

Zadanie

Zaprojektuj i wykonaj środowisko gier turowych, wykorzystujących algorytmy heurystyczne.

Projektowana gra ma być grą turową typu CTF (*capture the flag*) zlokalizowaną w dwuwymiarowym świecie, o wielkości min. 100x100 pól. Rywalizacja ma miejsce pomiędzy dwiema drużynami, reprezentowanymi przez boty. Zadaniem bota jest dotarcie do bazy przeciwnika, zabranie flagi i powrót z nią do swojej bazy.

Gra ma odbywać się w turach. W każdej turze boty przesuwają się tylko o jedno pole w dowolnym kierunku mapy.

Mapa ma składać się z:

- przeszkód statycznych, znanych z góry każdemu z botów:
 - blokujących (ściany), nie pozwalających na postawienie w nich bota;
 - nieblokujących (pusta przestrzeń) pozwalających na postawienie w nich bota;
 - o zadanym stopniu trudności (np. 1..5) do przejścia których bot potrzebuje pewną liczbę tur. Np. trudność = 3 oznacza, że bot drużyny niebieskich może wyjść z pola po trzeciej turze od wejścia.
 - *Fog of war*; widoczność w promieniu 5 pól
- przeszkód dynamicznych, nie podawanych z góry botom (jako mapy), np.:
 - mina, ustawiana przez serwer losowo w pustych miejscach
 - bot dowiaduje się o minie dzięki informacji odległościowej, np.: W zasięgu są 3 miny (5 pól, 4 pola, 2 pola).
 - wejście na minę bądź wspólna krawędź powoduje detonację i zniszczenie bota
 - bot przeciwnika – podobnie jak w przypadku min, informacja o obecności w pobliżu przekazywana jest jako odległości: W zasięgu znajduje się bot przeciwnika (3 pola).
 - wejście na pole bota przeciwnika bądź wspólna krawędź powoduje zniszczenie bota. W sytuacji, gdy bot A stoi a bot B wchodzi w sąsiedztwo (krawędź wspólną) wtedy ginie bot B.
 - Informacja o przeszkodach dynamicznych dostarczana jest botowi jedynie, gdy ten się do nich zbliży na odległość np. 5 pól, np.: w zasięgu 5 pól są 3 miny i 1 bot.
- Po zniszczeniu danego bota pojawia się nowy, w bazie jego drużyny (respawn). W przypadku, gdy zniszczono bota niosącego flagę, flaga pozostaje w miejscu finalnego ataku.

Architektura:

Grę należy zrealizować jako system klient-serwer. Serwer odpowiada za przechowywanie mapy oraz wykonywanie ruchów w ramach tury, podczas gdy klient = bot.

Serwer ma zarządzać mapą w taki sposób, aby niemożliwe było oszukiwanie:

- nie pozwala na przeszkakiwanie o więcej niż jedno pole w turze,

- nie przesyła informacji o wszystkich przeszkodach dynamicznych, a tylko tych w zasięgu (np. 10 pól),
- wyświetla liczbę tur i zniszczonych botów
- wykonuje wizualizację (możliwe rozróżnienie wizualne poszczególnych elementów mapy),
- monitoruje pozycję flag obu drużyn,
- określa kiedy wybucha mina oraz kiedy giną boty
- pozwala na określenie szybkości animacji gry oraz grę krok po kroku.

Implementacja:

Zalecany protokół tekstowy, oparty o TCP/IP. Język programowania serwera oraz klientów: dowolny, mogą być różne.

Zasady oceniania:

- Na ostatnich zajęciach przeprowadzony zostanie turniej, gdzie rywalizują wszystkie drużyny na zasadzie każdy z każdym. Rywalizacja to pięć kolejnych gier danej pary botów. Rywalizację wygrywa ta drużyna, która straciła najmniej botów we wszystkich pięciu grach. Zwycięska drużyna otrzymuje 5,0; II miejsce: 4,5; III i dalej: 4,0.
- Oddanie niedziałającego bota, w sensie poprawności komunikacji z serwerem, powoduje niezaliczenie przedmiotu.
- Prowadzący wymaga zdawania cotygodniowych raportów z postępów w pracach wszystkich zespołów.