



Korespondencyjny Kurs Przygotowawczy
z matematyki
dla kandydatów na wyższe studia techniczne

Kurs trwa do kwietnia 1985r. Włącznie. Polega on na rozwiązywaniu prac kontrolnych. Każda praca kontrolna składa się z 8 zadań, których stopień trudności odpowiada wymaganiom na egzaminie wstępnym. Tematy prac kontrolnych drukowane są w "Głosie Wybrzeża", "Sztandarze Młodych" i "Trybunie Robotniczej" /pierwszy wtorek miesiąca/ oraz w "Gazecie Robotniczej" i "Życiu Warszawy" /pierwszy czwartek miesiąca/. Tam też jest podany adres i sposób przesyłania rozwiązań, by otrzymać je z powrotem poprawione i ocenione wraz z pisemną konsultacją.

Korespondencyjny Kurs Przygotowawczy z matematyki jest zorganizowany w oparciu o kilkunastoletnie doświadczenia. Z tej powszechnie dostępnej formy autentycznej pomocy dla kandydatów na studia skorzystało już w latach ubiegłych kilkadziesiąt tysięcy osób.

SPIS TREŚCI

NUMERU 12 (132)

O algorytmie <i>prof. dr Władysław M. Turski</i>	str. 1
Jak przepłynąć labirynt, Symetrie, Wieża Hanoi	str. 4
Najkrótsza droga	str. 5
Jak kojarzyć trwałe małżeństwa	str. 6
Czy przez telefon można grać w karty	str. 7
Mała Delta	str. 8
O konkursie 7.11 raz jeszcze	str. 10
Zadania	str. 10
Przepis na anaglif	str. 11
Szybka transformacja Fouriera <i>doc. dr Antoni Kreczmar</i>	str. 12
Patrz w niebo	str. 14
Kalendarz	str. 15
Klub 44	str. 16

W następnym numerze:
Węzły

Od stycznia
do grudnia 1985r.
w "Delcie"

duża MAPA NIEBA

2000 gwiazd
tory planet
i komety Halleya

„Delta”
matematyczno-fizyczno-astronomiczny
miesięcznik popularny
Polskiego Towarzystwa
Matematycznego, Polskiego
Towarzystwa Fizycznego i Polskiego
Towarzystwa Astronomicznego
wydawany przy poparciu
Ministerstwa Oświaty i Wychowania

Komitet Redakcyjny
dr Jerzy Brojan
dr Maciej Bryński
dr Bogdan Cichoński
dr Alicja Derkowska
doc. dr Jan A. Gaj
doc. dr Bolesław Gleichgewicht
doc. dr Tadeusz Jarzembowski
doc. dr Marcin Kubiak
mgr Andrzej Mąkowski
dr Zbigniew Plochocki — v-przewodniczący
dr Jan Rempala
prof. dr Konrad Rudnicki
prof. dr Grzegorz Sitarski
prof. dr Józef I. Smak
prof. dr Kazimierz Stępień
prof. dr Mieczysław Subotowicz
dr Michał Szurek
doc. dr Andrzej Szymacha
doc. dr Aniela Wolska

prof. dr Andrzej Woszczyk
prof. dr Wojciech Zakowski —
przewodniczący

Redaguje kolegium w składzie:
mgr inż. Krzysztof Biesaga
mgr Maciej Jędrzejczak — z-ca red. nac.
mgr Krystyna Kordos — sekr. red.
dr Marek Kordos — red. nac.
dr Tomasz Kwast — z-ca red. nac.
mgr Andrzej Majhofer
dr inż. arch. Jacek Mazur
mgr Anna Rudnik
dr Jerzy Ryll
mgr Joanna Udalska
mgr Jan Zalewski

Adres Redakcji
ul. Koszykowa 6a
00-564 Warszawa

Krajowe Wydawnictwo Czasopism
RSW „Prasa—Książka—Ruch”
ul. Noakowskiego 14
00-666 Warszawa
Nakład 35 000 egz. Objętość 2 ark. wyd;
2,50 ark. druk;
papier offsetowy V kl. 70 g.
Wydrukowano w drukarni
im. Rewolucji Październikowej
Warszawa, ul. Mińska 65.
Nr zam. 6825/12/84. T-70

WARUNKI PRENUMERATY

- Cena prenumeraty rocznej zł 240,— cena prenumeraty półrocznej zł 120,—
- dla osób prawnych — instytucji i zakładów pracy:
— instytucje i zakłady pracy zlokalizowane w miastach wojewódzkich i pozostałych miastach, w których znajdują się siedziby oddziałów RSW „Prasa-Książka-Ruch”, zamawiają prenumeratę w tych oddziałach,
— instytucje i zakłady pracy zlokalizowane w miejscowościach, gdzie nie ma oddziałów RSW „Prasa-Książka-Ruch” i na terenach wiejskich opłacają prenumeratę w urzędach pocztowych i u doręczycieli.
 - dla osób fizycznych — indywidualnych prenumeratorów:
— osoby fizyczne zamieszkałe na wsi i w miejscowościach, gdzie nie ma oddziałów RSW „Prasa-Książka-Ruch”, opłacają prenumeratę w urzędach pocztowych i u doręczycieli,
— osoby fizyczne zamieszkałe w miastach — siedzibach oddziałów RSW „Prasa-Książka-Ruch” opłacają prenumeratę wyłącznie w urzędach pocztowych nadawczo-oddawczych właściwych dla miejsca zamieszkania prenumeratora. Wpłaty dokonują używając „blankietu wpłaty” na rachunek bankowy miejscowego oddziału RSW „Prasa-Książka-Ruch”.
 - Prenumeratę ze zleceniem wysyłki za granicę przyjmuje RSW „Prasa-Książka-Ruch”, Centrala Kolportażu Prasy i Wydawnictw, ul. Towarowa 28, 00-958 Warszawa, konto NBP XV Oddział w W-wie Nr 1153-201045-139-11. Prenumerata ze zleceniem wysyłki za granicę pocztą zwykłą jest droższa od prenumeraty krajowej o 50% dla zleceniodawców indywidualnych i o 100% dla zlecających instytucji i zakładów pracy.
- Terminy przyjmowania prenumeraty na kraj i za granicę:
— do dnia 10 listopada na I kwartał, I półrocze roku następnego oraz cały rok następny,
— do dnia 1-go każdego miesiąca poprzedzającego okres prenumeraty roku bieżącego,

Sprzedaj numerów bieżących i uprzednich

Instytucje państwowe i społeczne, zakłady pracy, szkoły i czytelnicy indywidualni mogą nabywać „DELTE”:
— w Księgarni Ośrodka Wydawnictw Naukowych PAN, Warszawa — Pałac Kultury,
— w Głównej Księgarni Naukowej, Warszawa — ul. Krakowskie Przedmieście 7,
— w Księgarni Ossolineum, Wrocław — Rynek 8,
— w Księgarni Naukowej, Kraków — Podwale 6.

Orders for this periodical from abroad can be placed with „Ars Polona” Krakowskie Przedmieście 7, 00-068 Warszawa, Poland or with
— Kubon & Sagner, Inhaber Otto Sagner, D8 München 34, Postfach 68, Bundesrepublik Deutschland,
— Earlcourt Publications Ltd., 130 Shepard Bush Centre, London W12, Great Britain,
— Licosa Commissionaria Sansoni, Via Lamarmora 45, 50 121 Firenze, Italia.

Cena 1 egzemplarza zł 20,—

Prof. dr Władysław M. TURSKI



Już starożytni Grecy, a przed nimi Egipcjanie, Babilończycy i inni zajmowali się praktycznie stosowaniem reguł rachunkowych, używali algorytmów. Najstynniejszy algorytm nosi imię Euklidesa, a samo pojęcie tak mocno zrosło się ze swą egzemplifikacją w „algorytmie Euklidesa”, że przez pewien czas uważano, iż słowo „algorytm” ma grecki źródłostów: wywodzono je od *algiros* (= bolesny, trudny) i *arithmos* (= liczba). Prawda jest jednak inna, słowo „algorytm”, pierwotnie oznaczające sztukę wykonywania obliczeń przy użyciu cyfr „arabskich”, pochodzi z arabskiego, dokładniej — od nazwiska Abu Dżafara Mohammeda ibn Musy al Horizmiego, w którym to kwiecistym nazwisku obok imion przodka i potomka występuje imię własne (Mohammed) i miasto (Horizm, dzisiejszy Chorezm w radzieckiej Azji Środkowej), z którego pochodził słynny matematyk bagdadzki, autor sławnej książki *Kitab al dżabr ʿal mukabala* (z tego tytułu wzięta się „algebra”!), z której to właśnie książki Europa nauczyła się liczyć używając cyfr „arabskich”. Jednym słowem, zupełna arabszczyzna, wyjąwszy, naturalnie, same cyfry, które pochodzą z Indii.

Zanim Europejczycy posiadli trudną sztukę wykonywania obliczeń w systemie dziesiętnym, posługiwali się rzymskimi abakami, czyli tabliczkami podzielonymi na przegródki, do których wkładano kamyki, czyli *calculi* (stąd „kalkulować”!). Ekspersi od takich rachunków nosili miano *abacystów*, uczniowie (pośredni!) *Horezmijczyka* — algorytmistów. Zwycięstwo algorytmistów nie przyszło szybko, nawet w drukowanych już książkach można znaleźć drzeworyty przedstawiające konkursy sprawnościowe z udziałem reprezentantów obydwu szkół. A więc wdrażanie algorytmów arytmetycznych trwało w Europie ponad 600 lat! (Książka *Kitab ...* powstała około roku 820). Warto o tym pamiętać, gdy sarka się na powolne tempo wdrażania nowych metod postępowania. Ma się te tradycje!

Dziś słowo „algorytm” ma tak wiele znaczeń, zależnych od okoliczności, że trudno byłoby dać uniwersalną, a jednocześnie precyzyjną wykładnię jego treści. Najogólniej (a przeto niezbyt precyzyjnie) powiedziawszy, algorytm to autonomicznie wyrażony przepis na osiągnięcie zamierzonego celu.

Zauważmy, że nie każdy przepis na osiągnięcie zamierzonego celu jest algorytmem: musi on być autonomicznie wyrażony, czyli sformułowany tak, aby mógł być stosowany niezależnie. To, od czego ma zachodzić owa niezależność, jest sprawą dość niejasną, a uściślanie tego aspektu pojmowania „algorytmu” prowadzi do wielu ciekawych, aczkolwiek trudnych do ujęcia we wspólne ramy, rozważań.

