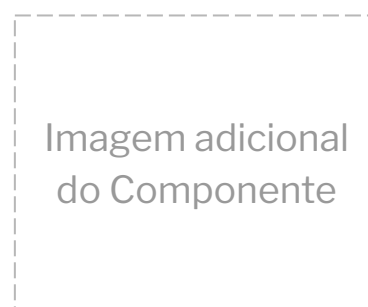
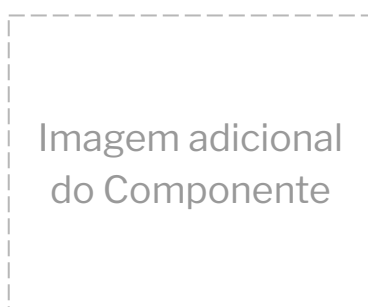
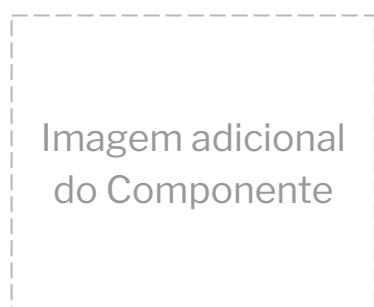
### Nome do Componente *(Nome Alternativo)*

---



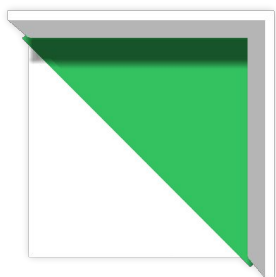
Descrição sobre o componente.

- ❖ Especificações técnicas serão apresentadas em **verde**.
- Informações adicionais utilizadas para especificar melhor **diferenças entre níveis** e outras informações.



## Componentes de Resgate

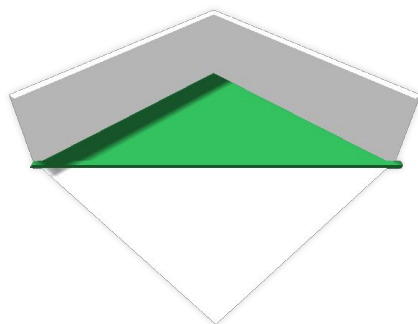
### Área de Resgate



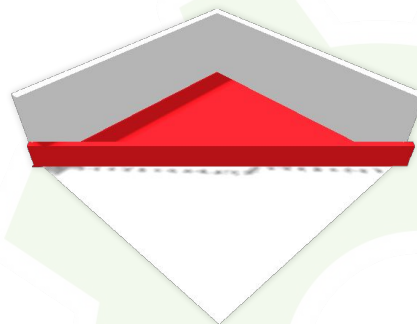
Área de resgate visto de cima

A área de resgate é um *triângulo retângulo*, onde as *Vítimas* devem ser entregues para a atribuição de **multiplicadores**, explicados na seção de *Multiplicadores de Resgate*. Vítimas **completamente dentro da área de resgate** que **não** estejam mais em **contato com o robô** serão consideradas **resgatadas**.

- ❖ Os lados do triângulo retângulo são do **tamanho de um ladrilho**. A área de resgate possui uma elevação em sua entrada (aresta que não toca a parede), onde para as equipes de **Nível 1** é uma pequena barreira, e para as equipes de **Nível 2** corresponde a uma barreira maior.
- A área de resgate será colocada **em uma das** quinas da *Sala de Resgate*, que não estejam sendo bloqueadas pela entrada ou saída.
- Após uma **Falha de Progresso** a Área de Resgate pode ser aleatorizada para um dos cantos, não obstruídos, da Sala de Resgate.
- A área de resgate será melhor explicada mais a frente com os outros *componentes* de resgate.



Representação da  
Área de Resgate  
**Nível 1**



Representação da  
Área de Resgate  
**Nível 2**



# Componentes de Resgate

## Vítimas



*Vítimas como vistas na Sala de Resgate*

Vítimas podem estar em **qualquer lugar** no piso da Sala de Resgate, e são divididas em dois tipos: **Vítimas Vivas**, cobertas de material prateado condutor, e **Vítimas Mortas**, cobertas de material escuro isolante. Vítimas vivas devem ser **entregues à Área de Resgate Verde** e as vítimas mortas devem ser entregues à **Área de Resgate Vermelha** para a atribuição de **multiplicadores**, explicados na seção de *Multiplicadores de Resgate*.

- ❖ *Vítimas são pequenas esferas leves que possuem uma textura dependendo de seu tipo.*
- Vítimas estarão localizadas **aleatoriamente** na Sala de Resgate;
- Após uma **Falha de Progresso** o posicionamento das vítimas será aleatorizado para outros, ou para os mesmos locais;
- Caso o robô lance a vítima para fora da área de resgate, ela só será reposicionada na próxima **tentativa** (Falha de Progresso);
- Após o resgate de uma vítima **não** será mais possível modificar a posição da mesma;
- Serão **exatamente duas Vítimas Vivas** e **uma Vítima Morta**.



## Multiplicadores de Resgate

Multiplicadores são dados quando vítimas são **entregues** na **Área de Resgate**, isso é, estão **completamente dentro das áreas verde e vermelha** e **não** estão mais em **contato com o robô**, onde são removidos para não atrapalhar a execução. Esses multiplicadores serão dados quando as vítimas forem entregues na **área de resgate correta** sendo a verde para as **vítimas vivas** e a **vermelha** para a **vítima morta**, sendo que esta só pontuará depois que todas as vítimas vivas já tenham sido salvas. Os multiplicadores são **acumulados** e **multiplicados** pela **pontuação total** ao **fim da execução** do desafio (ver seção Fim da Rotina). Somente as vítimas colocadas pelo robô no ponto de evacuação apropriado receberão multiplicadores.



