

Współbieżny program w języku Ada  
inspirowany grą

## “Roll for the Galaxy”

autorzy: Zuzanna Misztal, Wojciech Konieczkiewicz

### Informacje ogólne

“Roll for the Galaxy” jest grą kościaną, której autorami są Wei-Hwa Huang oraz Tom Lehmann. Przeznaczona jest dla dwóch do pięciu graczy, z których każdy buduje swoje własne kosmiczne imperium. Powodem, dla którego wybraliśmy właśnie tę grę, jest jej mechanika, a konkretnie fakt, że gracze wykonują swoje akcje w tym samym czasie. Nasz program nie realizuje wszystkich założeń tej gry - sporo zasad zostało pominiętych w celach ułatwienia implementacji.

### Przebieg gry

#### Rozpoczęcie

Każdy gracz losuje dwie planety z puli planet początkowych, umieszcza je w swoim imperium jako planety już zdobyte oraz otrzymuje zapewnione przez te planety kostki lub kredyty. Każda planeta przechowuje informacje o kolorze dostarczanej kostki, miejscu, w którym należy tę kostkę umieścić (w kubku, populacji, bądź jako dobro na tej planecie) oraz ilości kredytów, które są przyznawane w momencie osiedlenia tej planety. Gracze dostają też jeden kredyt, dwie białe kostki do populacji oraz cztery białe kostki do kubka. Następnie losują z głównego worka z planetami po jednej planecie i układają ją w kolejce planet czekających na skolonizowanie.

#### Rundy

W trakcie rundy każdy gracz w sekrecie rzuca kostkami, które znajdują się w jego kubku, przydziela jokery do dowolnie wybranych przez siebie akcji oraz wybiera jedną akcję, którą aktywuje. Gdy wszyscy gracze wybiorą akcję odsłania się zasłonki, tak aby wszyscy widzieli wyniki rzutów kośćmi, a przede wszystkim akcje aktywowane przez przeciwników. Kości, na których wypadły akcje nie aktywowane przez żadnego gracza nie będą używane w tej rundzie i wracają do kubków graczy do wykorzystania w kolejnej rundzie. Po tej synchronizacji gracze wykonują aktywowane akcje równocześnie wykorzystując do tego kostki o odpowiednich wynikach. Zużyta kostka wraca do populacji. Po zakończeniu wykonywania akcji następuje faza “zarządzania imperium”, czyli wydanie po jednym kredycie na przeniesienie dowolnej kostki z populacji do kubka, aż skończą się kredyty bądź kostki w populacji. Jeśli na koniec fazy “zarządzania imperium” gracz ma 0 kredytów, automatycznie dostaje 1 kredyt.

## Akcje

W grze dostępne są 4 akcje:

### 1. Odkrywaj (explore)

Każdą kostkę, na której wypadła akcja odkrywania można wykorzystać na jeden z dwóch sposobów: eksploracja albo magazynowanie. Magazynowanie polega na otrzymaniu dwóch kredytów. Eksploracja to wylosowanie planety z głównego worka z planetami i ułożenie jej na końcu kolejki planet czekających na skolonizowanie. Niezależnie od sposobu wykorzystania kostki wraca ona do populacji gracza.

### 2. Kolonizuj (settle)

Kości, na których wypadła akcja kolonizacji należy kolejno przenosić do strefy kolonizatorów. W momencie, gdy liczba kolonizatorów jest równa kosztowi (inaczej - wartości) pierwszej planety w kolejce do skolonizowania, planeta ta staje się planetą skolonizowaną. Przenosi się ją z kolejki do planet zdobytych oraz otrzymuje się zapewnione przez tę planetę kostki lub kredyty, a kostki ze strefy kolonizatorów wracają do populacji. Całość należy powtarzać, aż skończą się kostki kolonizatorów lub planety w kolejce. W tym drugim przypadku nie należy przenosić niewykorzystanych kości do strefy kolonizatorów. Zamiast tego wracają do kubka gracza.

### 3. Produkuj (produce)

Każda kość producenta może wyprodukować dobro na jednej z planet należących do imperium gracza. Na szarych planetach nie można wyprodukować dobra. Na jednej planecie może znajdować się co najwyżej jedno dobro. Opłacalne jest dobieranie koloru kostki, która ma stać się dobrem, do koloru planety, na której to dobro ma zostać wyprodukowane. Nadmiarowe kostki producentów wracają do kubka.

### 4. Dostarczaj (ship)

Każda kość z akcją dostarczania bierze dobro znajdujące się na jednej z planet w imperium gracza i "konsumuje". Po wykonaniu akcji, obie kości (dostarczająca i dobro) trafiają do populacji gracza. Każda konsumpcja daje graczowi od 1 do 3 punktów zwycięstwa. Jeden punkt przyznawany jest zawsze. Jeśli kolor dobra zgadza się z kolorem planety, na której się znajdował, przydzielany jest drugi punkt. Analogicznie, jeżeli kolor kości dostarczającej zgadza się z kolorem planety, z której jest dostarczane, przydzielany jest dodatkowy punkt. Fioletowa kostka zastępuje dowolny kolor i zawsze daje 1 punkt.