Łatwiej będzie zilustrować o co nam chodzi na przykładzie: nie jest algorytmem wewnętrznym „przepis postępowania” myszy, która znajduje drogę do kawałka sera ukrytego w głębi labiryntu. Istnieją natomiast dziesiątki algorytmów ustalających lepsze czy gorsze postępowanie prowadzące do znalezienia drogi w labiryncie. (Wiele z tych algorytmów zostało zrealizowanych w postaci programów sterujących różnymi mechanicznymi „myszkami”, które np. co roku współzawodniczą o tytuł najlepszej sztucznej myszy). Różnica polega na owym „autonomicznym wyrażeniu”. Nie znamy pełnego opisu postępowania myszy naturalnej, nie wiemy nawet, jakie czynniki na nie wpływają, co mysz uwzględniła decydując się na taki czy inny ruch mięśni, nie wiemy nawet, w jakim stopniu jej postępowanie zależy od odbieranych bodźców zewnętrznych (i jakich), a w jakim stopniu od pamięci, „świadomej” i „podświadomej”, a także od „wbudowanych” odruchów ścięgien, a nawet poszczególnych komórek. Postępowanie sztucznej myszy jest natomiast całkowicie opisane jawnymi formułami, nawet jeśli opisują one czasem zachowanie niedeterministyczne, bo i wtedy zakres tudzież mechanizm owego niedeterminizmu są wyraźnie opisane (podobnie możemy powiedzieć, że algorytm gry w „chińczyka” jest niedeterministyczny — pewne czynności gracza zależą od wyniku rzutu kostką).

Aby autonomiczność wyrażenia przepisu miała jakkolwiek sens, musi być określony — lub dany domyślnie — pewien układ odniesienia, względem którego formułuje się ów przepis. Taki układ odniesienia nazywa się często dziedziną algorytmiczną, a w jego skład wchodzi obiekty, być może pogrupowane w kategorie, różniące się pod względem tych czy innych cech, ustalone relacje

między obiektami, być może uwzględniające podział obiektów na kategorie, a także pewien repertuar operacji określonych na obiektach, znów być może z uwzględnieniem kategorii.

Dziedziną algorytmu Euklidesa jest zbiór liczb całkowitych nieujemnych, ze zwykłymi relacjami równości, mniejszości itp., oraz z operacjami na liczbach całkowitych.

Jeśli wśród tych operacji dopuścimy występowanie operacji dzielenia z resztą, algorytm Euklidesa może mieć postać:

Dane są dwie liczby całkowite dodatnie X i Y .

Niech R będzie resztą z dzielenia X przez Y .

Jeśli $R = 0$, to szukany wynikiem (największym wspólnym dzielnikiem) jest liczba Y i kończymy wykonywanie algorytmu.

Jeśli $R \neq 0$, to przyjmujemy za nową wartość X uprzednią wartość Y , a za nową wartość Y otrzymaną wartość R i powtarzamy całe postępowanie od początku.

Jeśli natomiast operacja dzielenia z resztą nie występuje w naszej dziedzinie, algorytm Euklidesa przyjmie postać:

Dane są dwie liczby całkowite dodatnie X i Y .

Jeśli $X = Y$, to szukany wynikiem jest liczba X i kończymy wykonywanie algorytmu.

Jeśli $X \neq Y$, to niech M będzie większą z liczb X , Y , a m — mniejszą.

Obliczamy $R = M - m$.

Za nową wartość X przyjmujemy obliczoną wartość R , a za nową wartość Y — wartość m i powtarzamy całe postępowanie od początku.

Bez trudu można zauważyć, że w większości przypadków pierwsza wersja algorytmu wymaga mniejszej liczby kroków niż druga. Wynika to stąd, że podstawowa robocza operacja wersji pierwszej — dzielenie z resztą — jest „mocniejsza” niż robocza operacja wersji drugiej — odejmowanie.

Liczba operacji, które należy wykonać, aby osiągnąć cel algorytmu, jest jedną z ważniejszych jego własności. Nie jest to jednak własność najważniejsza. Podstawowe pytanie dotyczące każdego algorytmu odnosi się do jego poprawności: czy można zagwarantować, że wiernie przestrzegając przepisu wyrażonego w algorytmie zawsze osiągniemy żądany cel. Praktycznie, oczywiście, interesują nas tylko takie algorytmy, co do których gwarancja jest jeszcze mocniejsza, a mianowicie, że cel osiągniemy w skończonej liczbie kroków. Pytanie to można zresztą postawić inaczej: czy można zagwarantować, że jeśli wykonanie algorytmu kończy się (po skończonej liczbie kroków przepis dyktuje zaprzestanie wykonywania dalszych działań), to cel zostaje osiągnięty. Jeśli odpowiedź na to pytanie jest twierdząca, algorytm nazywamy częściowo poprawnym. Jeśli ponadto można zagwarantować, że wykonanie algorytmu zawsze się kończy, to nazywamy go po prostu poprawnym albo całkowicie poprawnym.

Dowód tego, że wykonywanie algorytmu zawsze się kończy, jest często niezależny od dowodu, iż kończąc się osiąga pożądaną cel. Na przykład wykonanie drugiej wersji algorytmu Euklidesa musi się zawsze zakończyć, gdyż w każdym kroku bieżąca dodatnia wartość M jest zmniejszana o dodatnią liczbę całkowitą, mniejszą od M . Ponieważ proces zaczyna się ze skończoną wartością M , na pewno nie może liczyć więcej niż owo początkowe M kroków. Dowód tego, że wykonujący algorytm do końca osiągamy zamierzony cel, wymaga spostrzeżenia dwu faktów:

1. z tego, że wykonanie algorytmu dobiegło końca, wynika, iż bieżące wartości X i Y spełniają $X = Y$,
2. z własności największego wspólnego dzielnika (NWD) dwu liczb całkowitych wynika, dla dowolnych dodatnich $A > B$, że $\text{NWD}(A, B) = \text{NWD}(A - B, B)$.

Pierwszy z tych faktów jest konsekwencją samego algorytmu, drugi jest własnością dziedziny, której algorytm dotyczy. Mamy tu do czynienia z bardzo typowym zjawiskiem: dowód poprawności algorytmu zależy zarówno od własności jego konstrukcji, jak i od własności dziedziny algorytmicznej.

Podobnie, na konstrukcję — projektowanie — algorytmu wpływają dwa, na ogół rozłączne, zespoły wiedzy: jeden, dotyczący zasad budowy poprawnych i sprawnych algorytmów i drugi, ściśle związany z dziedziną, dla której pisze się ów algorytm. Czasem owa dziedzina ma charakter formalny, jak w przypadku algorytmu Euklidesa, jednakże w wielu praktycznie interesujących przypadkach jest nią pewna dziedzina wiedzy albo zgoła umiejętność, nie mająca adekwatnej reprezentacji formalnej. Sytuacja taka występuje na przykład wtedy, gdy chcemy ułożyć algorytmy dla banku, biura podróży czy towarzystwa ubezpieczeniowego, nie mówiąc już o takich zastosowaniach, jak przekład z jednego języka potocznego na inny albo zarządzanie gospodarcze.

Rozwiązanie zadania M 385.

Jeżeli $(a + b\sqrt{3})^2 = c + d\sqrt{3}$, to $(a - b\sqrt{3})^2 = c - d\sqrt{3}$ i $(a + b\sqrt{3})^2(a - b\sqrt{3})^2 = (c + d\sqrt{3})(c - d\sqrt{3}) = c^2 - 3d^2$. Gdyby więc istniały a i b żądane w treści zadania, to $(99999)^2 - 3(100000)^2 < 0$ byłoby kwadratem liczby $a^2 - 3b^2$.

Inne rozwiązanie: Gdyby było $99999 + 100000\sqrt{3} = (a + b\sqrt{3})^2$, mielibyśmy $a^2 + 3b^2 = 99999$ i $2ab = 100000$.

Z pierwszej z tych równości wynika, że a dzieli się przez 3, z drugiej zaś, że się nie dzieli, co daje sprzeczność.



Użyteczność i przydatność stosowania komputerów polega na tym, że są one urządzeniami w zasadzie bezbłędnie wykonującymi algorytmy. Naturalnie, po to, aby algorytm mógł być wykonany przez komputer, należy ten algorytm przedstawić w pewien określony sposób, wyrazić w języku programowania. Kwestia wyrażenia algorytmu w języku programowania, aczkolwiek niezmiernie ważna z „wewnątrz—informatycznego” punktu widzenia, jest jednak czysto techniczna w tym sensie, że nie stanowi ona żadnej istotnej bariery przed możliwością komputerowego rozwiązania problemu. Barię taką stanowi natomiast brak odpowiednich algorytmów, a także, niekiedy, nadmierna złożoność obliczeniowa istniejących algorytmów (przez „złożoność obliczeniową” algorytmu rozumie się oszacowanie kosztu jego realizacji, mierzonego liczbą wykonywanych operacji).

Informatyka bada metody budowy, analizy i przekształcania algorytmów dla dziedzin, które są na tyle dobrze opisane sformalizowanymi związkami, że stosując do nich (tj. dziedzin) zasady obliczalnych wnioskowań, nie należy obawiać się popełnienia grubego błędu. Stosowanie algorytmicznych metod postępowania w dziedzinach nie spełniających tego warunku jest błędem metodologicznym, nie dającym się zazwyczaj wykryć w żaden sposób przed tragiczną nieraz konfrontacją wyników takiego postępowania z rzeczywistością. Niestety, pokusa zastosowania komputerów w dziedzinach pozornie tylko albo połowicznie sformalizowanych bywa zbyt wielka.

Oprócz konstrukcji, analizy i przekształcania algorytmów, informatyka zajmuje się także algorytmami jako tworam abstrakcyjnymi. Badania takie mają fundamentalne znaczenie poznawcze. Próbuje się znaleźć odpowiedzi na takie na przykład pytania: Czy zawsze wtedy, gdy dany problem dopuszcza rozwiązanie algorytmiczne, istnieje algorytm optymalny dla tego zadania (np. algorytm najszybszy, o najmniejszej złożoności obliczeniowej)? Jaką klasę funkcji można obliczyć za pomocą algorytmów zbudowanych z danego zestawu operacji? Czy zmiana modelu realizacji algorytmu (np. wzbogacenie go o obliczenia niedeterministyczne) zmienia w istotnym stopniu zakres zadań, które można rozwiązać algorytmicznie?

