

# NJUTON

## GRA NA ZAWODY PIZZA 2013

### 1 Historyjka

Nie tak dawno temu, w sąsiedniej galaktyce, ojciec chrzestny Międzyplanetarnego Podziemia Gastronomicznego Han Pizzolo chciał przemycić wyjątkowo cenny i pożądanym również przez jego wrogów towar.

Kilka miesięcy wcześniej podpisał umowę z siecią przydrożnych barów znajdujących się przy głównej trasie międzyplanetarnej z Salamino do Capriciosso, w której zobowiązał się do dostarczenia bardzo rzadkich i drogich składników, z których wykonuje się Najlepszą Pizzę w Galaktyce. Cały interes nie jest zbyt legalny, bowiem nasz zdolny przedsiębiorca chciał uniknąć opłacenia obowiązujących w galaktyce podatków i przewieźć towary, pozostając niezauważonym. Niestety, nie jest jedynym przebiegłym cwaniakiem w tym rejonie Wszechświata, a inni wcale nie boją się go tak bardzo, jak zdaje się to jemu samemu. Postanawiają więc pokrzyżować mu plany i przejąć cenny ładunek z tajemniczymi składnikami.

Na domiar złego, atakują go w bardzo niefortunnym miejscu, a mianowicie w pobliżu gwiazdy Carbonaro o wyjątkowo silnej grawitacji, która utrudnia ucieczkę i walkę.

Jeśli nie chcesz, aby wrogowie Hana zaprzepaścili jego interes życia oraz odebrali galaktyce jeden z najbardziej poświadanych przysmaków, musisz mu pomóc!

### 2 Ogólny opis mechaniki gry

Gra przypomina znaną grę Asteroids – każdy z graczy ma jeden statek kosmiczny, który przy pomocy lasera i działka ma za zadanie zniszczyć statki wrogów. Zadanie utrudnia gwiazda umieszczona na środku planszy, która swoim polem grawitacyjnym ściąga statki w śmiertelną pułapkę.

Liczba punktów zależy liniowo od czasu jaki przeżyje statek licząc od początku rundy.

### 3 Opis wszechświata

Krawędź wszechświata wyznacza kwadrat o lewym dolnym rogu w punkcie  $(-100, -100)$  i wymiarach 200 na 200. Na środku planszy (w punkcie  $(0, 0)$ ) znajduje się gwiazda. Gracze rozpoczynają grę na pozycjach równoodległych od środka układu współrzędnych w równych oględnościach kątowych między sobą.

Wszechświat jest torusem, w tym sensie, że ma “zawinięte krawędzie”. Obiekty wylatujące poza prawą krawędź pojawiają się z lewej strony itd.

## 4 Opis statku

Statek (jak wszystkie obiekty w grze) jest kulą o stałej gęstości. Na pokładzie zamontowano laser oraz działko. Każdy statek ma również wyróżniony *kierunek* w którym leci oraz strzela, oraz *punkty życia*, początkowe równe 1, spadające przy kolizji z pociskiem.

Gracz może nakazać statkowi ustawić daną moc silnika, wystrzelić laserem bądź działkiem i obracać się (patrz komenda SET).

## 5 Opis fizyki

### 5.1 Grawitacja

Statki i pociski wystrzelone z działka oddziałują grawitacyjnie z gwiazdą zgodnie ze wzorem:

$$\mathbf{F} = \frac{Mm\mathbf{r}}{|\mathbf{r}|^3},$$

gdzie  $\mathbf{r}$  jest wektorem położenia ciała,  $m$  jego masą a  $M$  masą gwiazdy.

### 5.2 Kolizje

Każda kolizja statku z gwiazdą kończy się zniszczeniem statku. Kolizja statku z dowolnym typem pocisku kończy się zniszczeniem pocisku i zmniejszeniem ilości punktów życia statku (i jego zniszczeniem, gdy spadnie ona do 0 lub niżej). Kolizja statku z innym statkiem kończy się zniszczeniem obu. Nie występują kolizje między pociskami.

Uwaga do kolizji pocisków ze statkiem: ze względu na pole siłowe statku pocisk przelatujący daleko od środka masy może nie trafić w statek. Pociski przelatujące bliżej środka masy statku mają większe szanse na trafienie.

