Metoda bisekcji

#metody_numeryczne
#bisekcja

Czas na odrobinę wiadomości o metodach numerycznych, czyli takim sposobie rozwiązywania problemów matematycznych czy fizycznych, gdzie operuje się na konkretnych liczbach/wartościach, a nie na abstrakcyjnych symbolach. Każdy informatyk powinien znać przynajmniej podstawy tej dziedziny, a dla przyszłych przyrodników, inżynierów czy twórców gier to prawdziwy kanon.[*]

Zaczniemy skromnie i litościwie, od metody rozwiązywania równań algebraicznych z jedną niewiadomą. Jeśli srogi belfer zarzuci Wam równanie w rodzaju:

$$2x + 7 = 3,$$

to chyba większość z Was sobie poradzi, prawda? Nawet jeśli dostaniecie równanie typu:

$$3x^2 - 5x - 1 = 0,$$

to też jakoś pójdzie (jeśli będziecie mieli dobry dzień!). Nawet takie oto równanie:

$$4x^3 + 2x^2 + 3x + 1 = 0,$$

także da się rozwiązać, jeśli ktoś słyszał o twierdzeniu Bézout i wzorach Cardana (do wygooglania). Równanie czwartego stopnia też jest możliwe do rozwiązania w każdym przypadku, są na to wzory, choć już nieco bardziej skomplikowane.

Niestety z równaniami wyższych stopni jest problem. Na ogół nie da się ich rozwiązać analitycznie (czyli przy pomocy kartki papieru i ołówka). Tylko szczególne przypadki uda się nam rozkminić, ale w ogólności nie mamy szans.[†]

Niestety im dalej, tym gorzej. Jeśli w równaniu występują rozmaite funkcje elementarne (poza funkcją potęgową o wykładniku naturalnym), to z reguły możemy iść do domu. Takie równania nazywa się *równaniami przestępnymi* (ang. **transcendent equations**, wymawiaj: *transendent ekłejszyns*). Ich po prostu na ogół *nie da się* rozwiązać analitycznie, i tyle.

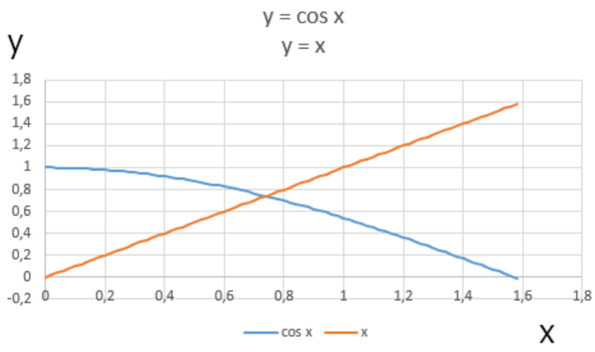
Jeśli ktoś nie wierzy, weźmy prosty przykład, równanie:

$$\cos x = x.$$

[*] Warto wiedzieć, że podstawy metod numerycznych pojawiły się ze 2000 lat temu, choć dopiero w czasach Newtona, Lagrange'a, Gaussa i Eulera nastąpił prawdziwy ich rozkwit. No, a pojawienie się komputerów, to dopiero była petarda!

[†] Każde równanie algebraiczne stopnia n ma dokładnie n pierwiastków, niekoniecznie różnych i niekoniecznie rzeczywistych – ale jednak ma. Jest to tak zwane *podstawowe twierdzenie algebry*. Tylko jak znaleźć te pierwiastki?

Czy ono ma rozwiązanie? Ano ma i to dokładnie jedno: wystarczy narysować sobie cosinusoidę oraz wykres funkcji liniowej $y = x$ i jak wół będzie widać, że się przetną. Czyli rozwiązanie istnieje, nieprawdaż? Gdzieś tam w przedziale $[0, \pi/2]$:



Nie jesteśmy jednak na przegranej pozycji, gdyż posiadamy następującą wiedzę:

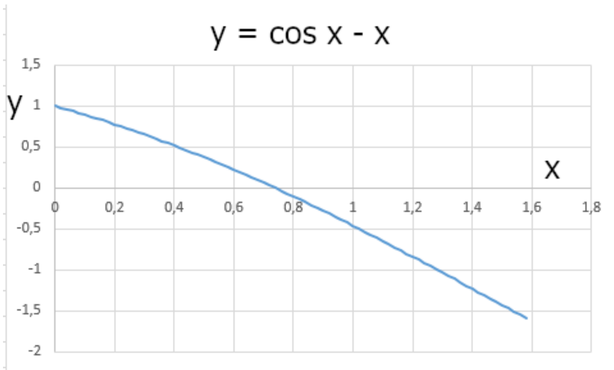
- Pierwiastek równania (x) istnieje i jest dokładnie jeden,
- Wartość x mieści się pomiędzy 0 a $\pi/2$.