### Resgate de Vítimas - Nível 1 e 2:

|   |       |
|---|-------|
| Resgate de uma <b>vítima viva entregue na área de resgate Verde</b>   | × 1.3 |
| Resgate de uma <b>vítima morta entregue na área de resgate Vermelha</b> após Resgate <b>total</b> das vivas | × 1.3 |
| Resgate de uma <b>vítima viva ou morta entregue na área de resgate invertida ou fora da ordem</b>           | × 0   |



## Falhas de Progresso (Tentativas)

Falhas de Progresso são eventos **induzidos manualmente** pelo *botão de nova tentativa* pela equipe ou causados por **comportamentos inesperados** do robô no percurso, como alterações e desvios do caminho correto, entre outras formas já descritas. Cada falha de progresso será contabilizada uma **nova tentativa** em um determinado *checkpoint*, o último superado pelo robô.

Após uma Falha de Progresso, a programação do robô será **reiniciada e executada novamente** de forma **automática**.



É considerada uma **Falha de Progresso**:

- Quando a equipe declara uma Falha de Progresso manualmente;
- Quando o robô perde a linha escura **ou** fica parado por **10 segundos**;
- Quando **qualquer condição** de Falha de Progresso descrita em algum *Perigo* do percurso é satisfeita;
- Quando o robô atinge uma linha que não está na sequência planejada.

**Observações:**

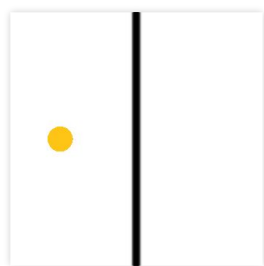
- Quando uma Falha de Progresso ocorre, o robô é posicionado no **último checkpoint superado pelo robô** virado para o **sentido da Sala de Resgate**;
- Não há limite de tentativas, ou Falhas de Progresso, a partir de um *checkpoint*;
- **A partir da terceira tentativa** a equipe terá como opção continuar a execução para pontuar itens *ainda não pontuados* ou **avançar** para a **primeira tentativa do próximo checkpoint** no percurso;
- Após uma Falha de Progresso, **comportamentos** relacionados às falhas descritos nos **Perigos** do trajeto serão realizados.



## Checkpoints (Pontos de partida)

Um *checkpoint* é um **ladrilho** demarcado no qual o robô será manualmente colocado após uma **Falha de Progresso**.

*Checkpoints* **não** estarão posicionados em ladrilhos que possuem elementos pontuáveis.



Referência de **um**  
Ladrilho *Checkpoint*.

O **Ladrilho de Partida** em si é um *checkpoint* no qual o robô inicia-se.

Um **marcador de *checkpoint*** é um pequeno objeto cilíndrico de cor *laranja* usado para marcar quais ladrilhos são *checkpoints*.

O número de *checkpoints* e seus locais será predeterminado pelo **elaborador do desenho da arena**.

O *checkpoint* será considerado alcançado quando o robô **entrar** no ladrilho em questão.

Quando ocorrer falta de progresso entre dois *checkpoints* (ou um *checkpoint* e o ladrilho objetivo) com a sala de resgate entre eles, cada um dos multiplicadores de resgate obtidos será deduzido:

Multiplicadores de resgate = multiplicadores de resgate obtido - (0.05 x (número de falha de progresso no trecho que contém uma sala de resgate)).

Os multiplicadores ganhos para resgates bem-sucedidos de vítimas nunca serão inferiores a 1,25.

