

# <⚡> HackArena 2.0

## Zasady Hackathonu

Witajcie na HackArena 2.0!

Waszym zadaniem jest **napisanie programów (botów)** do stworzonej przez nas gry **MonoTanks**. Rywalizacja będzie się opierać na turniejowym drzewku zaczynając od ćwierćfinałów.

Na początku zostaliście podzieleni **4 grupy po 4 drużyny**. Każda grupa rozegra w trakcie turnieju **mecz ćwierćfinałowy**. W jednym meczu biorą udział **4 drużyny, czyli 4 boty**.

Mecz polega na rozegraniu **5 rund**. Z każdego meczu **2 najlepsze boty awansują** do następnego etapu. Przy czym w finale oczywiście **zwycięzca jest tylko jeden**, a kolejne drużyny otrzymają odpowiednio drugie, trzecie i czwarte miejsce. Po więcej informacji jak wygląda ocenianie botów w meczu proszę spojrzeć na rozdział System oceniania w meczu poniżej.

Aby **zwyciężyć w HackArena 2.0** wasz bot, będzie musiał nie tylko bezbłędnie **implementować podstawowe mechaniki**, ale także będzie musiał wykazać się **"myśleniem strategicznym"**. Pomyślcie o tym w trakcie implementacji waszego bota! Pamiętajcie też, że w trakcie meczu, może wydarzyć się wiele **sytuacji brzegowych** (tzw. edge case) - im więcej z nich wyłapiecie w trakcie tworzenia bota - tym mniejsza szansa, że coś go zaskoczy w trakcie trwania turnieju!

**Szczegółowe informacje** na temat gry, systemu oceniania oraz tego jak **zacząć pisać bota** znajdziecie poniżej w tym pliku. Zapoznajcie się z nimi dokładnie!

Miłej lektury ;)

## Instrukcja gry

Instrukcja zawiera jedynie **podstawowe informacje** na temat zasad gry i **nie opisuje wszystkich możliwych sytuacji**, które mogą wystąpić podczas rozgrywki. Przedstawione tu mechaniki mają na celu wprowadzenie w fundamenty gry, jednak pełne zrozumienie synergii i nietypowych przypadków **wymaga samodzielnego eksperymentowania**. Zachęcamy do dalszego testowania i poznawania złożoności gry, gdyż może mieć istotny wpływ na Wasze osiągnięcia i wynik rozgrywki.

## Wstęp

MonoTanks to wieloosobowa gra od 2 do 4 graczy, gdzie każdy gra przeciwko każdemu. Polega ona na zdobywaniu punktów za zadawane obrażenia oraz przejmowanie stref.

## Mechaniki

### Czołgi

Czołg to jednostka w grze, którą steruje bot. Ma on 100 punktów życia. Ma on broń podstawową, oraz może mieć w ekwipunku maksymalnie jedną broń specjalną.

Po otrzymaniu od serwera informacji o stanie gry, bot może wykonać jedną z następujących akcji:

1. Obrót wieży o 90 stopni w lewo, w prawo lub nie obracać się i jednocześnie obrót czołgu o 90 stopni w lewo lub w prawo. (Obrót wieży nie wpływa na obrót czołgu ani na odwrót)
2. Poruszenie się o jedną kratkę w przód lub w tył w kierunku wskazywanym przez podwozie czołgu.
3. Strzał z broni podstawowej lub użycie broni specjalnej.
4. Czekanie (nie robienie nic).

## <⚡> HackArena 2.0

### Mgła wojny

Bot widzi tylko pola, które znajdują się w stożku 144 stopni przed czołgiem. Nie ma ograniczenia odległości. Dodatkowo bot widzi pola w linii prostej przed wieżą czołgu. Wszystkie inne pola zakrywane są przez mgłę wojny. Mgła wojny nie dotyczy za to ścian oraz stref. Oczywiście w strefie, która jest w mgle wojny, nadal nie widać czołgów innych botów.

