Program wykładu

Pewne algorytmy symboliczne

w r. akad. 2018/2019, semestr letni

Prowadzący: Paweł Woźny
Wykład: 30, ćwiczenia: 15, pracownia: 15.

Wymagania

- analiza matematyczna,
- umiejętność programowania w dowolnym języku.

Opis

Pakiety obliczeń symbolicznych są niezwykle popularne od wielu lat. Możliwości jakie dają najbardziej znane, czyli Maple i Mathematica, są naprawdę imponujące. Programy te pozwalają m.in. na symboliczne sumowanie, różniczkowanie, całkowanie, obliczanie granic, rozwiązywanie równań różniczkowych, całkowych i rekurencyjnych. Nie są im obce też metody symbolicznego rozwiązywania problemów z zakresu algebry, geometrii, statystyki, matematyki dyskretnej, algorytmiki i wielu innych, mniej lub bardziej znanych, działów matematyki. Można przy ich pomocy dokonywać obliczeń naukowych i inżynieryjnych (także numerycznych i to z dowolną dokładnością), a ich wyniki efektownie wizualizować.

Osoby interesując się informatyką już po krótkiej *zabawie* takimi programami muszą zadać sobie pytanie: *No dobrze, ale jak to wszystko działa?* Jak można się spodziewać, sprawa nie jest prosta. Algorytmy stosowane w pakietach obliczeń symbolicznych są skomplikowane i bazują na zaawansowanych teoriach matematycznych, a ich implementacja jest trudna. Są jednak i takie, które leżą w naszym *zasięgu*: ich idee można wyjaśnić bez odwoływania się do skomplikowanej matematyki.

Celem wykładu będzie przystępne przedstawienie pewnych algorytmów symbolicznych. Swoją uwagę skupimy głównie na metodach związanych z tzw. *sumowaniem symbolicznym.* Słuchacze dowiedzą się np. skąd Maple wie, że

$$\sum_{k=0}^{n} k^2 = \frac{n(n+1)(2n+1)}{6},$$

czego można się akurat domyśleć, czy, że

$$\sum_{k=0}^{n} 4^{-k} \binom{2n}{2k} \binom{2k}{k} = 2^{1-2n} \binom{4n-1}{2n-1} \qquad (n = 1, 2, \ldots),$$

co wydaje się już bardziej skomplikowane.

Jak się też przekonamy, pakiety te potrafią *same produkować* proste dowody poprawność nawet bardzo skomplikowanych tożsamości, np. takich:

$$\sum_{k=0}^{n} \frac{(-n)_k(d)_k(n+a)_k(1+\frac{d}{2})_k(d+b-a)_k(d+c-a)_k(1+a-b-c)_k}{k!(\frac{d}{2})_k(1+a-b)_k(1+a-c)_k(1+d+n)_k(b+c+d-a)_k(1+d-a-n)_k} = \\ = \frac{(b)_n(c)_n(d+1)_n(1+2a-b-c-d)_n}{(a-d)_n(1+a-b)_n(1+a-c)_n(b+c+d-a)_n},$$

gdzie $(a)_k := a(a+1)\cdots(a+k-1)$ (nie, nie używają indukcji!).

Przekazana wiedza może być szczególnie przydatna osobom interesującym się m.in. algorytmiką, matematyką dyskretną czy kombinatoryką. Wszystko zaczniemy jednak od krótkiego kursu Maple'a, który będzie naszym podstawowym narzędziem...

Program

1. Krótki kurs Maple'a.
2. Funkcja Gamma i jej własności.
3. Podstawowe wiadomości o związkach rekurencyjnych.
4. Tożsamości hipergeometryczne.
5. Metoda Siostry Celine.
6. Algorytm Gospera.
7. Algorytm Zeilbergera.
8. Metoda WZ.
9. Algorytm Petkovšeka.
10. Pewne uogólnienia poznanych metod.

Literatura

[1] W. Koepf, *Hypergeometric summation. An algorithmic approach to summation and special function identities*, Vieweg Verlag, 1998.

[2] M. Petkovšek, H. S. Wilf, D. Zeilberger, *A=B*, A. K. Peters, Wellesley, 1996.

P.S. Tak, to prawda. Ten wykład nie ma nic wspólnego z *numerkami*.

21 lutego 2019 r.