Widzimy więc, że problematyka algorytmów — autonomicznych przepisów efektywnego postępowania — obejmuje całe spektrum zainteresowań matematyki: od najbardziej podstawowych, filozoficznych niemal, do całkowicie praktycznych, by nie rzec: utylitarnych. Niektórzy powiadają, że prawdziwa matematyka to właśnie studium algorytmów. Jak zwykle przy takich mocnych uogólnieniach nie wszyscy zechcą je podzielić. W każdym razie trudno nie zgodzić się z tym, że studium algorytmów to prawdziwie pasjonujące zajęcie i kawał pięknej matematyki.



Rozwiązanie zadania M 383.

Przypuśćmy, że ta sama liczba pojawia się dwukrotnie po raz pierwszy w wierszu $n+1$. Oznaczmy ją przez r . Ponieważ wszystkie liczby w n -tym wierszu są z założenia różne, więc $r = u^2 = v+1$ dla pewnej pary u, v elementów n -tego wiersza. Mamy: $v = u^2 - 1$ i ponieważ największym kwadratem liczby naturalnej nie przewyższającym $u^2 - 1$ jest $(u-1)^2 = (u^2 - 1) - (2u - 2)$, więc dla otrzymania v potrzebujemy co najmniej $2u - 1$ operacji (podniesienie do kwadratu i $2u - 2$ dodawania jedynki). Wynika stąd, że $n \geq 2u + u - a \geq 2u - 1$, co jest niemożliwe, ponieważ $u > a > 1$.



Rozwiązanie zadania M 384.

Łatwo zauważyć, że części całkowite liczb x_1, \dots, x_n nie są w zadaniu istotne i możemy rozpatrywać jedynie części ułamkowe $y_1 = x_1 - [x_1], \dots, y_n = x_n - [x_n]$, przy czym możemy założyć, że $0 \leq y_1 \leq y_2 \leq \dots \leq y_n < 1$.

Wybermy teraz takie k , że $ky_k \leq \frac{n+1}{4}$

i $(k+1)y_{k+1} > \frac{n+1}{4}$ (ewentualnie $k=0$).

Ponieważ $y_{k+1} > \frac{n+1}{4(k+1)}$, więc

$$(n-k)(1-y_{k+1}) < (n-k)\left(1 - \frac{n+1}{4(k+1)}\right).$$

$$\text{Nierówność } (n-k)\left(1 - \frac{n+1}{4(k+1)}\right) \leq \frac{n+1}{4}$$

jest równoważna $(n-k)(4k-n+3) \leq (n+1)(k+1)$, a ta jest równoważna nierówności $(n-(2k+1))^2 \geq 0$ spełnionej dla dowolnych k, n .

Jeśli teraz y_1, \dots, y_k zaokrąglimy do 0, a y_{k+1}, \dots, y_n do 1, to największy błąd przy zastępowaniu składników sumy ich zaokrągleniami możemy popełnić rozpatrując sumy $y_1 + \dots + y_k$ i $y_{k+1} + \dots + y_n$. W pierwszym przypadku błąd wyniesie najwyżej ky_k , a w drugim najwyżej $(n-k)(1-y_{k+1})$.

Wynika stąd, że dla podanego wyżej wyboru

k zarówno $ky_k \leq \frac{n+1}{4}$, jak

$$\text{i } (n-k)(1-y_{k+1}) \leq \frac{n+1}{4}.$$

Jak przepłynąć labirynt?

Najpierw należy napęłnić labirynt wodą, a potem wskoczyć do niego. Po wodzie rozejdą się kręgi. Pierwszy, który dotrze do punktu docelowego, nie tylko pokaże, że można tam dopłynąć, ale na dodatek przebędzie najkrótszą z możliwych dróg. Ten prosty eksperyment, bardziej intuicyjny niż fizyczny (brak przecież w jego opisie wielu założeń co do fizycznych własności labiryntu) łatwo przekłada się na precyzyjny język liczb i algorytmów.

	5	4	3	3	2	2	2	3
		5	5	2	3	1	3	2
		4		2	1	S	1	2
		4	3	2	1	1	5	5
		4			2	2	4	5
C	5	5			3	3	4	5

Narysujmy labirynt na kartce kratkowanego papieru. Obierzmy w nim punkt startowy i docelowy. Ponumerujmy kolejnymi liczbami naturalnymi kręgi, rozchodzące się wokół startu. Pamiętajmy, że ścianki korytarzy labiryntu całkowicie pochłaniają dochodzące do nich kręgi, a w każdym korytarzu mogą się one rozchodzić tylko jeden raz.

Jeżeli zapełnimy w ten sposób labirynt i nie dojdziemy do celu, to poszukiwanej drogi nie ma. W przeciwnym przypadku — gdy dotrzemy wreszcie do miejsca docelowego — droga istnieje, a znaleźć ją łatwo cofając się tak, by przeskakiwać z jednego kręgu na najbliższy, o numerze o jeden niższym.

Opisany tu algorytm nie na wiele zdałby się Tezeuszowi — nie miał on przecież planu labiryntu kreteńskiego. Wykorzystywany jest za to przez twórców programów komputerowych przeznaczonych do projektowania połączeń między elementami układów elektronicznych.

Na koniec pytanie dla dociekliwych:

W jaki inny sposób można numerować kręgi w labiryncie, by odnaleźć drogę do celu? Jak użyć do tego najmniejszej liczby różnych liczb?

K. B.

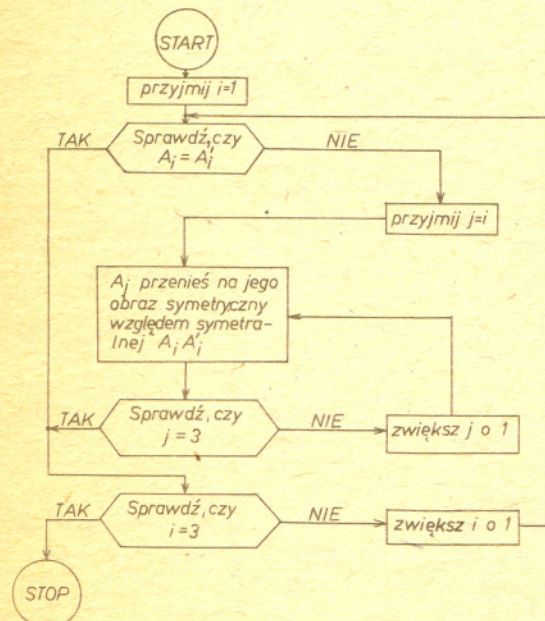
Symetrie

Poniższy algorytm stanowi konstruktywny dowód twierdzenia:

Dowolne dwa przystające trójkąty na płaszczyźnie można nałożyć za pomocą nie więcej niż trzech symetrii osiowych.

Jeśli znajdziemy się w punkcie START z dwoma przystającymi trójkątami $A_1 A_2 A_3$ i $A'_1 A'_2 A'_3$, to w punkcie STOP będziemy mieli $A_1 = A'_1$ dla $i = 1, 2, 3$.

Czytelnik zechce sprawdzić, że tak jest istotnie, a także uzasadnić, że jest to dowód przytoczonego twierdzenia.



Wieże Hanoi

Gdy w pępku świata, indyjskim mieście Benares, stawał Brahma na brązowej tabliczce trzy diamentowe pałeczki o wysokości jednego łokcia i o grubości tułowia pszczoły, wiedział już zapewne w swej niezmierzonej mądrości, że nim mniś przeloży 64 złote krążki i nastąpi koniec świata, ludzie będą podawać grę w „Wieże Hanoi” jako koronny przykład algorytmu rekursywnego.

Algorytm — to składający się ze skończonego ciągu elementarnych poleceń przepis na wykonanie pewnej bardziej złożonej czynności. Jedną z własności większości algorytmów jest to, że do działania niezbędny jest im zestaw produktów wejściowych, które my dalej będziemy nazywali parametrami. Jak z opisu tego widać, cały algorytm jest też pewnym poleceniem. Nic zatem nie stoi na przeszkodzie, aby go użyć jako elementu innego, bardziej złożonego polecenia. Stąd już tylko jeden krok do algorytmu rekursywnego, czyli takiego, który jako jednego z elementów używa samego siebie.

Przejdźmy do algorytmu gry w „Wieże Hanoi”. Przypomnijmy w skrócie zasady gry:

1. rekwizytami są trzy pałeczki (nazwijmy je A , B i C), na które nanizano krążki o parami różnych średnicach,
2. w stanie początkowym na pałeczce A znajduje się n krążków ułożonych tak, że każdy (oprócz najniższego) leży na krążku o większej średnicy,
3. zadanie polega na przeniesieniu wszystkich krążków, jednego po drugim, z pałeczki A na pałeczkę C z wykorzystaniem pałeczki B . Należy przy tym przestrzegać zasady, że krążek można przelożyć albo na pałeczkę pustą, albo na inny krążek, byle o większej średnicy.

Przystąpmy zatem do budowy algorytmu. Nadajmy mu nazwę i określmy jego parametry. Będziemy zapisywali:

Hanoi (n , *skąd*, *poprzez*, *dokąd*)

i odczytywali:

przenieś, według zasad gry, n krążków z pałeczki o nazwie *skąd* na pałeczkę o nazwie *dokąd*, wykorzystując pałeczkę *poprzez*.

W naszym algorytmie użyjemy tylko jednego elementarnego polecenia. Będziemy zapisywali:

przenieś (*skąd*, *dokąd*)

i odczytywali:

przenieś jeden krążek z pałeczki o nazwie *skąd* na pałeczkę o nazwie *dokąd*.

Algorytm *Hanoi* (n , *skąd*, *poprzez*, *dokąd*) ma działać dopóty, dopóki na pałeczce początkowej jest jeszcze choć jeden krążek. Aby grę zakończyć, wystarczy wykonać kolejno tylko trzy polecenia. Zapiszmy je od razu w postaci gotowego algorytmu używając wprowadzonych poprzednio oznaczeń.