Możemy pokusić się o znalezienie tegoż pierwiastka w sposób przybliżony – ale z dowolnie dużą dokładnością.[‡]

Zaczniemy od przeniesienia x z prawej strony na lewą. Tym samym nasze równanie przybiera postać:

$$f(x) = \cos x - x = 0,$$

czyli problem został sprowadzony do poszukiwania miejsca zerowego funkcji $f(x) = \cos x - x$ we wspomnianym przedziale:



Co możemy powiedzieć o funkcji $f(x)$? Jest na bank funkcją ciągłą[§] w przedziale domkniętym $[0, \pi/2]$. Ponadto możemy sprawdzić wartości funkcji $f(x)$ na krańcach tego przedziału. No i mamy:

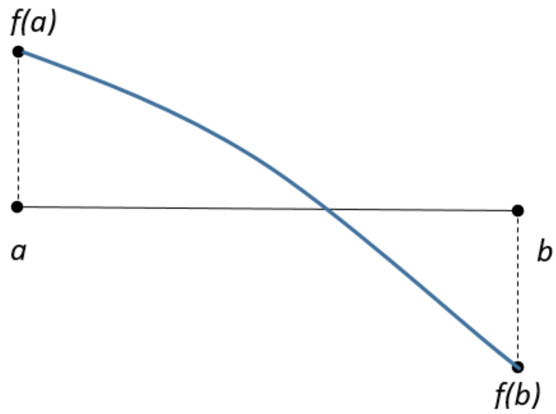
$$f(0) = \cos 0 - 0 = 1 - 0 = 1 > 0,$$
$$f\left(\frac{\pi}{2}\right) = \cos\frac{\pi}{2} - \frac{\pi}{2} = 0 - \frac{\pi}{2} = \frac{\pi}{2} < 0.$$

[‡]No, może nie taką dowolną, gdyż używany przez nas język programowania ma ograniczoną dokładność obliczeń zmiennoprzecinkowych. Jednak zawsze możemy liczyć na kilkanaście dokładnych cyfr wyniku.

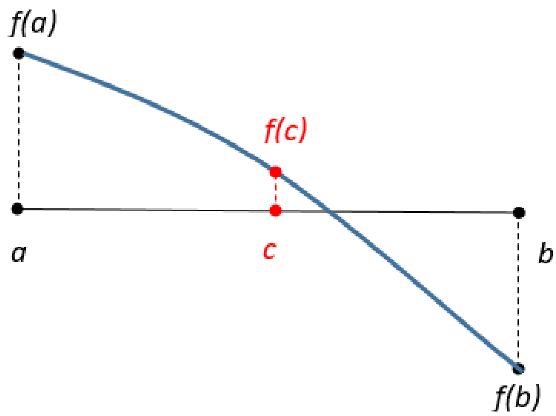
[§]Pojęcie ciągłości funkcji nie jest – wbrew pozorom – trywialnym zagadnieniem. Wszelako dla zwykłych, Bożych funkcji (dla których dziedziną jest zbiór liczb rzeczywistych – jak w tym przypadku), oznacza ono, że wykres na spójnym/jednokawałkowym przedziale zawierającym się w dziedzinie funkcji jest również jednokawałkowy (nie ma przerw).

Wartości te są przeciwnych znaków. I tu przychodzi nam z pomocą twierdzenie Bolzano-Cauchy'ego (równoważne tak zwanej własności Darboux), które w skrócie mówi, że w takiej sytuacji nasza funkcja $f(x)$ musi mieć przynajmniej jedno miejsce zerowe w tym przedziale.[¶]

OK, my już wiemy, że takie miejsce zerowe jest jedno. Możemy narysować sobie plan sytuacyjny, nawet dla nieco ogólniejszego przypadku. Na osi poziomej (osi odciętych, X) mamy przedział $[a, b]$ taki, że funkcja $f(x)$ na końcach tego przedziału przyjmuje wartości o przeciwnych znakach:



Oczywiście ważne jest tylko to, że wartości funkcji na krańcach przedziału są przeciwnych znaków, a nie to, która z nich jest rzeczywiście dodatnia czy ujemna. Teraz sprawdzamy wartość funkcji w środku przedziału, czyli w punkcie $c = \frac{a+b}{2}$:



Możemy teraz ograniczyć nasz przedział $[a, b]$ do podprzedziału $[a, c]$ lub $[c, b]$: jeśli znak $f(c)$ jest przeciwny, niż znak $f(a)$, wybieramy podprzedział $[a, c]$ (czyli b otrzymuje wartość c) – w przeciwnym razie wybieramy podprzedział $[c, b]$ (a otrzymuje wartość c). W ten sposób przedział, w którym na pewno znajduje się pierwiastek równania (miejsce zerowe funkcji $f(x)$), stał się dwukrotnie krótszy.

Nic nie stoi na przeszkodzie, aby powtórzyć tę operację i skrócić początkowy przedział czterokrotnie. Można to powtarzać i powtarzać, aż długość przedziału $(b - a)$ zejdzie poniżej ustalonej wartości (oznaczmy ją *epsilon*), która pełni rolę dokładności. Taka metoda wyzna-

[¶]Bernard Bolzano (1781-1848) – czeski matematyk i filozof, Augustin-Louis Cauchy (1789-1857) – francuski matematyk, inżynier i fizyk, oraz Jean Gaston Darboux (1842-1917) – francuski matematyk. Niezły zestaw nazwisk! Kocurro pamięta z kursu analizy matematycznej na studiach fizycznych na „ujocie", że większość twierdzeń z tej dziedziny nosiła miano Cauchy'ego-*kogoś tam*, *kogoś tam*-Cauchy'ego, względnie samego Cauchy'ego (wymawiaj: *kosziego*).

czania przybliżonej wartości pierwiastka nosi nazwę *metody bisekcji* lub *metody połowienia.*‖

Możemy już pisać kod funkcji `bisection()` realizującej powyższy algorytm:

```
double bisection(double a, double b, double epsilon)
{
  while(b - a >= epsilon)
  {
    double c = (a + b) / 2;
    if(f(a) * f(c) < 0)
      b = c;
    else
      a = c;
  }
  . . .
}
```

Jak widać, sprawdzanie znaków realizujemy przez sprawdzanie znaku iloczynu wartości funkcji (tak najprościej): jeśli znaki są przeciwne, wtedy iloczyn jest ujemny.

Pozostaje jeszcze ustalić, jaką wartość powinna zwrócić funkcja `bisection()`. Najlepszym kandydatem na przybliżoną wartość pierwiastka jest środek przedziału $[a, b]$ (już po tych wszystkich połowieniach), czyli $\frac{a+b}{2}$. **

Cały program demonstrujący działanie metody bisekcji dla naszej wybranej funkcji prezentuje się następująco:

```
#include <bits/stdc++.h>
using namespace std;

double f(double x)
{
  return cos(x) - x;
}

double bisection(double a, double b, double epsilon)
{
  while(b - a >= epsilon)
  {
    double c = (a + b) / 2;
    if(f(a) * f(c) < 0)
      b = c;
    else
      a = c;
  }
  return (a + b) / 2;
```

‖Metoda ta jest blisko spokrewiona z metodą wyszukiwania binarnego (ang. **binary search**, wymawiaj: *bajnary sercz*), której poświęcamy oddzielny podrozdział.

**Notabene, można by się zastanowić, z jaką faktyczną dokładnością obliczyliśmy wartość pierwiastka. Otóż błąd nie przekracza połowy długości przedziału $[a, b]$, czyli wielkości $\frac{b-a}{2}$ (dlaczego?).

```
}

int main()
{
  cout.precision(10);
  cout << bisection(0, M_PI / 2, 1e-12) << '\n';
  return 0;
}
```

Stała `M_PI` oznacza wartość liczby π, zaś stała `1e-12` oznacza 10^{-12}. Na ekranie pojawi się wartość:

```
0.7390851332
```

Na koniec przypomnijmy, że stosowanie metody bisekcji jest możliwe tylko wtedy, gdy określiliśmy przedział liczbowy, w którym znajduje się dokładnie jeden pierwiastek (nazywa się to *oddzielaniem pierwiastka*). Istnieją metody, które tego nie wymagają, na przykład metoda Newtona-Raphsona (metoda stycznych).