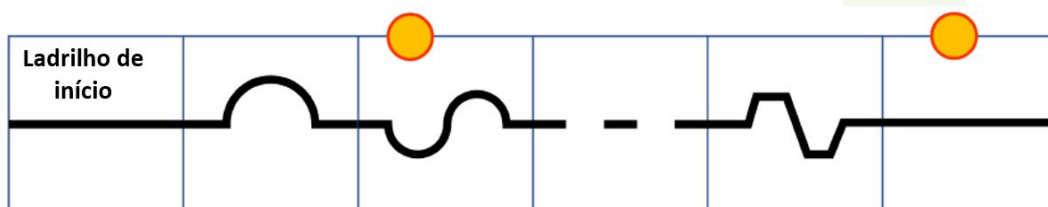
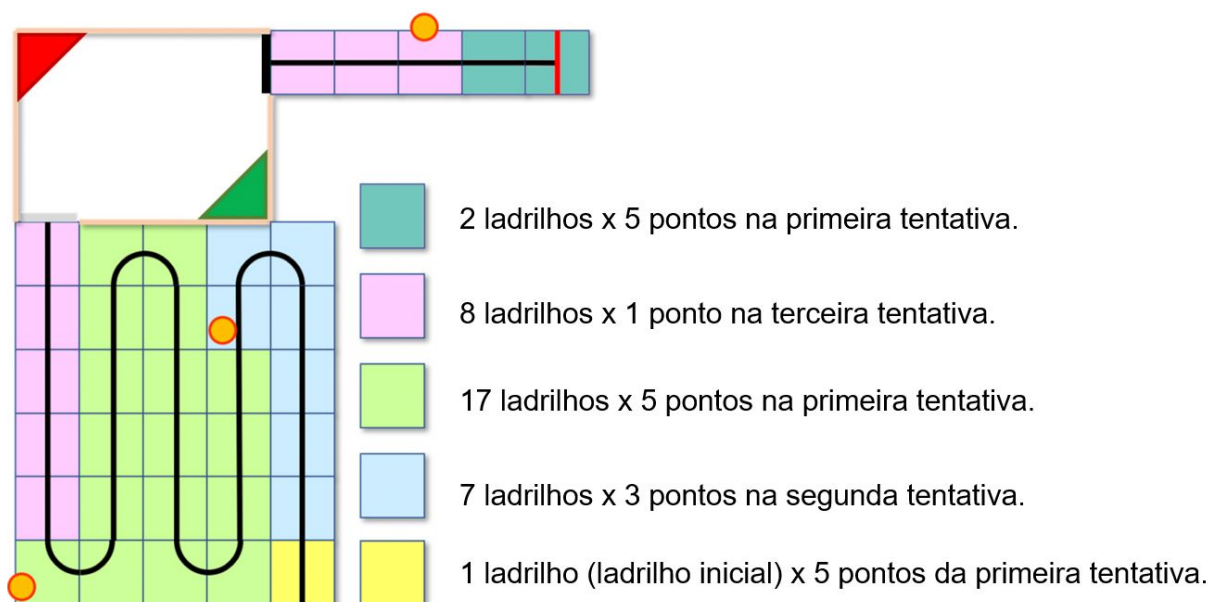




## Pontuação de *Checkpoints*

Quando um robô alcança um Ladrilho *Checkpoint* ele ganhará pontos para **cada ladrilho** que passou **desde o último checkpoint**. A pontuação por ladrilho dependerá de quantas **tentativas** o robô fez para alcançar o *checkpoint*, visível *abaixo* junto com um desenho que simula uma execução.

|                        |                              |
|------------------------|------------------------------|
| 1ª Tentativa           | <b>5 Pontos</b> por Ladrilho |
| 2ª Tentativa           | <b>3 Pontos</b> por Ladrilho |
| 3ª Tentativa           | <b>1 Ponto</b> por Ladrilho  |
| 4ª Tentativa em diante | <b>Sem Pontuação</b>         |



Quando o robô é colocado no ladrilho inicial ele recebe 5 pontos por ultrapassar o checkpoint implícito

O robô receberá pontos por 2 ladrilhos

Aqui o robô recebe 10 pontos pelo ladrilho de gap

O robô receberá pontos por 3 ladrilhos



## Rotina

---

A Rotina inicia-se quando a equipe decide executar sua programação. O robô então é posicionado no **Ladrilho de Partida** e deve começar a navegar pelo trajeto, superando os perigos e realizando o resgate de vítimas e finalizando com o **Ladrilho de Chegada** de forma autônoma (como já descrito em outras seções deste caderno).

*A trajetória das linhas oficiais de cada etapa da competição **não** será divulgada previamente em **hipótese alguma**. Assim, a capacidade do robô seguir um caminho desconhecido **faz parte do desafio**.*



### Observações:

- O tempo de execução da Rotina é de **8 minutos**;
- Alguns elementos como curvas, tamanho de obstáculos e outras pequenas alterações **podem ser realizadas** anteriormente ao início da Rotina, para evitar pré-mapeamento do desenho;
- O robô **não será alterado** durante a Rotina, incluindo o posicionamento de seus componentes como o Atuador.

*A equipe **não tem permissão** para mudar o programa ou alterar o robô após o início da Rotina.*



## Fim da Rotina

---

A equipe pode optar para parar a *Rotina* a **qualquer momento**, fazendo com que a pontuação e multiplicadores a serem consideradas sejam aquelas obtidas **até** o momento da desistência total da *Rotina*, e seu tempo de execução, para efeito de desempate, será o tempo máximo da *Rotina* (**8 minutos**).

A *Rotina* também se encerra após **8 minutos (480 segundos)**, parando **qualquer** execução **independente do local** e considerando tudo o que foi obtido **até** o momento da parada.

## Pontuação Final

---

A pontuação final será dada como o **inteiro mais próximo** do resultado calculado a partir da fórmula abaixo:

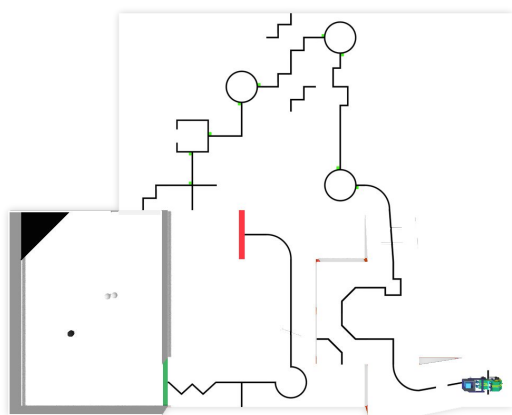
**Pontuação** = (Pontuações do Trajeto + Ladrilho de chegada) x (Multiplicadores de Resgate\*)

\* Multiplicação de **todos** os multiplicadores obtidos.