### Bronie

#### Pocisk podstawowy

Leci on z prędkością 2 kratek na tick i zadaje 20 obrażeń przy trafieniu. Amunicja ta regeneruje się pasywnie z prędkością 1 pocisk na 10 ticków. Czołg może przenosić maksymalnie 3 sztuki tego pocisku. Pociski te przy zderzeniu ze sobą anihilują się wzajemnie i po prostu znikają.

#### Bronie specjalne

Czołg nie posiada domyślnie przy sobie żadnej broni specjalnej. Aby ją zdobyć należy znaleźć ją na mapie i najechać na nią. Wtedy trafi ona do ekwipunku czołgu. Czołg może przenosić maksymalnie jedną broń specjalną, a w przypadku gdy już ją posiada i najedzie na kolejną broń specjalną, to nowej broni po prostu nie można będzie wziąć dopóki nie wykorzysta tej którą posiada.

#### Dubeltówka

Jest to broń specjalna, która wystrzeliwuje dwa równoległe pociski jednocześnie. Zadają one 2 razy więcej obrażeń niż pocisk podstawowy, czyli 40 obrażeń przy trafieniu.

#### Mina

Gdy bot użyje miny, jest ona stawiana na ziemi jedną kratkę za czołgiem. Mina zadaje 50 obrażeń gdy ktokolwiek na nią wjedzie.

#### Laser

Laser jest najsilniejszą bronią w grze. Jego użycie oznacza natychmiastowe rażenie wszystkiego co znajduje się na jego drodze,

## <⚡> HackArena 2.0

dopóki nie napotka ściany. Jego użycie trwa 10 ticków, to oznacza, że strzelec jest unieruchomiony na 10 ticków i nie może podjąć żadnej innej akcji. Zadaje on 80 obrażeń przy trafieniu w każdym ticku.

### Radar

Użycie radaru powoduje, że w następnym ticku widzimy jednorazowo całą mapę pomimo mgły wojny, a w szczególności czołgi wszystkich botów oraz bronie specjalne, które znajdują się na mapie.

### Punkty za obrażenia

Boty otrzymują punkty za zadawane obrażenia innym botom. Otrzymują one tyle punktów, ile obrażeń zadały. Przykładowo, jeśli przeciwnik posiada 20 punktów życia i został zniszczony przez Wasz bot za pomocą lasera to bot otrzyma 20 punktów.

### Tabela broni

Broń	Obrażenia (Zdobyte punkty)	Prędkość (kratka/tick)	Czas strzelania (tick)
Pocisk podstawowy	20 (10)	2	1
Dubeltówka	40 (20)	2	1
Laser	80 (80)	inf	10
Mina	50 (50)	N/A	1

### Strefy

Na mapie są zawsze 2 strefy. Każda strefa może być w jednym z 5 stanów:

1. Neutral - żaden bot nie posiada tej strefy.

## <⚡> HackArena 2.0

2. BeingCaptured - strefa jest właśnie przejmowana przez jakiegoś bota. To znaczy w strefie znajduje się tylko jeden bot i rośnie jego posiadanie strefy lub bot przejmujący wyszedł z niej i jego posiadanie się zmniejsza. Przejęcie strefy trwa 50 ticków.
3. Captured - strefa jest przejęta przez kogoś. Tylko w tym stanie bot dostaje punkty za posiadanie strefy. 0.5 punktów na tick.
4. BeingContested - w strefie znajduje się więcej niż jeden bot. Strefa nie jest przez nikogo przejmowana, dopóki jest więcej niż jeden bot w niej. Jeśli ktokolwiek przejął wcześniej strefę, to w tym stanie bot nie dostaje punktów za posiadanie strefy.
5. BeingRetaken - strefa została przejęta przez innego bota wcześniej. Teraz jest przejmowana przez innego bota. Przejęcie strefy trwa 50 ticków. \*W tym stanie poprzedni bot posiadający strefę też nie dostaje punktów za posiadanie strefy.\*

### Odrodzenie się

Czołg po zniszczeniu znika z mapy i zostanie na nią zrzucony po 50 tickach.