Algorytm *Hanoi*

parametry: n liczba przenoszonych krążków

skąd nazwa pałeczki początkowej

poprzez nazwa pałeczki pośredniej

dokąd nazwa pałeczki końcowej

1. jeżeli $n > 0$:
 - a. *Hanoi* ($n-1$, *skąd*, *dokąd*, *poprzez*)
 - b. *przenieś* (*skąd*, *dokąd*)
 - c. *Hanoi* ($n-1$, *poprzez*, *skąd*, *dokąd*)
 - d. koniec
2. jeżeli $n = 0$: koniec.

Zauważcie, jak prosty jest ten algorytm. Jako jedynego rzeczywistego działania używa on polecenia *przenieś*. Pozostałe kroki służą jedynie do organizowania rekursji i modyfikacji parametrów.

Na zakończenie słów kilka o końcu świata: kiedy on nastąpi? Jeżeli przez $P(n)$ oznaczymy liczbę elementarnych przeniesień dla gry z n krążkami, to ze struktury algorytmu widać, że:

$$P(n) = \begin{cases} 0 & \text{dla } n = 0, \\ 2P(n-1) + 1 & \text{dla } n > 0. \end{cases}$$

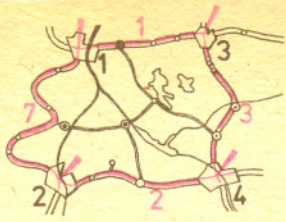
Radzimy dociekliwemu Czytelnikowi wykazać, że $P(n) = 2^n - 1$. My zaś pokażemy, że koniec świata nie może nastąpić wcześniej niż po $2^n - 1$ (gdzie $n = 64$) przelożeniach krążka. Oznaczmy przez $p(n)$ minimalną możliwą liczbę przelożeń dla n krążków. W momencie gry, w którym należy przenieść największy krążek z pałeczki początkowej na końcową, na pałeczce pośredniej musi być ułożona piramida z $n-1$ krążków. By stan ten uzyskać, trzeba wykonać zatem co najmniej $p(n-1)$ przelożeń. Teraz należy przelożyć krążek największy i znowu wykonać $p(n-1)$ ruchów. Stąd:

$$P(n) \geq 2p(n-1) + 1 \geq 2^n - 1 \quad (\text{gdzy } p(0) = 0).$$

Przy 64 krążkach mamy zatem jeszcze sporo czasu, nim mniś wykonają ostatnie z prawie 10^{20} przemieszczeń krążków.

K. B.

Najkrótsza droga



Mamy pewien zbiór miast i sieć łączących je dróg. Niech n oznacza liczbę miast, a l_{ij} długość drogi łączącej bezpośrednio miasto i z miastem j ($l_{ii} = 0$); jeżeli między tymi miastami nie biegnie droga, przyjmujemy $l_{ij} = \infty$. Pokażemy dwie metody obliczania długości najkrótszego połączenia drogowego dwóch dowolnych miast; dla miasta i oraz miasta j oznaczymy ją przez c_{ij} .

Przykład

Rozważmy cztery miasta połączone drogami tak, jak to przedstawiono na rysunku.

Metoda 1 (Warshalla)

polega na obliczaniu długości c_{ij}^k najkrótszego połączenia miast i oraz j takiego, że przechodzi ono jedynie przez miasta o numerach nie większych niż k .

Algorytm

- dla każdej pary miast (i, j) przyjmij $c_{ij}^0 = l_{ij}$,
- dla kolejnych liczb k z przedziału $[1, n]$ i dla każdej pary miast (i, j) , przyjmij $c_{ij}^k = \min(c_{ij}^{k-1}, c_{ik}^{k-1} + c_{kj}^{k-1})$,
- dla każdej pary miast (i, j) przyjmij $c_{ij} = c_{ij}^n$,
- koniec algorytmu.

Algorytm Warshalla buduje następujące tabelki wartości c_{ij}^k .

c_{ij}^0	1	2	3	4
1	0	7	1	∞
2	7	0	∞	2
3	1	∞	0	3
4	∞	2	3	0

c_{ij}^1	1	2	3	4
1	0	7	1	∞
2	7	0	8	2
3	1	8	0	3
4	∞	2	3	0

c_{ij}^2	1	2	3	4
1	0	6	1	4
2	7	0	8	2
3	1	8	0	3
4	4	2	3	0

c_{ij}^3	1	2	3	4
1	0	6	1	4
2	7	0	8	2
3	1	8	0	3
4	4	2	3	0

c_{ij}^4	1	2	3	4
1	0	6	1	4
2	7	0	8	2
3	1	8	0	3
4	4	2	3	0

Metoda 2 (Dijkstry)

polega na konstruowaniu podzbioru S zbioru wszystkich miast takiego, że najkrótsze połączenie pewnego miasta i z każdym miastem ze zbioru S przebiega przez miasta ze zbioru S . Długość takiego połączenia dla miast i oraz j oznaczymy przez d_{ij} .

Algorytm

- dla każdej pary miast (i, j) przyjmij $d_{ij} = l_{ij}$,
- weź $i = 0$,
- weź kolejną liczbę naturalną i ; jeżeli $i > n$: koniec algorytmu,
- przyjmij $S = \{i\}$,
- jeżeli $S = \{1, 2, \dots, n\}$, przyjmij $c_{ik} = d_{ik}$, dla każdego $1 \leq k \leq n$ i wykonaj 3.
- jeżeli $S \neq \{1, 2, \dots, n\}$, wykonaj:
 - ze zbioru $\{1, 2, \dots, n\} - S$ wybierz miasto j , dla którego d_{ij} jest najmniejsze i przyjmij $S = S \cup \{j\}$,
 - dla każdego $k \in \{1, 2, \dots, n\} - S$ przyjmij $d_{ik} = \min(d_{ik}, d_{ij} + l_{jk})$ i wykonaj 5.

Algorytm Dijkstry buduje następujące zbiory S i tabelki wartości d_{ij} .

$S = \{1\}$	j	1	2	3	4
d_{1j}		0	7	1	∞

$S = \{1, 3\}$	j	1	2	3	4
d_{1j}		0	7	1	4

$S = \{1, 3, 4\}$	j	1	2	3	4
d_{1j}		0	6	1	4

$S = \{1, 2, 3, 4\}$	j	1	2	3	4
c_{1j}		0	6	1	4

Pozostawiamy Czytelnikowi prześledzenie działania algorytmu Dijkstry dla pozostałych miast.

Algorytmy Warshalla i Dijkstry formułuje się zwykle inaczej niż zostało to tutaj przedstawione. Działają one prawidłowo także dla dróg jednokierunkowych i dlatego rozwiązany przez nie problem można definiować ogólniej, używając pojęć z teorii grafów zorientowanych.

Oba algorytmy są niemal optymalne, gdyż liczba elementarnych operacji jest dla każdego z nich rzędu n^3 . Kerr udowodnił, że każdy algorytm obliczający najkrótsze połączenie między n miastami, przy użyciu operacji dodawania i brania minimum wymaga wykonania mniej więcej n^3 elementarnych kroków.



L. B.

Aneks do artykułu

Jak kojarzyć trwale małżeństwa

Dla danego zestawu list preferencji może istnieć wiele

stabilnych układów (nawet $2^{\frac{n}{2}}$ układów). Uzyskany w algorytmie układ $U = \{(k_1, m_1), (k_2, m_2), \dots, (m_n, k_n)\}$ jest optymalny dla mężczyzn. Dla dowolnego bowiem innego stabilnego $U' = \{(m_1, k'_1), (m_2, k'_2), \dots, (m_n, k'_n)\}$ zachodzi $k_1 m_1 k'_1, k_2 m_2 k'_2, \dots, k_n m_n k'_n$. Udowodnimy, że jeżeli w trakcie wykonywania algorytmu mężczyzna m wykreślił kobietę k ze swojej listy, to nie istnieje inny układ stabilny, w którym mogłaby znaleźć się

para (m, k) . Przypuśćmy, że tak jest aż do momentu, gdy m_0 wykreśla ze swej listy k_0 i dopiero teraz daje się utworzyć układ stabilny U' z parą (m_0, k_0) . W takim przypadku w U istnieje para (m_1, k_0) taka, że $m_1 k_0 m_0$, a w U' obok (m_0, k_0) jest (m_1, k_1) , dla której $k_1 m_1 k_0$ (inaczej bowiem U' nie będzie stabilny). Skoro zatem w U jest (m_1, k_0) i $k_1 m_1 k_0$, to k_1 musiała być skreślona wcześniej niż k_0 z listy m_1 . W tamtym momencie, wobec przyjętego na wstępie założenia, stabilny układ U' z parą (k_1, m_1) nie może istnieć.

A zatem końcowy układ jest wyznaczony jednoznacznie, niezależnie od kolejnych wyborów wolnych mężczyzn.

Oto przykładowe listy preferencji dla $\mathcal{M} = \{A, B, C, D, E\}$ i $\mathcal{K} = \{1, 2, 3, 4, 5\}$:

A	2	5	1	3	4	1	E	A	D	B	C
B	1	2	3	4	5	2	D	E	B	A	C
C	2	3	5	4	1	3	A	D	B	C	E
D	1	3	2	4	5	4	C	B	D	A	E
E	5	3	2	1	4	5	D	B	C	E	A

Dane są dwa n -elementowe zbiory — zbiór kobiet \mathcal{K} i zbiór mężczyzn \mathcal{M} . Każdemu elementowi przypisana jest pewna lista preferencji. Wyjaśnimy to pojęcie na przykładzie wybranej kobiety $k \in \mathcal{K}$. Przypisana do k lista jest to pewna permutacja wszystkich elementów zbioru \mathcal{M} . Jeżeli mężczyzna m_1 znajduje się wyżej na liście preferencji kobiety k niż mężczyzna m_2 , to piszemy $m_1 k m_2$.

Należy mężczyzn i kobiety ze zbiorów \mathcal{M} i \mathcal{K} połączyć w pary tak, aby kobieta k_1 i mężczyzna m_2 z dwóch różnych par (m_1, k_1) i (m_2, k_2) nie cenili wyżej (w sensie swych list preferencji) siebie nawzajem niż swych aktualnych partnerów. Układ spełniający powyższy warunek nazywamy stabilnym.

Najprostszy możliwy algorytm — przeglądanie po kolei wszystkich układów par i wybieranie układów stabilnych — jest mało efektywny. Wszystkich układów par jest przecież $n!$. Nie wiemy w dodatku, czy układ stabilny zawsze istnieje.

