

Zaawansowane Metody Programowania Obiektowego – zadanie 4

Przetwarzanie drzew, użycie obiektów

UWAGA:

- 1. Pisząc własny program można użyć innego nazewnictwa niż to przedstawione w treści zadania. Należy jednak użyć jakiejś spójnej konwencji kodowania, zgodnie z wymaganiami kursu.**
- 2. Program NALEŻY NAPISAĆ OBIEKTOWO.**

W ramach zadania należy napisać własną metodę Programowania Genetycznego (PG). PG służy do rozpoznawania z jakiej funkcji wygenerowano określone dane, lub jaką funkcją należy je przybliżyć.

Wyobraźmy sobie następującą funkcję dwóch zmiennych:

$$f(x, y) = (x - 2) * (y + 4)$$

Na jej podstawie można utworzyć plik z danymi, który będzie zawierał następujące trójki:

<wartość x >; <wartość y >; <wartość $f(x, y)$ >

Na przykład:

2;1;0

2.5;2;3

10;23.8;222.4

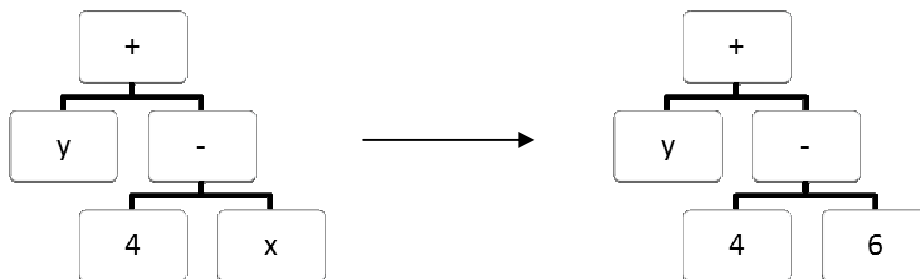
W ramach zadania nr 3 wykonany został obiekt CTree implementujący drzewo, które przechowuje określone wyrażenie algebraiczne (wczytywane w formie wyrażenia w notacji prefiksowej). Obiekt z CTree z zadania nr 3 należy rozszerzyć o dwie następujące funkcjonalności: mutację i krzyżowanie.

Mutacja

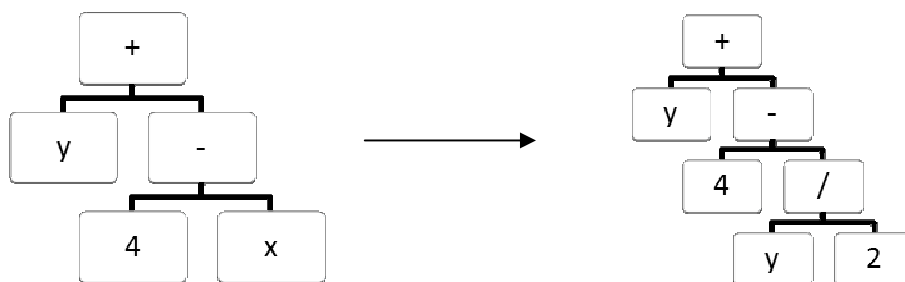
Mutacja polega na drobnej zmianie struktury drzewa i może być realizowana na dwa różne sposoby: zamianie losowo wybranego węzła lub liścia na nowo wygenerowany (w losowy sposób) węzeł lub liść, zamianie miejscami dwóch losowo wybranych węzłów, liści lub węzła i liścia. Poniżej znajdują się przykłady dla obu powyższych możliwości.