## Informacje Techniczne

Aby mieć pewność, że Wasze boty **będą działać poprawnie** na naszym serwerze, możecie w każdej chwili nas poprosić, abyśmy **przetestowali Waszą obecną wersję** na systemie sprawdzającym, czy na pewno wszystko się **kompiluje, buduje, uruchamia** i czy wyniki są w porządku. Jeśli na koniec hackathonu Wasz bot **nie będzie działał poprawnie** (nie zbuduje się obraz dockera, nie uruchomi się, będzie się crashował), to **Wasz bot zostanie zdyskwalifikowany** i turniej rozegra się bez niego.

## Jak zacząć pisać bota?

Dla każdego języka programowania będzie to wyglądało trochę inaczej. Ale generalnie będzie to wyglądało tak:

1. Potrzebujecie uruchomić lokalnie serwer gry, więc w pierwszej kolejności wchodzicie na repozytorium <https://github.com/INIT-SGGW/HackArena2024H2-Game> i tam w README znajdziecie, jak uruchamiać serwer na kilka sposobów.
2. Następnie wchodzicie na repozytorium klienta w języku, jakim chcecie pisać, i czytacie jego README, jak go uruchomić oraz quick start - jak pisać bota.

# <⚡> HackArena 2.0

## Dostępne języki programowania

1. Python <https://github.com/INIT-SGGW/HackArena2024H2-Python>
2. Java <https://github.com/INIT-SGGW/HackArena2024H2-Java>
3. JavaScript <https://github.com/INIT-SGGW/HackArena2024H2-JS>
4. TypeScript <https://github.com/INIT-SGGW/HackArena2024H2-TS>
5. C# <https://github.com/INIT-SGGW/HackArena2024H2-CSharp>
6. C++ <https://github.com/INIT-SGGW/HackArena2024H2-Cxx>
7. Rust <https://github.com/INIT-SGGW/HackArena2024H2-Rust>
8. GO <https://github.com/INIT-SGGW/HackArena2024H2-Go>

## Architektura systemu

Stosujemy klasyczną architekturę klient-serwer, dzięki czemu możecie pisać bota w dowolnym języku programowania. Komunikacja sieciowa odbywa się przy użyciu WebSocketów.

- Serwer - Zajmuje się on zarówno sieciowym aspektem połączenia ze sobą klientów (graczy i botów), jak i logiką gry.
- Klient - Może to być klient GUI za pomocą którego gracz może obserwować rozgrywkę lub samemu grać, albo też klient-bot, którego wy piszecie.

## Jak będziemy testować Wasze boty?

1. Poprosimy was o spakowanie zmodyfikowanego repozytorium do pliku .zip. Oczywiście prosimy o usunięcie wszystkich plików i folderów tymczasowych zawierających zbudowane .exe czy .jar, czy node\_modules, aby plik zip nie był zbyt duży.
2. Po zebraniu wszystkich paczek zip od was, przerzucimy je na nasz serwer.
3. Napisaliśmy skrypt w języku Python wraz z biblioteką Docker SDK for Python, który w pierwszej kolejności pozwoli nam na zbudowanie obrazów dockera dla każdego z Waszych botów. Dlatego ważne jest, abyście nie modyfikowali żadnych plików poza wskazanymi plikami i folderami w README każdego z repozytoriów.

## <⚡> HackArena 2.0

4. Następnie skrypt uruchomi odpowiednio kontener z serwerem, jak i klientami (Wasze boty), aby rozegrać mecze. I tak podobnie dla całej drabinki turniejowej.

W ten sposób przeprowadzimy symulację turnieju, rozgrywając dla każdego meczu 5 rund. Pozwoli nam to sprawdzić, który z Waszych botów jest najlepszy.