Metoda polegająca na uzyskiwaniu układu stabilnego przez wymianę partnerów w parach powodujących niestabilność również nie zawsze daje pozytywne rezultaty (patrz obok). Skonstruujemy algorytm, w którym po drodze będą występować tylko układy stabilne, ale niekoniecznie pełne, tzn. nie wszyscy mężczyźni (i nie wszystkie kobiety) muszą być połączeni w pary. Algorytm rozpocznie swoje działanie od najprostszego układu stabilnego — zbioru pustego, następnie będzie z układu stabilnego otrzymywał inny układ stabilny, aż dojdzie do układu pełnego.

(A,1) (B,3) (C,2) (D,4) (E,5) (A,2)
 (A,2) (B,3) (C,1) (D,4) (E,5) (B,2)
 (A,3) (B,2) (C,1) (D,4) (E,5) (B,1)
 (A,3) (B,1) (C,2) (D,4) (E,5) (A,1)
 (A,1) (B,3) (C,2) (D,4) (E,5) !!!
 Algorytm zapęłta się.

Do budowy algorytmu użyjemy operacji *oświadcza się*. Dla dowolnego $k \in \mathcal{K}$ i $m \in \mathcal{M}$ operacja m oświadcza się k wykonywana jest w następujący sposób:

Jeżeli kobieta k jest wolna, to tworzy się para (m, k) . W przeciwnym przypadku kobieta dokonuje wyboru tego mężczyzny, spośród m i aktualnego partnera, który znajduje się wyżej na jej liście preferencji. Jednocześnie k zostaje skreślona z listy preferencji odrzuconego mężczyzny. Wybrany mężczyzna tworzy parę z kobietą k , a odrzucony staje się wolny.

Algorytm

dopóki istnieje wolny mężczyzna, wykonuj:

1. wybierz dowolnego wolnego mężczyznę m ,
 2. wykonaj m oświadcza się k , dla k znajdującej się aktualnie najwyżej na liście preferencji m .
- koniec algorytmu

Spróbujcie zastosować ten algorytm do podanego na marginesie przykładowego zestawu list preferencji. Nietrudno zauważyć, że każdemu wykreśleniu kobiety z którejś z list towarzyszy wybór jednego z dwóch mężczyzn. W ten sposób kobieta dobiera sobie coraz lepszego partnera. Czy algorytm zatrzyma się kiedykolwiek? Tak. Każdy mężczyzna oświadcza się danej kobiecie tylko raz, a więc w najgorszym przypadku musi nastąpić moment, kiedy na liście preferencji mężczyzny m pozostanie już tylko jedna kobieta k . Inne kobiety muszą więc mieć partnerów. Kobieta k jest zatem wolna. Wykonanie operacji m oświadcza się k utworzy ostatnią parę (m, k) . Algorytm zatrzymuje się. Tylko n operacjom oświadcza się nie towarzyszy skreślenie kobiety z listy preferencji, a więc algorytm zatrzyma się po co najwyżej $n + (n-1)^2$ krokach.

Pokażemy, że kolejne układy otrzymywane w algorytmie są stabilne. Rozpatrzmy pewien układ stabilny przed zakończeniem działania naszego algorytmu w momencie tworzenia pary (m, k) . Dla mężczyzny m z pary (m, k) może istnieć kobieta k_1 taka, że $k_1 m k$. Ponieważ m oświadczał się kolejno kobietom według własnej listy preferencji, to k_1 musiała być z niej wykreślona. Zdarzyć się to mogło jednak tylko wtedy, gdy k_1 znalazła lepszego partnera niż m . Zatem utworzenie pary (m, k) nie narusza stabilności i operacja oświadcza się zachowuje stabilność całego układu. Układ początkowy (układ pusty) był stabilny, tak więc wszystkie kolejne układy, łącznie z końcowym (pełnym), są stabilne.

Opisany tutaj algorytm zakończy się w skończonej liczbie kroków i utworzy stabilny układ par. Czy jest to jednak jedyny taki układ? Zauważmy, że w algorytmie wybieramy pewnego wolnego mężczyznę (pozwala to ominąć szczegóły techniczne w opisie algorytmu). Czy końcowy układ zależy od kolejnych wyborów? Czy otrzymany układ preferuje którąś ze stron? Odpowiedź w numerze.

Rozwiązanie zadania F 163.

1. Rozwiązanie c) odrzucamy, bo wymiar prawej strony równania nie jest wymiarem siły.

2. Wartość poszukiwanego naprężenia nie powinna zmięknąć się przy równoczesnej zmianie F_1 na $-F_2$ i H na $L-H$, co odpowiada obrotowi o 180° wokół osi pionowej przechodzącej przez oś symetrii pręta. Warunku tego nie spełnia rozwiązanie a).

3. Na końcach pręta naprężenia muszą być równe odpowiednio F_2 i F_3 , co eliminuje rozwiązanie b).

4. Gdy pręt poddany jest działaniu równych co do wartości sił, naprężenie w każdym punkcie pręta jest stałe i równe sile zewnętrznej. Rozwiązanie c) nie spełnia tego warunku.

Ponieważ jedno z rozwiązań jest prawidłowe, musi nim być rozwiązanie c).

Czy przez telefon można grać w karty?

na podstawie artykułu „Poker bez kart” z książki „*The Mathematical Gardner*” (A. Shamir, R. Rivest, L. Adelman).

O telefonicznej czy korespondencyjnej grze w szachy słyszał każdy. Ale jak grać w ten sposób w brydża lub w pokera? Problemem jest oczywiście rozdawanie kart. Przypuśćmy, że grają dwie osoby i mają rozdać po pięć kart. Rozdać to znaczy:

Każdy ma wiedzieć, jakie pięć kart dostał.

Karty otrzymane przez graczy są różne.

Żaden z graczy nie ma dodatkowej informacji o kartach partnera, ale po grze może sprawdzić, czy partner nie oszukiwał, czy grał swoimi kartami.

Każdy rozkład kart jest jednakowo prawdopodobny.

Wszystko to należy wykonać porozumiewając się wyłącznie przez telefon i bez pomocy osób trzecich. Oto sposób umożliwiający w praktyce rozdawanie kart (liczb naturalnych $1, \dots, 52$) przez telefon. Gracze wybierają najpierw dwie rodziny funkcji o argumentach i wartościach naturalnych: $\mathcal{K} = \{K_\alpha: \alpha \in \Omega\}$ — funkcje kodujące i $\mathcal{D} = \{D_\alpha: \alpha \in \Omega\}$ — funkcje dekodujące (zbiór Ω nazywamy zbiorem kodów — powinien on mieć dużo elementów). Rodziny \mathcal{K} i \mathcal{D} muszą mieć następujące własności:

Dziedzina każdej funkcji K_α zawiera zbiór $\{1, \dots, 52\}$.

Dla dowolnego kodu α funkcja D_α jest odwrotna do funkcji K_α (rozszyfrowuje ona sygnał zakodowany za pomocą funkcji K_α), tzn. $D_\alpha(K_\alpha(n)) = n$ dla liczb naturalnych z dziedziny funkcji K_α .

Dla dowolnych kodów α i β funkcje K_α i K_β są przemienne, tzn. $K_\alpha(K_\beta(n)) = K_\beta(K_\alpha(n))$.

Różne funkcje kodujące mają rozłączne zbiory wartości.

Znajomość liczb naturalnych n i $K_\alpha(n)$ nie daje *praktycznie* możliwości znalezienia kodu α .

Rozdawanie kart jest już proste. Gracze wybierają (w tajemnicy przed sobą) kody, np. A — kod α , B — kod β . Gracz A koduje liczby $1, \dots, 52$ i przesyła je (w dowolnej kolejności) graczowi B . Ten wybiera w pierw pięć kart dla A : $K_\alpha(a_1), \dots, K_\alpha(a_5)$ i odsyła mu je — A musi je rozszyfrować funkcją D_α . Następnie wybiera pięć kart dla siebie: $K_\alpha(b_1), \dots, K_\alpha(b_5)$, szyfruje je funkcją K_β i wysyła do A . Gracz A rozszyfrowuje je funkcją D_α i odsyła do gracza B (tzn. przesyła $D_\alpha(K_\beta(K_\alpha(b_i))) = D_\alpha K_\beta(K_\alpha(b_i)) = K_\beta(b_i)$). Gracz B musi jeszcze rozszyfrować je funkcją D_β i ... karty zostały rozdane.

Po grze partnerzy ujawniają swoje kody. Z drugiej i czwartej własności rodzin \mathcal{K} i \mathcal{D} wynika, że jeśli $K_{\alpha_1}(m_1) = K_{\alpha_2}(m_2)$, to $\alpha_1 = \alpha_2$ i $m_1 = m_2$. Tak więc gracze nie mogą oszukiwać i podawać innego układu kart i innego kodu.

Powyższy opis umożliwia w *praktyce* rozdawanie kart. *Teoretycznie* bowiem jest to niemożliwe.

J. R.

Oto przykład rodzin \mathcal{K} i \mathcal{D} : p ustalona duża liczba pierwsza,
 $\Omega = \{2, \dots, p-1\}$,
 $K_\alpha(n) \equiv n^\alpha \pmod{p}$,
 $D_\alpha(n) \equiv n^\beta \pmod{p}$,
gdzie $1 \leq n \leq p$ i $\beta \cdot \alpha \equiv 1 \pmod{p}$
Oczywiście gracze muszą w tym przypadku korzystać z pomocy komputerów, ale rodzina \mathcal{K} ma nadal ostatnią z własności, tzn. *praktycznie* ze znajomości n i $K_\alpha(n)$ nie uda się wyznaczyć α .

Rozdawanie kart, jeśli zamienić telefon na pocztę, można opisać tak. A wkłada karty do jednakowych pudełek (do każdego jedną kartę), zamyka każde pudełko taką samą kłódką i odsyła je do B . B odsyła pięć pudełek (są to karty gracza A) oraz pięć pudełek zamkniętych dodatkowo swoją kłódką. A wyjmuje swoje karty oraz zdejmuje swoje kłódkę z pudełek z kartami B i odsyła te pudełka do B . B wyjmuje swoje karty.