- Zamiana losowo wybranego węzła lub liścia na nowo wygenerowany, w sposób losowy, węzeł lub liść

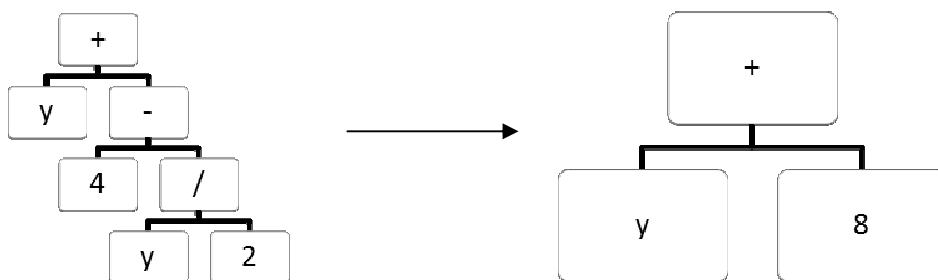
- przykład zamiany wybranego liścia na nowy liść



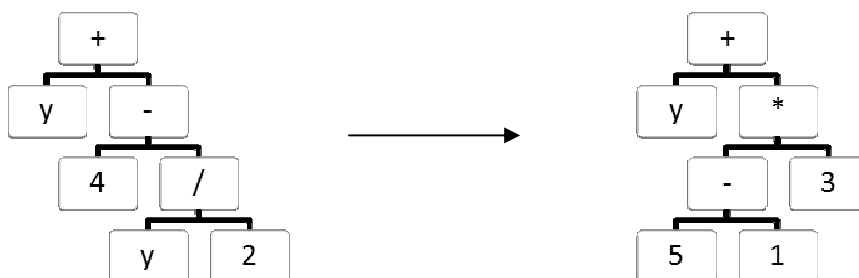
- przykład zamiany wybranego liścia na nowy węzeł



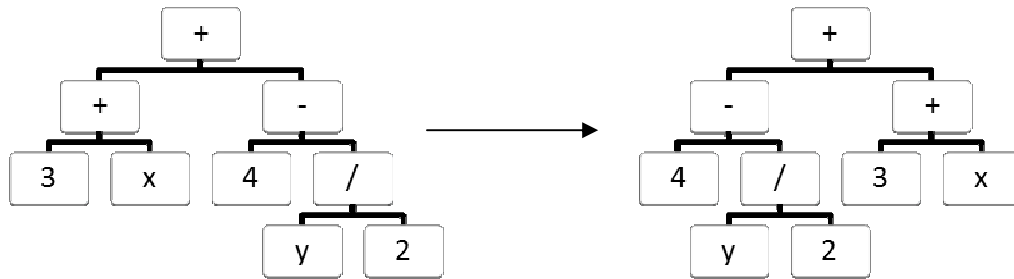
- przykład zamiany wybranego węzła na nowy liść



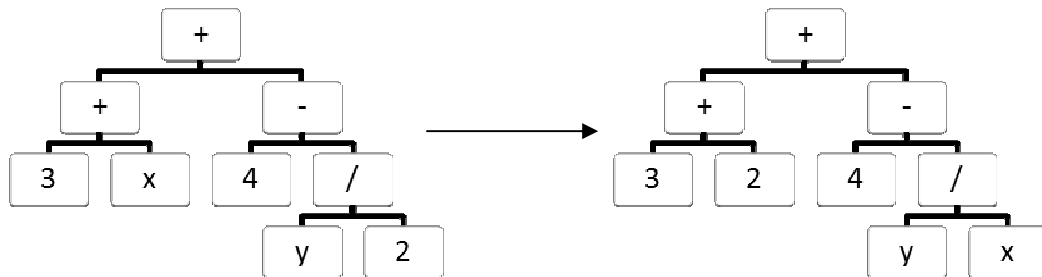
- przykład zamiany wybranego węzła na nowy węzeł