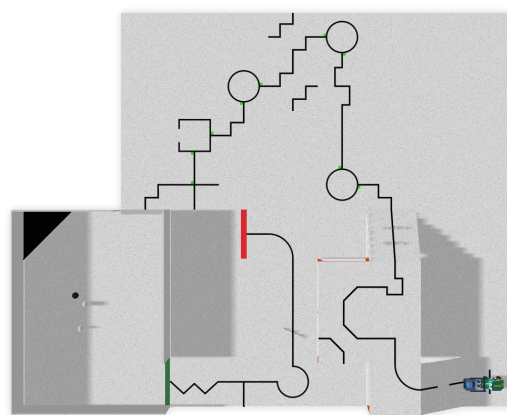


## Condições de Ambiente

As condições de ambiente na competição **podem ser diferentes** das condições de treino, os times devem estar preparados para **ajustar os robôs** para a ocasião. Uma dessas condições é a iluminação, cada arena possui condições de iluminação diferentes baseadas no horário da arena, que podem influenciar diretamente a leitura dos sensores do robô.



Exemplo de arena às 13h



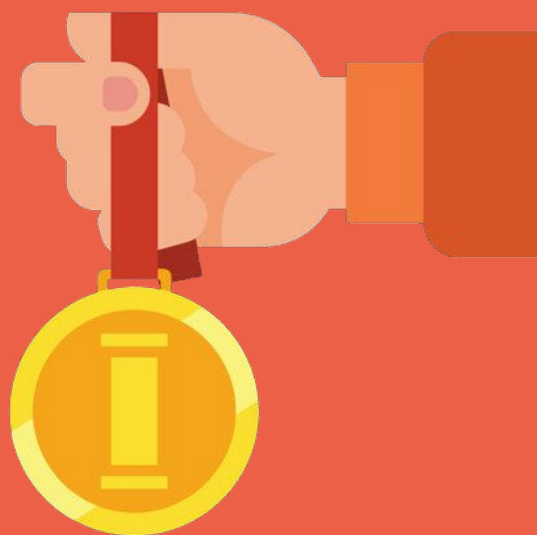
Exemplo de arena às 17h

As calibrações podem ser realizadas **antes** do momento oficial de participação da equipe. No entanto, **não é possível alterar a programação do robô após o início da participação.**

Outra condição de ambiente é a imperfeição de elementos da arena, que sofrem pequenas aleatorizações pelo simulador como parte da **randomização de ambiente**, melhor descrito no **Caderno Azul** na subseção de mesmo nome.



# A Competição

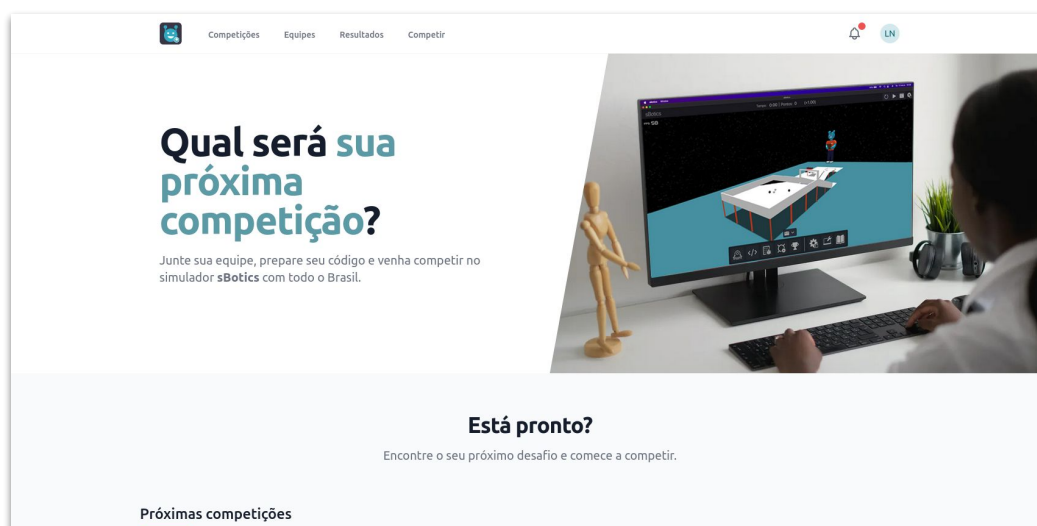


**Regras gerais, limitações e funcionamento da competição e seu(s) sistema(s).**

# Introdução

Neste caderno, serão apresentados os detalhes sobre como se definem as **etapas**, as **premiações**, os **campeões** e as **pontuações finais** das equipes nesta importante competição virtual de robôs autônomos inteligentes.

A competição acontecerá a partir do módulo de competições **sBotics Events** ([events.sbotics.net](https://events.sbotics.net)) do simulador sBotics.



*Captura de tela inicial do ambiente de competição Events.*

Somente os membros com a atribuição de **líder da equipe** importada do Sistema Olimpo poderão **gerar o código** e **executar as Rodadas** (explicados posteriormente), sendo permitida uma **única participação** no dia da competição em **cada Rodada**.