Dni juliańskie

Obliczenie odstępu czasu między dwiema datami jest procedurą bardzo często spotykaną w rozmaitych działach astronomii, choćby przy obliczaniu przewidywanych położań ciał niebieskich. Ogromnie pomaga temu tzw. juliańska rachuba dni, według której każdy dzień, poczynając od umownego, dość odległego momentu w przeszłości, ma swój numer kolejny (jeśli ktoś koniecznie chce wiedzieć, chodzi o południe 1 stycznia 4713 p.n.e.).

Niech funkcja $[x]$ przypisuje liczbie rzeczywistej x jej część całkowitą. Zapiszmy datę (naszej ery) w postaci $DD MM RRRR$, gdzie DD oznacza numer dnia w miesiącu, MM numer miesiąca i $RRRR$ rok. Początkowi danego dnia (czyli północy czasu uniwersalnego rozpoczynającej daną datę) odpowiada w rachubie juliańskiej numer dnia (JD) obliczany według następującego schematu:

Jeżeli $MM = 1$ lub 2 , to $r = RRRR - 1$ oraz $m = MM + 12$.

Jeżeli $MM > 3$, to $r = RRRR$ oraz $m = MM$.

$JD = [365,25r] + [30,6001(m+1)] + DD + \{2 - [r/100] + [r/400]\} + 1720994,5$.

Numer dnia juliańskiego zmieniany jest w południe czasu uniwersalnego, stąd 0,5 na końcu wzoru. Trzeba też tu wiedzieć, że do 4 X 1582 r. obowiązywał tzw. kalendarz juliański (uwaga: zbieżność nazw kalendarza juliańskiego i juliańskiej rachuby dni jest czysto przypadkowa), a kalendarz gregoriański od dnia następnego, któremu jednak przypisano datę 15 X 1582. Tak więc daty od 5 X 1582 do 14 X 1582 formalnie nie istnieją. Formułę na JD stosujemy w pełnej postaci dla dat kalendarza gregoriańskiego, zaś wyrażenie w nawiasie klamrowym pomijamy dla dat kalendarza juliańskiego. Jeżeli chodzi nam o obliczenie odstępu czasu między dwiema datami, to dodawanie 1720994,5 można oczywiście pominąć.

T.K.

mata delta

Konkurs δ

W pięknej książce *The Mathematical Gardner* (tytuł jest żartobliwy: oznacza fonetycznie „matematyczny ogrodnik”; tytuł taki nadano dla umieszczenia nazwiska Martina Gardnera, najbardziej chyba zasłużonego popularyzatora matematyki), która składa się z artykułów o matematyce pisanych przez całą plejadę wybitnych uczonych, Herbert Taylor daje kilka przykładów brył, które są równoważne topologicznie. Nie dowodzi jednak tej równoważności na drodze rozważań teoretycznych, lecz pokazuje jak przez rozciąganie, ściąganie, wyginanie, słowem przez odkształcenia bez sklejanja i rozcyania zrobić z jednej inną. Pierwszy z przykładów Taylora na rysunku.

Konkurs zaś polega na wykazaniu, że i następna narysowana bryła jest równoważna poprzednim. Wykazanie powinno polegać na ciągu rysunków prowadzących od bryły III do bryły I. Najładniejszy taki ciąg wydrukujemy, a wśród autorów dobrych rozwiązań rozlosujemy nagrody. Termin nadsyłania rysunków — do 15 lutego 1985 roku.

Bryła III.



Bryła I.



Bryła II.

Post scriptum: *The Mathematical Gardner* wydano w USA. Jest też wydanie rosyjskie. Nasuwa to na myśl fakt, że najlepsza polska książka popularyzująca matematykę: *Kalejdoskop matematyczny* Hugona Steinhausa, ma wiele powojennych, stale wznawianych wydań w ZSRR i USA, w Polsce zaś była ostatni raz wydana ponad ćwierć wieku temu. A my się dziwimy, że we wszystkich zastosowaniach matematyki jesteśmy analfabetami. Nie ma się co dziwić — analfabeta to taki, co nie czyta.

Anser

Vulpecula

SAGITTA

z teleskopu

Piramidka

Prowadzenie współczesnych obserwacji astronomicznych przy użyciu dużych teleskopów wymaga ogromnej precyzji. Wystarczy stwierdzić, że od instrumentów tych oczekuje się dokładności ustawienia co najmniej 1" (jest to kąt, pod jakim widzimy złotówkę z odległości 3 km) i utrzymania tej dokładności przez wiele godzin. Dla zilustrowania trudności zauważmy, że mechanizm prowadzący teleskop skierowany na gwiazdę przesuwaną się po nieboskłonie w wyniku ruchu dobowego pokonuje 15" na sekundę. Do tego dochodzi ogromna liczba dodatkowych ruchów, które muszą zostać uwzględnione, m.in. zmieniająca się refrakcja nad horyzontem, zmieniający się kształt i długość tubusa na skutek zmiennych naprężeń i różnic temperatur, czasem niedokładne ustawienie osi instrumentu, kołysanie itd. Przy obserwacjach komet czy planetoid dochodzi jeszcze konieczność uwzględnienia ruchu tych ciał względem gwiazd. Ewentualny komputer sterujący poprawkami do ruchu teleskopu wymagałby ogromnej liczby parametrów, zastosowano więc tu urządzenie znacznie prostsze — tzw. piramidkę. Umożliwia ona śledzenie powoli poruszających się ciał niebieskich, niezależnie od ich trajektorii, na tyle efektywnie, że obecnie przy zastosowaniu piramidki rezygnuje się z tzw. montażu paralaktycznych.

Na drodze promienia światła przychodzącego od gwiazdy przez teleskop ustawia się szybkę, która odbija znikomą część światła w kierunku piramidki, czyli ostrosłupa o podstawie kwadratowej otoczonego z czterech stron fotokomórkami — każda z nich połączona jest z silniczkiem sterującym ruchem teleskopu. Słowo „silniczek” zostało tu użyte świadomie. Te wielotonowe giganty są wyważone tak idealnie, że do ich poruszania wystarczą silniczki o mocy 0,5 W!

Otóż, dopóki światło jest rozpraszane na wierzchołku piramidki, wszystkie cztery fotokomórki oświetlane są tak samo i teleskop pracuje zgodnie z określonym programem. Jednak jeśli zacznie się opóźniać lub z innych powodów gwiazda „zejdzie” ze środka pola widzenia — wiązka światła oświetli bardziej jedną ze ścianek piramidki, a więc i fotokomórkę, która włączy odpowiedni silnik w celu sprowadzenia do sytuacji wyjściowej. Silnik będzie pracował dopóty, dopóki wiązka światła nie przestanie oświetlać „jego” ścianki piramidki.

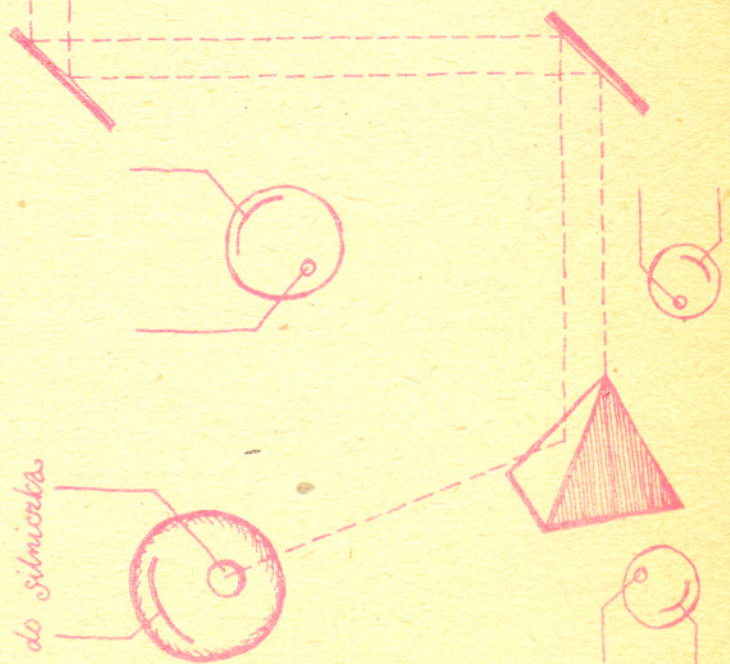
Początkowo stosowano urządzenia tego typu jedynie w celu korygowania ruchu teleskopu poruszanego mechanicznie wokół osi skierowanej na biegun (wspomniany wyżej montaż paralaktyczny). Obecnie jednak coraz częściej stosuje się znacznie prostsze, tańsze i lżejsze montaże zenitalne (teleskop obraca się wokół osi pionowej i poziomej), a resztę załatwiają ciągle pracujące silniczki.

Piramidki wykorzystywane są również do znalezienia odpowiedniej gwiazdy. Wystarczy zadać komputerowi sterującemu teleskopem przybliżone współrzędne gwiazdy, a następnie, gdy znajduje się ona w polu widzenia, naprowadzamy ją przy użyciu piramidki na środek.

Jest to więc nieoceniona pomoc w pracy astronoma, może już niedługo będzie również ostrzyła żyłki, aby ułatwić mu golenie po nieprzespanej nocy obserwacyjnej.

Małą Deltę przygotowali Tomasz CHLEBOWSKI i Marek KORDOS

gigantów niebo
gigantów niebo obok



John Smith kupił cztery towary, takie, że zarówno suma, jak i iloczyn ich cen były równe 7.11. Trzeba obliczyć, ile kosztował każdy z nich.

Zadanie sprowadza się do znalezienia takiej czwórki liczb (a, b, c, d) , że:

- $a+b+c+d = 7,11$,
- $abcd = 7,11$.

W dziedzinie liczb rzeczywistych układ ten ma oczywiście continuum rozwiązań. Naturalne jest jednak przyjąć, że ceny muszą wyrażać się całkowitymi liczbami centów. Jeżeli więc oznaczymy $K = 100a$, $L = 100b$, $M = 100c$, $N = 100d$, to warunki 1. i 2. przyjmą postać:

- $K+L+M+N = 100a+100b+100c+100d = 100(a+b+c+d) = 711$,
- $KLMN = 100a \cdot 100b \cdot 100c \cdot 100d = 100^4 abcd = 711\,000\,000$,
oraz dodatkowo:
- $K, L, M, N \in \mathbb{N}$.