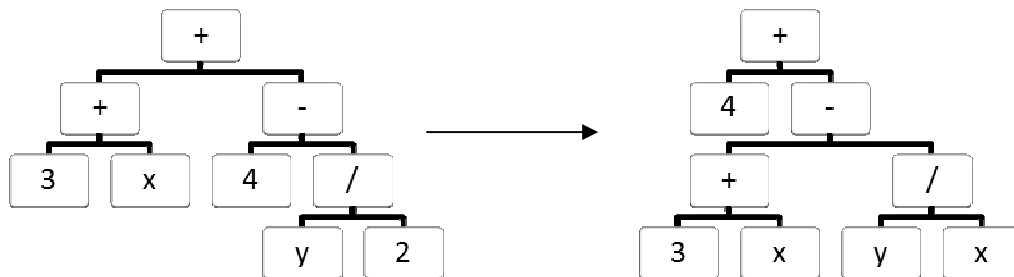
- zamiana miejscami dwóch losowo wybranych węzłów, liści lub węzła i liścia
 - przykład zamiany miejscami dwóch węzłów



- przykład zamiany miejscami dwóch liści



- przykład zamiany miejscami węzła i liścia

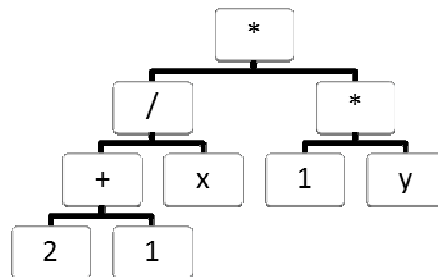
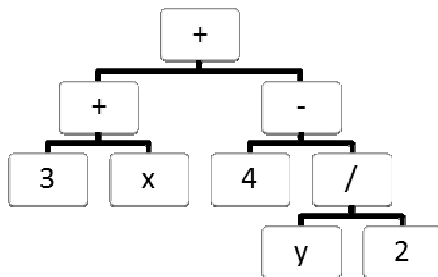


Krzyżowanie

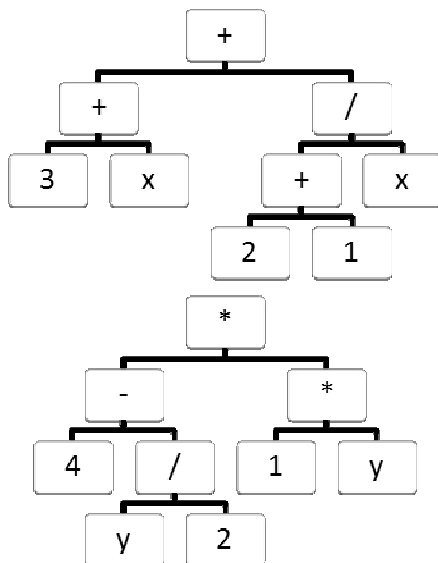
Podczas operacji krzyżowania, drzewo krzyżowane jest z innym drzewem. W wyniku krzyżowania powstają dwa nowe drzewa. Krzyżowane drzewa nazywane są drzewami rodzicielskimi, natomiast drzewa powstałe w wyniku krzyżowania to drzewa potomne. Krzyżowanie polega na losowym podziale każdego drzewa rodzicielskiego na dwa poddrzewa, a następnie sklejaniu pierwszego poddrzewa pierwszego drzewa rodzicielskiego z drugim poddrzewem drugiego drzewa rodzicielskiego i na odwrót. W wyniku sklejania powstają dwa drzewa potomne.

Przykład krzyżowania:

- drzewa rodzicielskie



- drzewa potomne



Programowanie genetyczne

Klasa CTree zaimplementowana w ramach zadania nr 3 i rozszerzona o operacje krzyżowania i mutacji, pozwala na wyliczenie wartości reprezentowanego wyrażenia algebraicznego dla podanych x i y . Wykorzystując plik utworzony podczas wykonywania etapu 1, można wyliczyć jakość reprezentowanego przez drzewo wyrażenia algebraicznego tzn. jak bardzo wyrażenie algebraiczne reprezentowane przez drzewo jest podobne do zdefiniowanej w etapie 1 funkcji f . Do tego celu należy użyć następującej miary:

$$\sum_{i=1}^n (z_i - \hat{z}_i)^2$$

gdzie n oznacza liczbę wyliczeń w pliku, z_i to wartość zdefiniowanej funkcji f dla x i y z i -tego wyliczenia, \hat{z}_i jest wartością wyrażenia algebraicznego reprezentowanego przez drzewo dla x i y z i -tego wyliczenia. Im mniejsza wartość powyższej miary tym wyrażenie algebraiczne reprezentowane przez drzewo jest bardziej podobne do funkcji f .