## FAQ

1. Czy mogę używać zewnętrznych bibliotek/narzędzi do pisania botów?

→ W 99% przypadków tak. Jeśli biblioteka nie ma efektów ubocznych, które wpływałyby na system operacyjny lub networking, lub wymagałyby korzystania z karty graficznej czy inne tego typu drastyczne rzeczy, to powinno być wszystko w porządku. Wszelkie proste biblioteki do analizy danych czy przyjemniejszego przetwarzania tablic to pewnie. Oczywiście nie wolno usuwać ani zmieniać wersji obecnie wykorzystywanych przez repozytoria bibliotek, gdyż może to wpłynąć na kompatybilność z naszym serwerem i sprawdzaniem botów.

2. Czy bot może wykonywać zapytania do zewnętrznych serwerów w trakcie gry?

→ Nie, nie może. W środowisku dockera na serwerze kontenery nie będą miały dostępu do internetu.



## <⚡> HackArena 2.0

3. Czy mogę w repozytorium umieścić pliki na przykład wag sieci neuronowych?

→ Tak, powinno być wszystko w porządku. W każdym z API wrapperów w README jest opisane, jak to zrobić.

4. Chciałbym napisać bota z użyciem tensorflow i użyć karty graficznej. Czy mój bot podczas sprawdzania będzie mógł korzystać z karty graficznej?

→ Nie, nie będzie mógł. Boty uruchamiamy jako kontenery dockera na serwerze, który nie ma żadnej karty graficznej.

5. Ile pamięci RAM będzie dostępne mojemu botowi podczas gry?

→ Dostępny będzie 1GB RAMu na jednego bota.

6. Ile rdzeni procesora będzie dostępne mojemu botowi podczas gry?

→ Dostępny będzie 1 rdzeń procesora na jednego bota.

7. Jak duże moje repozytorium może być i jak dużo przestrzeni dyskowej będzie dostępne dla mojego bota?

→ Im mniejsze tym lepsze. Aczkolwiek jeśli ktoś potrzebuje dodać pliki statyczne to maksymalnie repozytorium po rozpakowaniu może mieć 1GB.

## <⚡> HackArena 2.0

8. Czy mogę uruchamiać asynchroniczne funkcje w moim rozwiązaniu?

→ Zależy od naszej implementacji konkretnego klienta. Niektóre na to pozwalają, niektóre nie. Musicie przeczytać README.

## System oceniania

Turniej finałowy HackArena 2.0 zostanie przesymulowany automatycznie i botom zostaną przydzielone punkty zgodnie z poniższymi zasadami.

Aby zagwarantować jak najbardziej rzetelne wyniki, każdy mecz zostanie rozegrany w 5 rundach, a sumaryczne wyniki zadecydują o kolejności drużyn w meczu. Tak więc mówiąc **"mecz"** mamy na myśli pięciokrotnie przesymulowane rozgrywki. Taka pojedyncza symulacja to właśnie **"runda"** w meczu.

Każdy z 4 botów (A, B, C, D) grających mecz z wieloma rundami otrzymuje punkty na podstawie ostatecznych wyników oraz ilości zabójstw jakie zrobił dany bot w danej rundzie.

Poprzez **"wynik w rundzie"** rozumiemy ilość punktów zdobytych w grze za zadawanie obrażeń oraz przejmowanie stref. A poprzez **"punkty"** rozumiemy zbierane przez boty punkty zbierane w całym meczu za zajęte miejsca w rundach.

- 1. miejsce (783 wyniku i 3 zabicia) otrzymuje 3 punkty
- 2. miejsce (457 wyniku i 2 zabicia) otrzymuje 2 punkty
- 3. miejsce (457 wyniku i 1 zabicia) otrzymuje 1 punkt
- 4. miejsce (122 wyniku i 3 zabicia) otrzymuje 0 punktów

### Przykład:

Założmy dla uproszczenia, że boty grają 3 rundy (w rzeczywistości będzie ich 5). Poniżej przedstawiono, jak przyznawane są punkty po każdej rundzie.