Gdy w ten sposób sprecyzowaliśmy ściśle problem, należy wybrać metodę jego rozwiązania. Najszybszym i wymagającym najmniej pracy sposobem jest wyłączenie się komputerem. Dla maszyny musimy jednak przygotować dokładny przepis (algorytm) szukania rozwiązania. Najprostszy algorytm polega na tworzeniu wszystkich czwórek liczb naturalnych mniejszych od 711 i sprawdzaniu, czy spełniają one warunki zadania. Jednak przeszukiwanie wszystkich kombinacji jest bardzo uciążliwe — nawet szybki komputer robiłby to niezwykle długo. Konieczne jest więc ograniczenie liczby prób. Trzy takie ograniczenia nasuwają się od razu:

- będziemy rozpatrywać tylko rosnące ciągi liczb — unikniemy w ten sposób wyliczania wszystkich permutacji i będziemy mogli

skorzystać z zależności, że jeżeli $a_1+a_2+\dots+a_n = C$ i

$$a_1 \leq a_2 \leq \dots \leq a_n, \text{ to } a_1 \leq \frac{1}{n} \cdot C,$$

- jeżeli iloczyn poszukiwanych n liczb nie jest podzielny przez iloczyn pewnego ich podzbioru, to nie ma potrzeby poszukiwać pozostałych,
- ostatnią liczbę można wyliczyć z pozostałych i warunku na sumę.

Można wprowadzić dalsze ograniczenia na liczbę sprawdzeń, ale trzeba przecież wybrać jakiś kompromis między efektywnością algorytmu a wysiłkiem włożonym w jego opracowanie — nie chodzi o to, by rozwiązać zadanie „na piechotę”, a komputerowi kazać tylko wydrukować wyniki.

Oto algorytm:

- Wybierz kolejną liczbę naturalną $K \geq 1$; jeżeli $K > 711/4$, to koniec.
- Jeżeli K jest dzielnikiem $7,11 \cdot 100^4$, wykonaj 3, w przeciwnym przypadku wykonaj 1.
- Wybierz kolejną liczbę naturalną $L \geq K$; jeżeli $L > (711-K)/3$, to wykonaj 1.
- Jeżeli L jest dzielnikiem $7,11 \cdot 100^4/K$, wykonaj 5, w przeciwnym przypadku wykonaj 3.
- Wybierz kolejną liczbę naturalną $M \geq L$; jeżeli $M > (711-K-L)/2$, to wykonaj 3.
- Jeżeli M jest dzielnikiem $7,11 \cdot 100^4/(K \cdot L)$, wykonaj 7, w przeciwnym przypadku wykonaj 5.
- Jeżeli $7,11 \cdot 100^4/(K \cdot L \cdot M) = 711-K-L-M$, to czwórka $(K, L, M, N = 711-K-L-M)$ jest rozwiązaniem.
- Wykonaj 5.

Algorytm ten zapisany w języku FORTRAN wykonywał się na maszynie CDC CYBER 73 przez 4,5 s i sprawdził tylko 5435 pełnych czwórek.

Ksawery STOJDA

Zadania

Redaguje dr Krzysztof S. NOWIŃSKI

M 383. Budujemy tablicę, zawierającą w n -tym wierszu 2^{n-1} liczb naturalnych umieszczając w pierwszym wierszu dowolną liczbę naturalną $a > 1$, a następnie pod każdą liczbą b występującą w k -tym wierszu umieszczamy $b+1$ oraz b^2 :

Wykazać, że dla każdego n wszystkie liczby w n -tym wierszu są różne.

Rozwiązanie na str. 3

M 384. Liczbę całkowitą m taką, że $|x-m| < 1$ nazwiemy zaokrągleniem liczby rzeczywistej x . (Gdy x nie jest całkowite, mamy dwa zaokrąglenia: $[x]$ i $[x]+1$.) Wykazać, że każdą z n liczb x_1, \dots, x_n można zaokrąglić tak, że suma dowolnego podzbioru zbioru $\{x_1, \dots, x_n\}$ będzie się różnić od sumy wybranych zaokrągleń liczb najwyżej o $\frac{n+1}{4}$.

Rozwiązanie na str. 3

M 385. Wykazać, że $99999 + 100000\sqrt{3}$ nie jest kwadratem liczby postaci $a+b\sqrt{3}$, gdzie a i b są całkowite.

Rozwiązanie na str. 2

Redaguje mgr Tomasz TRATKIEWICZ

F 163. Na poziomej gładkiej powierzchni znajduje się sztywny, jednorodny pręt o długości L poddany działaniu sił F_1 i F_2 (rysunek). Jaka siła (naprężenie) działa w przekroju odległym o H od końca pręta?

Tym razem wraz z treścią podajemy kilka wyników, a wśród nich prawidłowy. Zadaniem Czytelników jest wskazanie go bez uciekania się do rozwiązywania problemu.

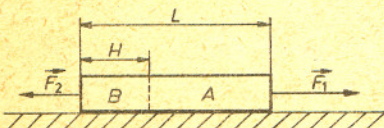
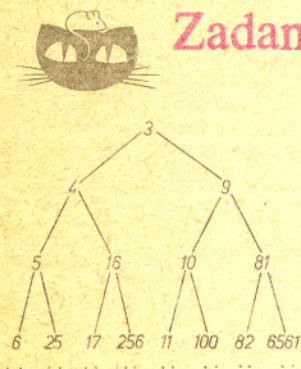
$$\text{a) } N = \left(\frac{H}{L}\right)^2 (F_1 + F_2) - F_2, \quad \text{b) } N = \frac{H}{L} (F_1 + F_2) - F_1,$$

$$\text{c) } N = H(F_1 + F_2) - LF_2, \quad \text{d) } N = \frac{H}{L} (F_1 + F_2) - F_2,$$

$$\text{e) } N = \left(\frac{H}{L}\right)^2 F_1 - \left(1 - \frac{H}{L}\right)^2 F_2$$

(N jest siłą, z jaką element A działa na B).

Rozwiązanie na str. 6



Artykuł profesora Otto w numerze poświęconym anaglifom zawierał dość szczegółowy wykład teorii perspektywy i jej zastosowania do konstrukcji anaglifów, jednak jego praktyczne wykorzystanie nie jest łatwe. Dlatego warto przedstawić problem konstrukcji anaglifów metodą „książki kucharskiej”. Uzasadnieniem podanego niżej przepisu mogą być anaglify prezentowane w numerze 4 i następnych. Rysował je wprawdzie program komputerowy, jednak to, co powiemy, jest ścisłą graficzną analogią wykonywanych rachunków. Wykorzystujemy tu też prosty fakt, że obraz perspektywiczny punktów leżących w płaszczyźnie równoległej do płaszczyzny rysunku powstaje z rzutu prostopadłego tych punktów przez pewną jednokładność. Środkiem tej jednokładności jest rzut prostopadły środka perspektywy (oka) na płaszczyznę rysunku, a stosunkiem podobieństwa jest stosunek odległości oka od płaszczyzny rysunku do odległości oka od płaszczyzny, w której dane punkty leżą (patrz rys. 1).

Wynika stąd, że bez większych trudności możemy rysować anaglify nawet dość skomplikowanych obiektów, takich np. jak pokazana na str. 13 numeru 4 hiperboloida, jeżeli tylko ich charakterystyczne punkty (tzn. punkty, z których możemy samą linijką odtworzyć cały obraz) leżą w niezbyt licznych zbiorach płaszczyzn równoległych do płaszczyzny rysunku. Ogranicza to wprawdzie swobodę umieszczenia obiektu w przestrzeni, nie zawsze jednak mamy pod ręką komputer. (Możliwości obliczeniowe komputera pozwalają ominąć powyższe ograniczenie, dzięki czemu narysowanie np. sieci przestrzennej grafitu w „położeniu dowolnym” (*Delta* 4/1984, str. 2) nie było trudniejsze niż sporządzenie rysunku pokazanego obok.)

Spróbujmy więc narysować anaglif czternastościanu pokazanego na str. 7 w numerze 4 (drugi od lewej u dołu). Jest to graniastosłup trójkątny, na którego ścianach bocznych dokleiono ostrosłupy czworokątne, przy czym wszystkie czternaście ścian to trójkąty foremne. Łatwo widzieć, że wszystkie dziewięć jego wierzchołków leży w trzech równoległych płaszczyznach, dając w każdej z nich trójkąt równoboczny.

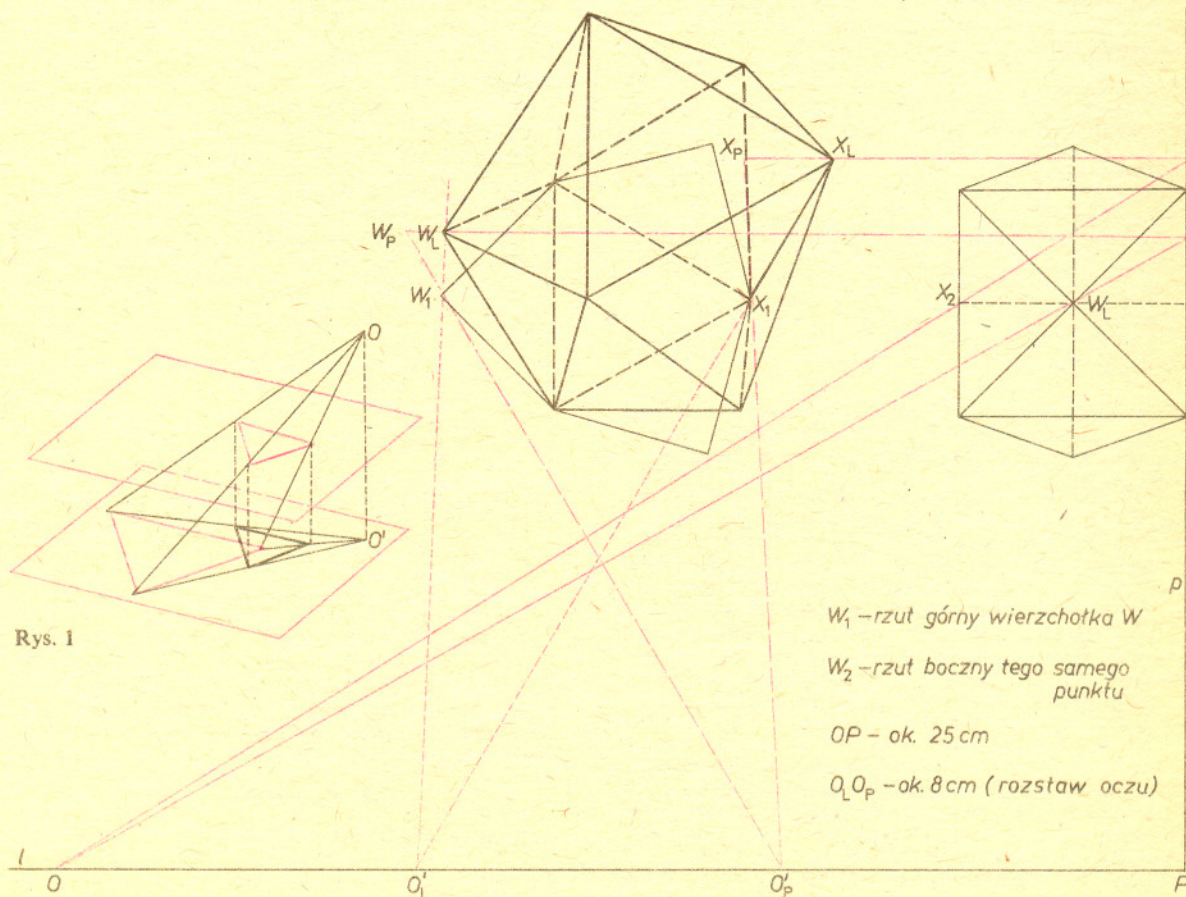
Przy rysowaniu zachowamy przyjęte w anaglifach *Delty* położenie oka względem przedmiotu (i rysunku): zauważyliście pewnie, że anaglify wyglądają najbardziej „naturalnie”, gdy patrzymy z wysokości około 25 cm na rysunek leżący poziomo 12–15 cm przed nami.