Program wykorzystujący programowanie genetyczne składa się z czterech parametrów:

- N_p – rozmiar populacji,
- N_i – liczba iteracji,
- P_k – prawdopodobieństwo krzyżowania,
- P_m – prawdopodobieństwo mutacji.

W ramach tego zadania populację stanowią będą drzewa reprezentujące wyrażenia algebraiczne. Schemat działania programowania genetycznego można zapisać następującym pseudokodem:

```
inicjacja
ewaluacja
i <- 0
dopóki i < Ni wykonaj
    selekcja
    krzyżowanie
    mutacja
    ewaluacja
    i <- i + 1
koniec dopóki
```

Wynikiem każdej iteracji programowania genetycznego jest nowa populacja drzew.

Inicjacja

Podczas inicjalizacji należy losowo wygenerować N_p drzew. Wygenerowane losowo drzewa stanowią populację początkową.

Ewaluacja

Każdemu drzewu w populacji należy przypisać wartość miary zdefiniowanej w etapie 3. Wartość ta nazywana jest wartością funkcji przystosowania. Im mniejsza jej wartość tym drzewo jest lepiej przystosowane.

Selekcja

Programowanie genetyczne opiera się na idei ewolucji, z którą ściśle związana jest selekcja naturalna. Podczas selekcji wybierane są do krzyżowania drzewa o lepszym przystosowaniu. W tym celu należy użyć selekcji turniejowej, która polega na wykonaniu N_t turniejów. W każdym turnieju biorą udział dwa losowo wybrane ze zwracaniem drzewa z populacji. Zwycięzca każdego turnieju (drzewo o lepszym przystosowaniu) dodawany jest do populacji rodziców. Po przeprowadzeniu selekcji, populacja rodziców powinna zawierać N_p drzew, które następnie zostaną poddane krzyżowaniu.

Krzyżowanie

Wynikiem tej fazy jest populacja potomków. Drzewa znajdujące się w populacji rodziców należy połączyć w pary. Każda para ma P_k szans, że zostanie skrzyżowana. Jeżeli para została skrzyżowana to do populacji potomków dodajemy dwa drzewa potomne powstałe zgodnie ze schematem opisanym w ramach etapu 2. Jeżeli krzyżowanie się nie wykonało to do populacji potomków należy dodać dwa drzewa rodzicielskie.

Mutacja

Każde drzewo w populacji potomków ma P_m szans, że zostanie zmutowane. Mutacja pojedynczego drzewa odbywa się zgodnie ze schematem opisanym w etapie 2. Po zakończeniu tej fazy populacja potomków staje się nową populacją.

Wynikiem działania programowania genetycznego jest drzewo, które uzyskało najmniejszą wartość miary zdefiniowanej w etapie 3 podczas działania całego programu.

W ramach zadania należy spróbować dobrać takie wartości parametrów programowania genetycznego (liczba drzew, prawdopodobieństwo krzyżowania) by wynikowe drzewo reprezentowało wyrażenie algebraiczne, które jest praktycznie równoważne funkcji f . Innymi słowy, wartość miary zdefiniowanej w etapie 3 dla wynikowego drzewa powinna być zbliżona do 0.

Dla wykonania zadania należy przyjąć, że poszukiwana funkcja bierze 3 parametry wejściowe.

Program nie musi posiadać interfejsu użytkownika. Po uruchomieniu ma wczytywać dane z pliku z danymi, uruchamiać metodę PG, zaprezentować wynikowe drzewo na ekranie i zapisać je do pliku.