## 6 Tury i rundy

Rozgrywka jest podzielona na tury i rundy. Runda trwa pojedyncze minuty i składa się z wielu tur trwających około  $\frac{1}{30}$  sekundy. Gracz może wydać komendy swojemu statkowi przy początku każdej tury. Po zakończeniu czasu trwania rundy (lub zniszczeniu wszystkich statków) jest uruchamiana nowa runda.

## 7 Komendy

Każda komenda może zakończyć się błędem ERR 404 no ship left, gdy gracz pozostał bez statku.

## 8 Limit prędkości

Prędkość światła wyznacza maksymalną prędkość statku gracza. Prędkość lasera jest stała (równa prędkości światła). Prędkość pocisku z działka jest stała i wiadoma (patrz komenda `CONSTANTS`).

### 8.1 WAIT

Komenda `WAIT` zwraca natychmiastowo jedną linię z napisem “OK”. Kolejna linia jest wysyłana na początku nowej tury. Ma ona format `OK <typ_kroku> <status_statku>`, gdzie:

- `<typ_kroku>` to `NEXT_TURN` lub `NEXT_ROUND` odpowiednio w przypadku nowej tury i nowej rundy,
- `<status_statku>` to `ALIVE` lub `DEAD` odpowiednio w przypadku posiadania i nieposiadania statku.

Jeżeli gracz posiada niezniszczony statek, pojawią się kolejne linie opisujące stan wszechświata. W pierwszej linii znajduje się opis statku gracza, kolejno: współrzędna  $x$ , współrzędna  $y$ , obrót w radianach (0 stopni to kierunek zgodnie z osią  $x$ ) i ilość punktów życia.

W kolejnej linii podana jest ilość innych statków na planszy  $n$ , a następnie  $n$  linii z opisem statków wg formatu podanego powyżej.

W kolejnej linii podana jest ilość pocisków na planszy  $m$ , a następnie  $m$  linii zawiera opis pocisku, czyli dwie liczby oznaczające kolejno współrzędne  $x$  i  $y$ .

### 8.2 CONSTANTS

Zwraca stałe fizyczne obowiązujące w bieżącej rundzie. W jednej linii zwraca następujące stałe oddzielone spacjami:

- `dt` - krok czasowy na jedną turę,
- `rotationSpeed` - prędkość obrotowa statku (w radianach na sekundę),
- `missileSpeed` - prędkość pocisku wystrzelonego z działka (w jednostkach odległości na sekundę),
- `engineMaxForce` - moc odsilnikowa działająca na statek przy pełnym ciągu silnika,
- `projectileLifeTime` - czas po jakim pocisk znika z planszy (w sekundach),
- `shipMass` - masa statku,
- `missileMass` - masa pocisku wystrzelonego z działka,

- `shipRadius` - promień statku,
- `missileDamage` - ilość punktów życia jaką utraci statek po kolizji z pociskiem z działka,
- `laserDamage` - ilość punktów życia jaką utraci statek po kolizji z laserem,
- `roundTime` - czas bieżącej rundy (w sekundach),
- `starMass` - masa gwiazdy,
- `starRadius` - promień gwiazdy (w jednostkach takich jakie powinny być),
- `projectileLimit` - limit pocisków istniejących jednocześnie na planszy (na jednego gracza),
- `lightSpeed` - prędkość światła (w jednostkach odległości na sekundę).

### 8.3 ROUND\_STATUS

Zwraca w jednej linii numer bieżącej rundy oraz informację ile czasu ona już trwa (w sekundach). Przykład:

```
> ROUND_STATUS  
< OK  
< 10 1.56
```

### 8.4 SET

Komenda `SET` pozwala wydać komendy załodze statku na następny krok czasowy: ciąg silnika (liczba zmiennoprzecinkowa od 0 do 1), komendę wystrzelenia pocisku i komendę obrotu.

Składnia komendy: `SET <przyspieszenie> <komenda_obrotu> <komenda_strzału>`, gdzie:

- `<przyspieszenie>` jest liczbą z przedziału  $[0, 1]$ ,
- `<komenda_obrotu>` jest liczbą 0, 1 lub 2: 0 oznacza brak obrotu, 1 obrót przeciwnie do ruchu wskazówek zegara a 2 zgodnie z ruchem wskazówek zegara,
- `<komenda_strzału>` jest liczbą 0, 1 lub 2: 0 oznacza brak strzału, 1 oznacza strzał laserem a 2 strzał działkiem.

Możliwe błędy:

- `ERR 403 too many projectiles` - przekroczono limit pocisków na planszy