As datas das **Etapas Estaduais e Nacional** são definidas pela organização da OBR e serão divulgadas no portal do participante no site da OBR ([obr.robotcup.org.br](https://obr.robotcup.org.br)). É de responsabilidade do **professor responsável** pelas inscrições acompanhar o site da OBR e o Sistema Olimpo e verificar a data de participação de sua(s) equipe(s)"

## Equipe

As equipes são compostas pelos estudantes **já inscritos** na equipe que participou da etapa Modalidade Prática Virtual (Estadual), mantendo no mínimo 2 estudantes até um máximo de 4 estudantes, mais um professor ou técnico. Todos os estudantes devem pertencer a um, e somente um, dos níveis:

- **Nível 1:** Para estudantes do 1º ao 8º ano do Ensino Fundamental. Participantes da etapa estadual, **que avançaram** para a Etapa Nacional Virtual;
- **Nível 2:** Para estudantes do 8º e 9º ano do Ensino Fundamental e Ensino Médio ou Técnico. Participantes da etapa estadual, **que avançaram** para a Etapa Nacional Virtual.

Um estudante pode ser registrado em apenas **uma equipe** na etapa Nacional da **Modalidade Resgate Virtual Simulação**. A equipe deve estar relacionada a somente **um nível**.

*Os alunos membros das equipes devem estar **todos** cadastrados na **equipe importada do Sistema Olimpo** no sBotics Events, onde ao menos 1 (um) integrante deve ter o cargo de **Líder**, responsável por gerar o código de competição e executar a Rodada no simulador.*



**Não** será realizada a troca de integrantes das equipes que se classificarem para a Etapa Nacional Virtual. **Quaisquer divergências ou inconsistências** no cadastro dos membros inscritos originalmente no sistema Olimpo para o sistema de competição, acarretará na **desclassificação da equipe**.





## Trabalho em Equipe

A utilização de programações criadas por outras equipes **não é permitida**. Os mentores, tutores e professores do colégio ou da equipe devem sempre incentivar o cumprimento do desafio pelos estudantes e **não realizar as tarefas para eles** apenas com o intuito de ganhar a competição. Incentivamos o compartilhamento de programações entre equipes com o objetivo de disseminar o conhecimento, auxiliar equipes iniciantes, ensinar, mas não para que as programações sejam reutilizadas de forma parcial ou integral por outras equipes.

Destaca-se que cada equipe deve desenvolver sua **própria** programação, sendo a programação dos robôs sujeita à inspeção a **qualquer momento da competição**.



**Adultos** (mentores, técnicos, professores, pais, responsáveis) **não estão autorizados** a programar ou se envolver na programação dos robôs, enfim, fazer o trabalho pelos estudantes durante o período de preparo da competição. Os estudantes deverão ser capazes de estudar e programar apenas com a **mediação** de um adulto.



## Condições Gerais

Uma **Rodada** é uma **Rotina** (como descrita no **Caderno Verde**) no contexto de uma **competição**.

Serão realizadas **3 Rodadas** em **3 arenas diferentes** elaboradas pela organização e carregadas automaticamente pelo sistema de competições. Após três rodadas, a menor pontuação será descartada, e as outras duas **serão somadas e consideradas** para gerar a pontuação final de cada equipe de cada Nível.

Os 10 finalistas de cada Nível terão a sua Rodada adicional somada a pontuação anterior, a ser executada com dia e hora marcada, sendo transmitida em uma videochamada.

As arenas **podem e devem** mudar de configuração e de complexidade a cada rodada, sendo pelo menos 1 com complexidade baixa, fácil, e uma com complexidade alta, difícil.

**Cada Rodada será disponibilizada de acordo com os horários listados abaixo, não sendo possível a realização de uma rodada fora do período.**

|           |                                 |
|-----------|---------------------------------|
| 1ª Rodada | De <b>09:00</b> às <b>12:30</b> |
| 2ª Rodada | De <b>11:30</b> às <b>15:30</b> |
| 3ª Rodada | De <b>14:30</b> às <b>18:00</b> |

**LEMBRE-SE:** Para a equipe executar a programação, o horário de encerramento da **rodada** deve ser levado em conta. Não serão contabilizados os resultados cuja execução ultrapasse o horário final da **rodada** e/ou **competição**.

**SUGERE-SE:** Não deixar para executar cada **rodada** próximo aos finais dos períodos a fim de que nenhum imprevisto impacte no desempenho da sua equipe.



## Requisitos da(s) Plataforma(s) de Competição

---

É imprescindível para a participação, que ao menos um membro da equipe possua um computador **compatível com o simulador** e **acesso a internet** no dia de cada etapa da competição. Não havendo o acesso a esses requisitos, no dia da etapa, a equipe será considerada **desistente**.

Para a fase Nacional da Olimpíada Brasileira de Robótica, as equipes que já participaram e se classificaram nas etapas estaduais, serão **realocadas** dentro do sBotics Events para o evento Nacional respeitando o Nível de inscrição no Sistema Olimpo. Os respectivos técnicos serão contatados via **e-mail** cadastrado no Sistema Olimpo (mesmo e-mail para sistema Olimpo e sBotics Events) para completar os membros da equipe e suas atribuições caso ainda não tenha sido feito anteriormente. Esse cadastro será a oficialização da equipe no simulador para fins de participação, acesso ao sistema e contabilização da pontuação nas etapas da competição. **Ao menos um membro** deverá ter a atribuição de *líder* e deverá ser o responsável por gerar o código de competição e executar a rodada no simulador. **Não sendo permitido após qualquer alteração de nome ou membros da equipe.**



## Calibração Pré-Rodada

No dia da competição, os competidores poderão consultar através da plataforma de competição as condições de iluminação de cada *Rodada* para realização da calibração.

| RODADA    | LUMINOSIDADE |
|-----------|--------------|
| 1ª Rodada | 10:00        |
| 2ª Rodada | 11:30        |
| 3ª Rodada | 15:30        |

*Fragmento da página de competição  
no sistema sBotics Events*

Esta calibração deverá ser realizada pelo competidor e inserida na programação oficial que será executada pelo robô na *Rodada* em questão **antes de seu início**.