#### Wyniki Rundy 1:

- 1. miejsce: Bot A (3 punkty)
- 2. miejsce: Bot C (2 punkty)
- 3. miejsce: Bot D (1 punkt)
- 4. miejsce: Bot B (0 punktów)

## <⚡> HackArena 2.0

### Wyniki Rundy 2:

- 1. miejsce: Bot B (3 punkty)
- 2. miejsce: Bot A (2 punkty)
- 3. miejsce: Bot C (1 punkt)
- 4. miejsce: Bot D (0 punktów)

### Wyniki Rundy 3:

- 1. miejsce: Bot C (3 punkty)
- 2. miejsce: Bot D (2 punkty)
- 3. miejsce: Bot B (1 punkt)
- 4. miejsce: Bot A (0 punktów)

### Ostateczna Tabela Wyników:

- Bot A: 3 (Runda 1) + 2 (Runda 2) + 0 (Runda 3) = 5 punktów
- Bot B: 0 (Runda 1) + 3 (Runda 2) + 1 (Runda 3) = 4 punkty
- Bot C: 2 (Runda 1) + 1 (Runda 2) + 3 (Runda 3) = 6 punktów
- Bot D: 1 (Runda 1) + 0 (Runda 2) + 2 (Runda 3) = 3 punkty

Po rozegraniu wszystkich rund, Bot C ma najwięcej punktów i zostaje ogólnym zwycięzcą meczu. Drugie miejsce zajmuje Bot A. Te 2 boty awansują do kolejnego etapu.

## Przypadki szczególne

### 1. Remisy

W przypadku, gdy dwóch lub więcej botów będzie miało taki sam wynik w rundzie (mało prawdopodobne, ale jednak), boty z tym samym wynikiem otrzymują punkty za miejsce wyższe. Na przykład, jeśli Bot A i Bot B zajmą 1. miejsce w rundzie, obaj dostaną po 3 punkty, a kolejny bot zostanie sklasyfikowany na 3. miejscu i otrzymuje 1 punkt (ponieważ 2. miejsce jest pomijane). Zapewnia to sprawiedliwy podział punktów w przypadku remisów, a boty są jednakowo nagradzani za podobne wyniki. W przypadku remisu na 2. miejscu, obaj boty otrzymują po 2 punkty, a 4. bot automatycznie otrzymuje 0 punktów.

## <⚡> HackArena 2.0

### 2. Bot, który nic nie zrobił

Jeśli jakikolwiek bot nie robi nic w rundzie – to znaczy, że ma wynik 0 z rundy, innymi słowy zgromadzi 0 punktów w grze – taki bot automatycznie otrzymuje 0 punktów, niezależnie od zajmowanej pozycji. Taka sytuacja mogłaby mieć miejsce tylko gdy jest kilku botów, którzy by mieli wynik 0 z rundy.

### 3. Trzech botów ex aequo w meczu po wszystkich rundach (tyle samo punktów i zabić)

W takim przypadku, rozgrywamy dodatkowe rundy do momentu, gdy dwóch botów będzie ewidentnymi zwycięzcami. A w przypadku finału do momentu gdy wszystkie 4 miejsca będą jednoznaczne.

### 4. W meczu uczestniczy mniej niż 4 boty

W tym przypadku jedyne co się zmienia to punktacja. W przypadku trzech botów pierwsze miejsce otrzymuje 2 punkty, drugie 1 punkt, a trzecie 0 punktów. W przypadku dwóch botów pierwsze miejsce otrzymuje 1 punkt, drugie 0 punktów.

W przypadku gdy w meczu jest tylko jeden bot (jego przeciwnicy nie dojechali :P), mecz nie jest rozegrany. I bot automatycznie awansuje do następnego etapu.



## Powodzenia!

W przypadku jakichkolwiek problemów zgłaszajcie się do organizatorów HackArena 2.0 - z pewnością pomożemy!

Życzymy dobrej zabawy i zwycięstwa w drugiej edycji hackathonu HackArena!

Zespół HackArena  
Członkowie KN init