

# KURS PROGRAMOWANIA W JAVIE

## ROBAKI

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

### Zadanie.

Dana jest kwadratowa *plansza* podzielona na  $N$  wierszy i  $M$  kolumn, gdzie  $N$  i  $M$  są ustalonymi liczbami naturalnymi. Na planszy znajduje się więc  $N \cdot M$  prostokątnych pól. Plansza jest zawinięta, czyli przeciwległe brzegi planszy stykają się. Pole sąsiednie to takie, z którym dane pole ma wspólny bok — ponieważ plansza jest zawinięta, to każde pole ma czterech sąsiadów, również pola znajdujące się na brzegach. Pomiędzy polami można się poruszać przechodząc w jednym kroku do pola sąsiedniego.

Na planszy tej znajdują się *robaki*. Jedno pole na planszy może być w danym momencie zajęte co najwyżej przez jednego żerującego robaka. Kiedy robak skończy żerować na swoim polu to przemieszcza się na pole sąsiednie; gdy pole to jest zajęte przez innego robaka, wówczas następuje atak. Po zakończeniu żerowania robak wybiera sąsiednie pole, do którego będzie się przemieszczał losowo, ale z zachowaniem określonych proporcji: utrzymuje dotychczasowy kierunek marszu z prawdopodobieństwem  $\frac{3}{4}$ , skręca w lewo bądź w prawo z prawdopodobieństwem  $\frac{1}{8}$  i nigdy się nie cofa.

Robaki oprócz *głowy* mają także *ogony* i ciągną je za sobą zajmując w rzeczywistości nie jedno a kilka pól. Gdy robak przemieszczając się trafi na ogon innego robaka, to odgryza rywalowi jego ogon w tym polu; rywal żyje dalej z krótszym ogonem, ale za to robakowi który zaatakował wydłuża się ogon o jedno pole. Gdy robak trafi na swój własny ogon, to również go odgryza (po samookaleczeniu robakowi nie wydłuża się ogon). W sytuacji gdy robak przemieszczając się trafi na głowę innego robaka, wtedy zaatakowany rywal ginie.

Napisz aplikację okienkową w technologii *Swing*, która zilustruje zachowanie się robaków na planszy. Każdy robak powinien być osobnym wątkiem. Plansza niech będzie obiektem, na którym te wątki się synchronizują. Wątek robaka wykonuje cyklicznie dwie czynności: przemieszcza się z pola na pole (w losowo wybranym kierunku zgodnie z opisanymi wcześniej prawdopodobieństwami) i żeruje (przebywając na polu losową liczbę milisekund — od 100 do 1000). Osobny wątek będący obiektem klasy `javax.swing.Timer` ma odświeżać rysunek planszy na obiekcie typu `javax.swing.JPanel` przynajmniej co  $\frac{1}{20}$  sekundy.

W czasie działania aplikacji użytkownik ma mieć możliwość dodawania nowych robaków za pomocą podwójnego kliknięcia myszy na obszarze rysunkowym `JPanel`.

Rozmiar planszy  $N \times M$  określ za pomocą argumentów wywołania programu, a jeśli użytkownik nie poda żadnych argumentów programowi, to niech domyślny rozmiar planszy wynosi  $10 \times 10$ . Początkowo umieść na planszy  $N + M$  robaków na losowo wybranych pozycjach. Zakładamy, że każdy nowy robak jest krótki (zajmuje tylko jedno pole) oraz ma określony kierunek ruchu (wybrany losowo).

### Uwaga.

Program należy napisać, skompilować i uruchomić w środowisku zintegrowanym *NetBeans*.