**Não** será permitida a realização de pré-mapeamento das arenas de competição ou no posicionamento da área de resgate, e assim como já descrito pelo **Caderno Verde** (na seção que trata da *Rotina*), após o início de cada *Rodada* **não** serão permitidas **modificações** na programação em execução no robô virtual.

## Carregando Rodadas

No dia e **períodos determinados da competição**, para execução de cada uma das três rodadas (**em seus respectivos horários, ver página 50**), um dos membros da equipe com a atribuição de **líder**, dada pelo professor tutor, deverá acessar o sistema sBotics Events, fazer login no sistema, clicar em competir, escolher a rodada e gerar o **código único da rodada** escolhida. Esse código é único e intransferível, caso a equipe perca o código da rodada, deverá se encaminhar ao sBotics Events e desistir da rodada, não sendo possível gerar um novo código de competição para aquela rodada.

The image shows two screenshots from the sBotics Events system. The top screenshot displays a form with two dropdown menus: 'Competição' (set to 'Competição Demonstração - OBR') and 'Rodada' (set to '1ª Rodada'). A dark blue button labeled 'COMPETIR' is located at the bottom right of this form. A red arrow points from this button to the second screenshot. The second screenshot shows a text input field containing a long alphanumeric code: '11225|7jwKmlR4B7a10K8z5iZJnGrZXIYny1K5Bwc8PHh'. A dark blue button labeled 'COPIAR' is positioned at the bottom right of this second form.

*Importante lembrar que, a calibração **independe** da geração de código, e o horário da rodada pode ser visualizada na página da competição como descrito pela seção de “Calibração Pré-Rodada”.*

## Executando Rodadas

---

Após a calibração e checagem do programa, o competidor, logado no simulador com uma conta de líder da equipe, deverá clicar no módulo de competições e **colar o código gerado no sBotics Events** e *aceitar os termos*.

Ao inserir o código de competição, uma tela de confirmação aparecerá informando o nome da competição, a rodada, condição de iluminação, tempo para ajustes, nome da equipe e o nome do programa que será executado pelo robô.

Após clicar em prosseguir **não será permitido realizar modificações no cenário ou no programa**. Será concedida à equipe o tempo de 30 segundos para clicar em iniciar rotina, caso o tempo se esgote a execução da rodada terá seu início automaticamente.

A pontuação será **automaticamente contabilizada** pelo simulador e ao término do desafio ou após desistência a pontuação será enviada para os servidores da OBR responsáveis por realizar a divulgação dos resultados. Os robôs terão um máximo de **8 minutos para completar a tarefa** por rodada. O tempo de cada rodada será contabilizado **automaticamente** pelo simulador. O cronômetro é contínuo, não havendo interrupção da contagem.

Após o período destinado à etapa não será mais possível **realizar a participação na rodada**, dessa forma, equipes que não realizarem a participação em alguma rodada ficarão com a pontuação zerada na mesma.

Caso haja algum problema de conexão com a internet ou falha com o computador após o início de alguma das rodadas a pontuação **não** será contabilizada e a rodada será descartada, **não** havendo a possibilidade de repetição ou reinício da rodada.

**LEMBRE-SE:** Para a equipe executar a programação, o horário de encerramento **da rodada** deve ser levado em conta. **Não** serão contabilizados os resultados cuja execução ultrapasse o horário final da **rodada** e/ou **competição**.



## Violações

---

É responsabilidade das equipes a verificação dos **pré-requisitos para participação**. Dessa forma, **não** será oferecidos **tempo extra** às equipes que tiverem **problemas em seus computadores ou conexão**. Caso seja detectada alguma tentativa de *burlar, danificar o simulador ou sistema web* a equipe será **desclassificada** do torneio.

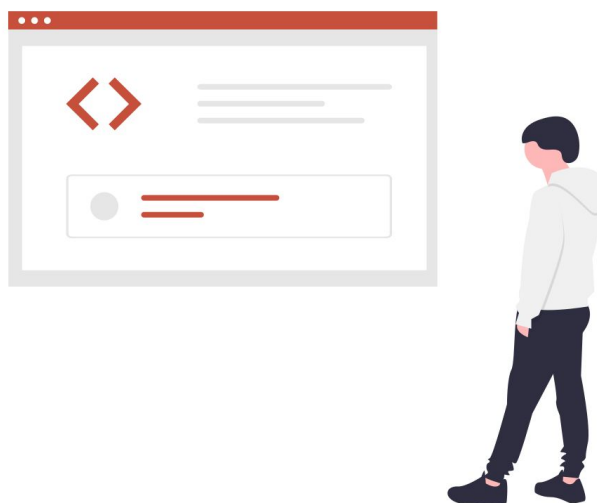


Os mentores, tutores e professores da equipe devem **sempre** incentivar o desenvolvimento da programação pelos estudantes e **não** realizar as tarefas para eles apenas com o intuito de *ganhar a competição*. Caso exista a interferência de **adulto** (pais, professores, ou outras pessoas estranhas ao grupo de alunos integrantes do time), **reutilização de programações** de outras equipes, ou **conduta inadequada**, as equipes serão sumariamente **desclassificadas** da competição. Outras possíveis violações também são descritas ao decorrer deste manual e caderno.