## Koniec gry

Gra kończy się, gdy jeden z graczy ma już co najmniej 7 skolonizowanych planet (wliczając w to dwie początkowe). W takiej sytuacji należy dokończyć rundę, a następnie policzyć końcowe punkty, czyli do zebranych punktów zwycięstwa dodać sumę wartości (kosztów) wszystkich skolonizowanych planet w imperium gracza.

## Decyzje graczy

Pierwszą decyzją jaką muszą podjąć gracze jest wybór aktywowanej akcji. W oryginalnej grze inteligentny gracz bierze pod uwagę zapotrzebowanie na nowe planety w kolejce, kredyty, ilość dóbr na planetach, podejrzenia co mogą aktywować pozostali gracze oraz wyniki wyrzuconych kości. W naszej implementacji gracze nie są tak inteligentni i aktywują akcję, która wypadła na ich kościach najczęściej razy (przy czym w przypadku remisu kierują się kolejnością wyboru: kolonizacja, produkcja, dostawa, odkrywanie). Przydzielenie jokerów z kostek do konkretnych akcji również leży w decyzji gracza i mogłoby być wykonane na wiele różnych sposobów - nasi wirtualni gracze przydzielają wszystkie jokery do akcji, którą aktywowali. Parę decyzji graczy zostało u nas w ogóle wyeliminowanych np. możliwość przydzielenia jednej kostki do innej akcji niż na niej wypadła, "płacąc" za to przeniesieniem innej dowolnej kostki z powrotem do kubka (tak zwany "dyktat"), czy możliwość wymieniania planet z kolejki na nowe z worka.

Podczas akcji produkcji oraz dostawy przydatne jest dopasowanie koloru kostki do koloru planety i jest to uwzględnione w programie. W pierwszej kolejności każda nie szara planeta stara się wyprodukować dobro w swoim kolorze, a dopiero gdy już takie dopasowania się skończą użyte są pozostałe kostki (analogicznie dla dostawy).

Ostatnią decyzją w ciągu rundy jest zakup kostek z populacji do kubka. Jeśli gracz stać na wszystkie kostki kupuje wszystkie (zgodnie z zasadami gry musi tak zrobić), w przeciwnym przypadku wybiera kostki losowo w liczbie równej posiadanym kredytom.

## Komunikacja między zadaniami

Implementacja gry opiera się na zadaniu `Game` (zadanie kontrolujące przebieg całej rozgrywki oraz częściowo przekazujące informacje między graczami) oraz czterech zadaniach typu `Player` (imitujących prawdziwych graczy, każdy `Player` ma swój wyróżnik - numer gracza).

Głównym wyzwaniem w implementacji gry było wymuszenie oczekiwania graczy po rzucie kośćmi (realizowane przez `Game.Sync` oraz tablicę semaforów `Sems` - każdemu graczowi odpowiada jeden semafor) i przekazaniu swojej aktywowanej akcji tak, żeby zadanie `Game` mogło wybrać aktywowane akcje dla wszystkich graczy (realizowane przez `Game.Get`). Następnie, zgodnie z oryginalnymi zasadami gry, każdy `Player` może niezależnie wykonać swoją turę aż do kolejnego rzutu kośćmi i wyboru swojej akcji. Schemat komunikacji między zadaniami `Player` a zadaniem `Game` został przedstawiony w załączonym pliku *schemat.pdf*. Dzięki wykorzystaniu dodatkowego semafora `F` możliwe jest zakończenie wszystkich zadań w odpowiednim momencie (jeżeli w trakcie wykonywania tury którykolwiek z graczy osiągnie warunek końca gry, to informuje o tym `Game` przy następnym `Game.Sync` poprzez argument - wtedy `Game` zajmuje semafor `F`, co pozwala wszystkim zadaniom zakończyć swoje działanie).

Dodatkowo wykorzystano semafor `Sack_Semaphore` do ograniczenia dostępu do

wspólnego dla wszystkich graczy worka z kartami planet oraz semafor `Init_Semaphore`, który ogranicza dostęp do wspólnych zasobów niezbędnych do rozpoczęcia gry.

## Struktura projektu

W folderze `src` znajdują się pliki źródłowe:

- `roll.adb` - zawierający zadania, semafony oraz pętlę główną programu;
- `player_operations.ads` - zawierający nagłówki paczki implementującej mechanikę rozgrywania tury przez gracza;
- `player_operations.adb` - zawierający ciała paczki implementującej mechanikę rozgrywania tury przez gracza.

W folderze głównym znajduje się plik `roll.gpr`, służący do kompilacji całego projektu (`gnatmake -P roll.gpr`).

Aby rozpocząć rozgrywkę, należy uruchomić plik wykonywalny `roll` znajdujący się w folderze `obj`.

