

Szymon Majkut

## "Golce" – kooperacyjna sieciowa gra labiryntowa

### Opis

Celem projektu jest zaprojektowanie oraz zaimplementowanie gry sieciowej opartej na architekturze klient-serwer, w której jeden lub więcej graczy, używając dedykowanej aplikacji klienckiej łączy się do wspólnej aplikacji serwerowej, odpowiedzialnej za synchronizację i wymianę danych. Tematem przewodnim samej rozgrywki będzie grywalne zasymulowanie cyklu życia kolonii gołców piaskowych – niewielkich ssaków z rzędu gryzoni, żyjących w kilkunastoosobowych koloniach, z podziałem na kasty oraz dosyć długim czasem życia, jak na zwierzęta tego rozmiaru.

Bardzo istotnym elementem życia gołców piaskowych jest struktura eusocjalna ich kolonii, występuje tam tylko jedna samica, odpowiedzialna za rozród (tak zwana królowa), pozostałe osobniki zajmują się pozyskiwaniem żywności, budują tunele oraz zdobywają żywność. W projekcie gry, gracze wcielają się z osobnika niższej kasty, który dla dobra całej kolonii, musi na bieżąco dostarczać żywność, razem z innymi graczami, tak aby kolonia była w stanie przetrwać jak najdłużej. Z założenia gracze poruszają się po przygotowanych już wcześniej tunelach, które na początku każdej rozgrywki zostają wygenerowane losowo, wraz z ustaleniem jednego punktu centralnego – głównej komory, do której należy sprowadzać żywność dla kolonii.

Golce piaskowe przystosowały się do życia w warunkach ciemnych i wąskich tuneli, poprzez zatracenie zmysłu wzroku, ale znaczne wyczulenie pozostałych zmysłów. Są w stanie precyzyjnie odbierać i reagować na wszystko, co dzieje się w tunelach, rozpoznając zagrożenia po zapachu lub dźwięku, ta cecha będzie stanowiła ważny element rozgrywki, gracz będzie znał precyzyjnie tylko najbliższe otoczenie, natomiast pozostała część informacji będzie pojawiała się wokół podróżującego golca w pewnej odległości od niego i zawsze będzie stanowiła sygnał zdobyty innym zmysłem.

Gracze będą mogli wpływać na otoczenie przez zamknięty zestaw komunikatów - dźwięków, symulujących nieustanne chrząkania gołców, dzięki temu będą w stanie nawigować się nawzajem w tunelach. Dodatkowo każdy golec pozostawia ślad zapachowy w tunelu, dzięki czemu gracz jest w stanie określić czy znajduje się jeszcze blisko miejsca, w którym istnieje szansa spotkania jakiegoś golca czy może zabłądził gdzieś daleko od komory głównej.

Głównym zadaniem każdego gracza jest przynoszenie do komory głównej pożywienia – głównie korzenie oraz bulwy roślin, które mogą pojawiać się w tunelu w losowych momentach, ze zmiennym prawdopodobieństwem, im dalej dana odnoga tunelu znajduje się od komory głównej, tym większa szansa znalezienia pożywienia.

### Komunikacja

Po uruchomieniu serwera oraz aplikacji klienckiej każdy z graczy poprzez aplikację kliencką podaje namiary na serwer oraz port i wybiera pseudonim. W pierwszym etapie

serwer oczekuje na zgłoszenia graczy oraz otrzymanie komunikatu o rozpoczęciu rozgrywki.

Gdy serwer otrzymuje polecenie rozpoczęcia nowej gry, następuje generacja labiryntu oraz początkowe umiejscowienie graczy w komorze głównej. Serwer przez cały czas trwania rozgrywki przechowuje oraz aktualizuje strukturę danych, w której zapisane są wszystkie szczegóły dotyczące mapy rozgrywki oraz położenie wszystkich graczy. Następnie serwer przygotowuje odpowiedź dla każdego z graczy.

Pierwsza odpowiedź serwera składa się z:

- unikatowego identyfikatora gracza,
- zestawu wszystkich pseudonimów pozostałych graczy (ale nie ich identyfikatorów),
- struktury danych zawierającej wycinek obszaru gry, bezpośrednio otaczający gracza wraz z wszystkimi efektami dalszymi od niego, które byłyby w stanie dostrzec przy pomocy innych zmysłów,
- poziom śladu zapachowego w aktualnym miejscu,
- identyfikator kolejnej komendy od gracza,
- czas pozostały do przetrwania roju (służy do zaktualizowania wartości licznika działającego w aplikacji klienckiej).

Dalsza komunikacja inicjowana jest przez klienta, który może przesłać jedną z dostępnych komend (to która komenda zostanie wysłana, zależy od klawisza wciśniętego przez gracza):

- ruch wraz z podaniem wybranego kierunku (góra, dół, lewo, prawo),
- wydanie odgłosu z podaniem wybranego z dostępnych,
- zebranie żywności,
- odstawienie żywności.

Wraz z komendą przesyłany jest identyfikator zachowany po poprzedniej wiadomości z serwera.

W sytuacji, gdy gracz wysyła więcej niż jedną komendę, tylko pierwsza z danym identyfikatorem jest brana pod uwagę w trakcie. Serwer oczekuje na komendy, a po ustalonym odstępie czasu, aktualizuje strukturę danych przechowującą szczegóły gry, zgodnie z kolejnością wybranych komend, po czym rozsyła każdemu z graczy odpowiedź analogiczną do pierwszej, biorąc pod uwagę zaktualizowaną strukturę danych gry. Zmieniany jest też identyfikator kolejnej komendy dla każdego z graczy.

Po każdej otrzymanej odpowiedzi serwera klient aktualizuje stan gry z punktu widzenia samego gracza i zaczyna nasłuchiwać i natychmiast przysyłać wszystkie kolejne komendy.

Gra kończy się w momencie, gdy czas pozostały do przetrwania roju spadnie poniżej sekundy. Czas ten wydłużany jest przez każde przyniesione przez graczy i pozostawione w komorze głównej pożywienie. Jednocześnie serwer przelicza całkowity czas rozgrywki, aby zwrócić go w ostatecznym podsumowaniu rozgrywki dla każdego z graczy.