

Sprawozdanie z projektu PWiR

Temat: Hodowla ryb (karpi)

Jan Godlewski, 303107

Joanna Partyka, 296929

Informatyka, 3 rok, 5 semestr

1. Tytuł programu: Funkcjonowanie instytucji - hodowla ryb.
2. Dane studentów:
 - a. Jan Godlewski 303107 3 rok, Informatyka EAlIB, grupa 2a
 - b. Joanna Partyka 296929 3 rok, Informatyka EAlIB, grupa 2b
3. Data oddania programu: 27.01.2021
4. Cel programu:

Celem programu jest stworzenie symulacji hodowli ryb w języku Erlang. Wybraliśmy, że będą hodowane karpie ze względu na popularność tego gatunku. Program znajduje się w pliku ryby.erl.
5. Opis i schemat struktury zadaniowej programu:

Podstawowe zmienne:

K - lista przechowująca hodowane karpie; każdy element to lista karpie w danym wieku (np. $K[1]$ = liczba karpie w wieku 1 roku); indeksów na liście jest 20, ponieważ na wolności karpie dożywają 20 lat; karpie w wieku poniżej 3 lat nie są dojrzałe i nie mogą się rozmnażać, z tego powodu ilośc karpie dzielony jest na dwie podgrupy - karpie małe i dorosłe; przy każdym kroku następuje rozmnażanie ryb; przy uruchomieniu symulacji lista jest zapełniana losowymi wartościami

Y - bieżący rok (krok symulacji)

P - racje żywieniowe (0-3, domyślnie 2); racje żywieniowe mają wpływ na żywotność ryb

UWAGA! wybór racji 0 spowoduje śmierć głodową wszystkich ryb w następnej iteracji

Zmienne dodatkowe:

W - wiek dodawanych ryb

N - numer indeksu przy iterowaniu listy karpie

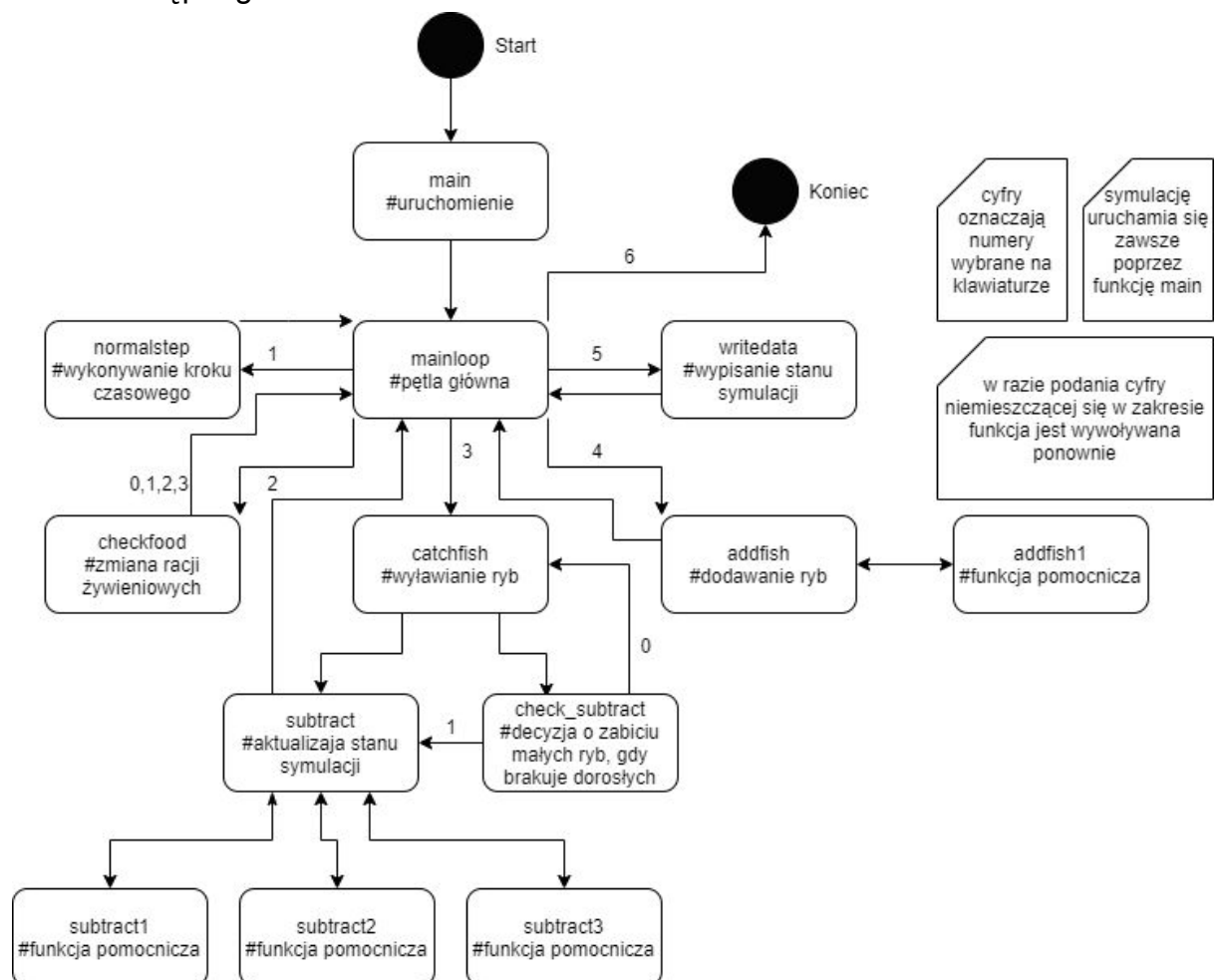
A - pozostała liczba karpie do usunięcia z bazy