## Inspeção

---

As programações das equipes salvas no sistema **poderão** ser inspecionados pelos organizadores **antes, durante ou depois das Rodadas**, ou em qualquer momento que houver dúvidas quanto ao atendimento ou não das regras da competição.



Caso haja dúvidas quanto ao programa utilizado nas *Rodadas*, **todos os membros** (alunos e/ou técnicos) da equipe podem passar por questionamentos sobre propriedade e o funcionamento das suas programações através de uma **videochamada**, previamente agendada, realizada pelos organizadores. Estas entrevistas verificarão se, de fato, a programação utilizada é produto do trabalho **dos estudantes**.

Equipes que, sob **qualquer alegação**, ao serem convocados para a entrevista se **negarem** a participar, estão **passíveis de desclassificação** do torneio a critério do *comitê organizador nacional*.

A interferência **visível** (visual) de um adulto na programação da equipe poderá acarretar também na **desclassificação** da equipe.



## Critérios para Definir os Vencedores

---

Para indicar as equipes vencedoras em cada nível, as pontuações das rodadas deverão ser consideradas, sendo cada rodada realizada em uma arena diferente. Será declarada campeã a equipe que:

1. Possuir a **maior soma das duas maiores pontuações** obtidas considerando as 3 (três) rodadas (descarta-se a menor pontuação e soma-se as outras duas), **com a rodada final**.
2. Em caso de empate no item 1, o desempate será dado pelo menor tempo da rodada final.
3. Em caso de novo empate, o desempate será dado pela soma dos tempos **nas 3 (três) rodadas** realizadas. A equipe com a menor soma de tempo é a vencedora.
4. Em caso **de novo** empate, o desempate será dado pelo menor tempo obtido de qualquer **rodada de maior pontuação** da equipe. A que obteve o menor tempo, será a vencedora.
5. Se ainda persistir o empate, poderão ser usados como critério de desempate, a **maior pontuação obtida na rodada descartada**.
6. Se ainda persistir o empate, o desempate será dado pela maior pontuação obtida na primeira rodada. Mantendo o empate, a maior pontuação obtida na segunda rodada e mantendo, a maior pontuação terceira rodada. A equipe que obteve a maior pontuação seguindo a sequência de rodadas descritas, será a vencedora.
7. Caso todos os critérios descritos nos itens anteriores não resolvam o empate, uma regra a critério da Comissão Organizadora Nacional será utilizada.



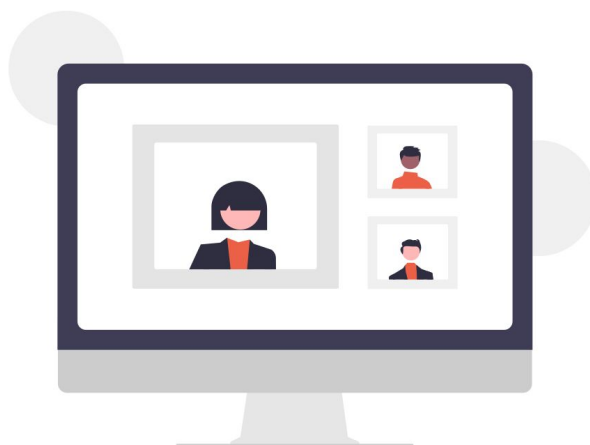


## Classificação

---

As equipes classificadas para a Etapa Final Nacional Virtual serão alocadas em ordem **decrecente da pontuação máxima**, por **nível**, sendo finalistas as **dez equipes por nível** com maior pontuação.

A pontuação da rodada final será somada a pontuação que o classificou entre os 10 finalistas.



As Rodadas da etapa final ocorrerão em uma **videoconferência** em horário marcado divulgado pela equipe organizadora. Caso a equipe execute a Rodada antes do seu horário marcado ou fora da videoconferência a Rodada será **desconsiderada e atribuída pontuação 0** com o tempo máximo.

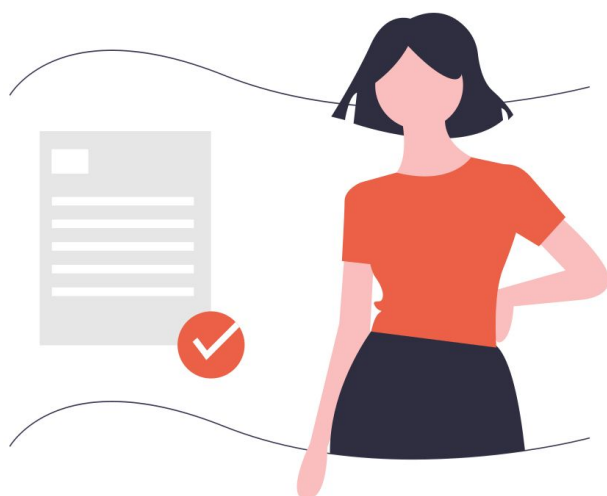


## Premiações

---

**Todas** as equipes classificadas para a Etapa Nacional Virtual recebem o Certificado de **Finalista Nacional Virtual**. As equipes que forem classificadas para Etapa Final Virtual e finalizem em uma das **3 (três) primeiras colocações** da competição receberão **Medalhas**, sendo que a 1ª receberá a medalha de **Ouro**, a 2ª receberá de **Prata** e a 3ª de **Bronze**.