A,B,C itp. - zmienne tymczasowe

 - a. main - główna funkcja wypisująca wstępne informacje; od niej należy uruchamiać program
 - b. mainloop - pętla główna programu; przy pomocy klawiatury wybiera się w niej następne działanie; w razie wpisania niewłaściwej cyfry następuje ponowne wywołanie funkcji
 - c. normalstep - wykonywanie kolejnego kroku symulacji; automatycznie wypisuje stan symulacji
 - d. checkfood - zmiana racji żywieniowych

UWAGA! wybór racji 0 spowoduje śmierć głodową wszystkich ryb w następnej iteracji

- e. catchfish - umożliwia wyławianie (i zabijanie) ryb; gdy brakuje dorosłych ryb i konieczne byłoby zabicie małych ryb, wyświetlane jest pytanie o potwierdzenie (funkcja pomocnicza check_subtract); w razie wpisania niewłaściwej cyfry następuje kolejna iteracja pętli
 - i. subtract - usuwa wyłowione ryby z symulacji (funkcje pomocnicze subtract1, subtract2 i subtract3)
- f. addfish - możliwość dodania dodatkowych ryb do hodowli (funkcja pomocnicza addfish1); wymagane jest podanie wieku dodawanych ryb
- g. writedata - wypisanie aktualnego stanu symulacji bez wykonania następnego kroku



6. Informacje o stosowanych pakietach zewnętrznych:
Zaimportowano pakiet lists umożliwiający wykonywanie bardziej zaawansowanych operacji na listach.
7. Specyficzne metody rozwiązywania problemów:
 - a. iterator automatycznie tworzący i uzupełniający listę
[rand:uniform(10) || _ <- lists:seq(1,20)]
8. Krótka instrukcja obsługi:
 - a. Uruchomić główną funkcję programu c(ryby).
ryby:main().

b. Możliwość wyboru funkcji.

- i. Następny krok symulacji
- ii. Zmiana ilości pożywienia (wartości 0-3)
UWAGA! wybór racji 0 spowoduje śmierć głodową wszystkich ryb w następnej iteracji
- iii. Odłów części ryb (lub wszystkich)
- iv. Dokup ryby
- v. Wypisz aktualne dane (bieżący stan symulacji)
- vi. Wyłączenie programu

9. Testy, przykłady:

Uruchomienie programu

```
Symulator hodowli karpia
Jan Godlewski Joanna Partyka
Program symuluje hodowle ryb. W zaleznosci od ilosci pozywienia
i polowow suma ryb zmienia sie w czasie. Generowanie hodowli

Rok numer: 0
Ilosc pozywienia: 2
Wygenerowane karpie:
[2,10,1,8,7,4,6,2,4,1,5,9,1,4,9,1,7,2,10,10]
Suma wszystkich karpia: 103
Suma karpia dzieci: 12
Suma karpia doroslych: 91
```

Główny interfejs

```
Wybierz dzialanie:
1-wczytaj nastepny krok:
2-zmien ilosc pozywienia:
3-odlow czesc ryb:
4-dokup ryby:
5-wypisz aktualne dane:
6-zamknij:
Dzialanie: █
```

Zmiana ilości pożywienia

```
Dzialanie: 2
Zmiana ilosci pozywienia
0-brak pozywienia (wszystkie ryby umra)
1-malo pozywienia (mala rozrodczosc)
2-srednio pozywienia (srednia rozrodczosc)
3-duzo pozywienia (duza rozrodczosc)
Podaj nowa ilosc pozywienia
Ilosc: █
```

Wylawianie ryb

```
Dzialanie: 3
Wprowadz liczbe ryb do wylowienia
Liczba ryb: 120
Konieczne bylyby zabicie niedoroslych ryb
Czy chcesz kontynuowac?
0-nie
1-tak
Dzialanie: 1
```

10. Możliwe rozszerzenia programu:

W przyszłości do programu można zaimplementować system obsługi pieniędzy. Sprzedaż karpia (nie zagłodzenie) dawałaby pieniądze, dodawanie nowych karpia zabierałoby pieniądze.

11. Ograniczenia programu:

Nie można dokonywać zmian w symulacji między krokami.

12. Inne informacje:

Zaletą języka Erlang jest to, że bardzo łatwo można zaimplementować obsługę niewłaściwych danych w konsoli. W innych językach kończyłoby się to zwykle awarią całego programu bez dodania rozbudowanej obsługi wyjątków.