Também serão concedidas **Bolsas de Iniciação Científica Júnior** para as equipes que alcançarem os 3 primeiros lugares do nível 1 e nível 2 e que atenderem aos **requisitos do CNPq** dispostos no **Portal do Participante**.



*Caderno Cinza*

# Mudanças



**Mudanças desde o último manual lançado, ideal para competidores com experiências anteriores.**

## Este Ano

---

- As vítimas vivas deverão ser salvas na área de resgate verde, e a morta na área de resgate vermelha. Vítimas mortas salvas antes das vivas não recebem multiplicadores. Vítimas salvas nas áreas de resgate invertidas não recebem multiplicadores.
- Caso haja falha de progresso entre dois checkpoints (ou um checkpoint e o ladrilho de chegada) com a sala de resgate entre eles, haverá um decréscimo no(s) multiplicador(es) de resgate.



# Versionamento

---

**1.0 - Setembro/2024**



*Caderno Laranja*

# Conflitos



**Fair Play, Esclarecimento das Regras, Circunstâncias  
Especiais e Recursos.**

## Introdução e Fair Play

---

Durante a competição podem surgir conflitos e desentendimentos que devem ser tratados sempre com **respeito mútuo** entre os participantes. Como a competição será realizada utilizando um **mesmo simulador** para todas as equipes, todas as pontuações finais serão as atribuídas diretamente pelo simulador. Caso haja alguma suspeita de manipulação dos resultados ou de que as programações não foram implementadas pelos competidores, a equipe, técnico responsável e/ou alunos, pode ser convocada para uma **videochamada** a fim de explicar o funcionamento do seu robô virtual.



## Esclarecimento das Regras

---

**Cada equipe** é responsável por verificar a versão mais recente das regras no site oficial da OBR **antes da competição**.

O esclarecimento das regras será feito pela Comissão de Arbitragem, ou, previamente, pela organização geral, através da lista de questões frequentes (FAQ). Pode-se também obter esclarecimentos pelo e-mail [obr.pratica@robocup.org.br](mailto:obr.pratica@robocup.org.br).

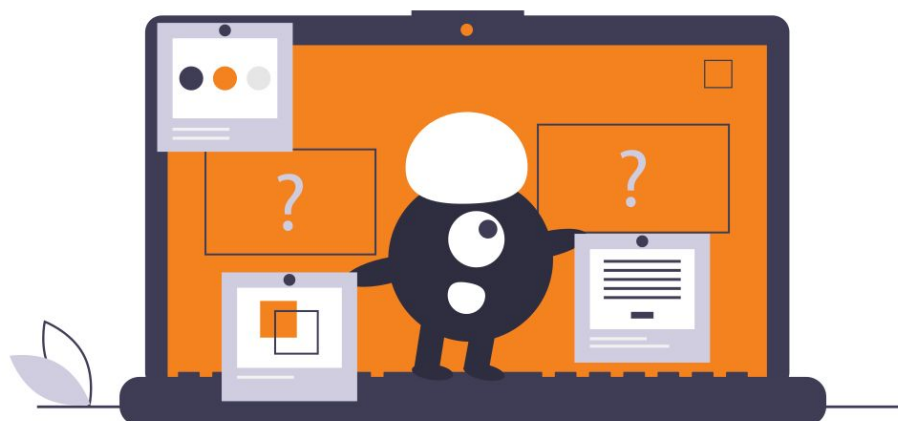




## Circunstâncias Especiais

---

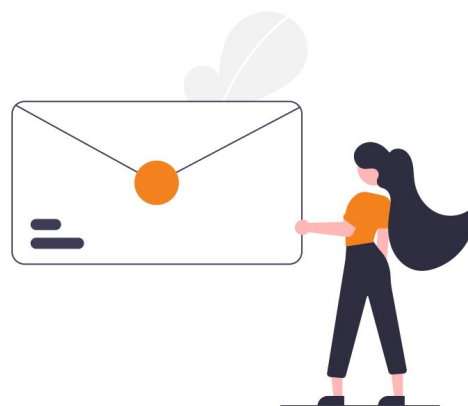
Modificações especiais nas regras para atender às circunstâncias especiais, tais como problemas não previstos e problemas e/ou capacidades dos times, podem ser acordadas **até o início do torneio**, cabendo, neste caso, concordância da organização da competição.



## Recursos

A equipe ou competidor que se sentir prejudicada(o) pelo funcionamento do simulador durante as etapas pode registrar detalhadamente **todas as informações** e enviar um e-mail com o título "**Recurso – Resgate Virtual Simulação – Etapa Nacional**", para [obr.pratica@robocup.org.br](mailto:obr.pratica@robocup.org.br).

Os e-mails com recursos poderão ser enviados no prazo **máximo de 1h** (uma hora) **após a finalização da etapa**. Decorrido o prazo estipulado, todas as equipes serão declaradas como de acordo com os resultados, nada mais havendo a reclamar.



Nota-se ainda que a OBR espera que seus competidores participem do evento com **respeito e cooperação**, lembrando sempre de nosso **Código de Conduta** e buscando acordos através de discussões de forma respeitosa e **amistosa** entre equipes e organização.