Umieśmy teraz rzut pionowy i boczny przedmiotu tak, jak na rysunku 2 — około 12 cm nad „prostą podstawy” l , na której zaznaczyliśmy położenie rzutu oka prawego O'_p , lewego O'_l oraz punkt pomocniczy O oddalony o 25 cm od prostej p będącej podstawą rzutu bocznego. Ściana, na której nasz wieńciany stoi, odwzorowuje się tożsamościowo, a aby otrzymać rzut W'_l dowolnego innego wierzchołka W , prowadzimy trzy zaznaczone na rys. 2 proste pomocnicze. (Pozostałe punkty leżące w tej samej płaszczyźnie co W możemy odwzorować korzystając z tego, że ich obrazy powstają przez znaną nam jednokładność ze środkiem O'_l .)

W ten sposób łatwo uzyskujemy anaglif „lewy”, anaglif „prawy” najłatwiej jest uzyskać pamiętając o tym, że obrazy jednego punktu leżą w tej samej odległości od prostej l — tak uzyskaliśmy np. obraz W_p .

Gdy dysponujemy kalką techniczną, parę anaglifów obiektu o około dwudziestu punktach charakterystycznych można uzyskać w godzinę.

dr Krzysztof S. NOWIŃSKI



Klasyczna transformacja Fouriera odgrywa zasadniczą rolę we współczesnej matematyce, w szczególności w analizie harmonicznej. Jak pisze prof. Maurin, zarówno w rachunkach, jak i rozważaniach teoretycznych oddaje ona nieocenione usługi, a pewien wybitny matematyk stwierdził: „ludzie wymyślili wiele transformacji, ale transformację Fouriera stworzył Bóg” (matematyk ten jest ateistą). Młodszą siostrą klasycznej transformacji Fouriera jest dyskretna transformacja Fouriera, która odgrywa zasadniczą rolę we współczesnej informatyce.

Aby w możliwie prosty sposób zdefiniować dyskretną transformację Fouriera, rozważmy dowolny wektor $a = [a_0, a_1, \dots, a_{n-1}]$, którego współrzędne są liczbami zespolonymi, przez P oznaczmy wielomian

$$P(z) = a_0 + a_1 z + \dots + a_{n-1} z^{n-1}.$$

Niech ω będzie pierwotnym pierwiastkiem n -tego stopnia z jedności, wówczas ciąg $\omega^0, \omega^1, \dots, \omega^{n-1}$ określa wszystkie pierwiastki n -tego stopnia z jedności. Otóż dyskretna transformacja Fouriera jest to funkcja F , która danemu wektorowi a przyporządkowuje wektor $b = [b_0, b_1, \dots, b_{n-1}]$ o współrzędnych określonych przez wartości wielomianu P w kolejnych punktach $\omega^0, \omega^1, \dots, \omega^{n-1}$, tzn.

$$b = F(a) = [P(\omega^0), P(\omega^1), \dots, P(\omega^{n-1})].$$

Wiadomo, że iloczyn liczb zespolonych $w_1 = r_1(\cos \alpha_1 + i \sin \alpha_1)$ i $w_2 = r_2(\cos \alpha_2 + i \sin \alpha_2)$ danych w postaci trygonometrycznej jest liczbą $w_1 \cdot w_2 = r_1 \cdot r_2(\cos(\alpha_1 + \alpha_2) + i \sin(\alpha_1 + \alpha_2))$. Tak więc pierwiastkiem n -tego stopnia z 1 będzie każda taka liczba $z = \cos \varphi + i \sin \varphi$, że $n \cdot \varphi = 2k\pi$, gdzie k jest liczbą całkowitą. Liczby $\omega_k = \cos \frac{2k\pi}{n} + i \sin \frac{2k\pi}{n}$; $k = 0, \dots, n-1$ są więc wszystkimi pierwiastkami n -tego stopnia z 1. Zauważmy, że $(\omega_k)^k = \omega_k - 1$ taki pierwiastek, którego potęgami są wszystkie inne pierwiastki n -tego stopnia z 1 nazywamy pierwiastkiem pierwotnym.

Czyli współrzędne wektora b są postaci $b_j = \sum_{k=0}^{n-1} a_k (\omega^k)^j$.

Bardzo ważną cechą klasycznej transformacji Fouriera jest to, że transformacja odwrotna ma postać bardzo zbliżoną do transformacji zwykłej. Dyskretna transformacja Fouriera ma podobną własność. Mianowicie, łatwo pokazać, że

$$F^{-1}(a) = \frac{1}{n} [P(\omega^{-0}), P(\omega^{-1}), \dots, P(\omega^{-(n-1)})].$$

Wystarczy wykazać, że $F^{-1}(F(a)) = a$. Oznaczmy przez c_k ($0 \leq k < n$) k -tą współzrędną wektora $F^{-1}(F(a))$. Wówczas:

$$n \cdot c_k = \sum_{j=0}^{n-1} b_j \omega^{-jk} = \sum_{j=0}^{n-1} \sum_{i=0}^{n-1} a_i \omega^{ij} \omega^{-jk} = \sum_{i=0}^{n-1} a_i \sum_{j=0}^{n-1} \omega^{j(i-k)}.$$

Ponieważ jednak ω jest pierwiastkiem n -tego stopnia z jedności, zatem dla

$$i \neq k \quad \sum_{j=0}^{n-1} \omega^{j(i-k)} = \frac{\omega^{n(i-k)} - 1}{\omega^{i-k} - 1} = 0, \text{ oraz dla } i = k \quad \sum_{j=0}^{n-1} \omega^{j(i-k)} = n.$$

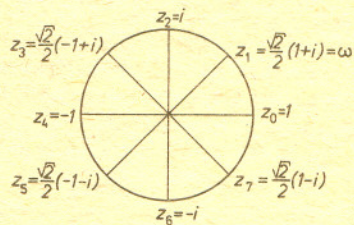
A więc rzeczywiście $F^{-1}(F(a)) = a$.

Po co informatykom dyskretna transformacja Fouriera? O tym, drodzy Czytelnicy, dowiecie się w dalszej części niniejszego artykułu. Na razie zajmijmy się szybkim sposobem obliczania takich transformacji.

Algorytm szybkiego obliczania dyskretnych transformacji Fouriera w literaturze fachowej nosi skrótową nazwę FFT, od angielskiego Fast Fourier Transform. Załóżmy najpierw, że n jest potęgą liczby 2 (takie założenie w informatyce nie zmniejsza ogólności zastosowań). Niech zatem $n = 2^m$ dla pewnego całkowitego $m > 0$. Zadanie nasze polega na wyznaczeniu wartości wielomianu P w n punktach $\omega^0, \omega^1, \dots, \omega^{n-1}$ (dla transformacji odwrotnej są to te same punkty, ale w trochę innej kolejności, co tylko w nieznacznym stopniu modyfikuje sposób postępowania).

Z twierdzenia Bezout wiemy, że dla dowolnego zespolonego z' wartość $P(z')$ równa się reszcie z dzielenia P przez dwumian $(z - z')$. No dobrze, ale zastanówmy się teraz, jak można obliczyć wartość wielomianu P w dwóch punktach z' i z'' . Oczywiście można to zrobić dla każdego $z = z'$ i $z = z''$ oddzielnie. Można także podzielić najpierw P przez wielomian $(z - z') \cdot (z - z'')$,

otrzymaną resztę stopnia pierwszego $az + b$ podzielić przez $(z - z')$ otrzymując $P(z')$, a następnie przez $(z - z'')$ otrzymując $P(z'')$. Takie dzielenie w ogólnym przypadku jest bardzo pracochłonne, ale dla naszego zbioru punktów można postąpić jeszcze sprytniej. Ponieważ $n = 2^m$, zatem pierwiastki n -tego stopnia z jedności rozłożą się na okręgu jednostkowym symetrycznie (rys. 1).



Rys. 1

Pogrupujmy je parami tak, aby iloczyn $(z - z') \cdot (z - z'')$ był zawsze wielomianem bez składnika liniowego, czyli aby był postaci $z^2 + a$. Okazuje się, że jest to możliwe. Dla $n = 8$ takie pary mogą być następujące:

$$(z - z_0)(z - z_4) = (z - 1)(z + 1) = (z^2 - 1) = (z^2 - z_0),$$

$$(z - z_2)(z - z_6) = (z - i)(z + i) = (z^2 + 1) = (z^2 - z_4),$$

$$(z - z_1)(z - z_5) = \left(z - \frac{\sqrt{2}}{2}(1+i)\right) \left(z + \frac{\sqrt{2}}{2}(1+i)\right) = (z^2 - i) = (z^2 - z_2),$$

$$(z - z_3)(z - z_7) = \left(z - \frac{\sqrt{2}}{2}(-1+i)\right) \left(z + \frac{\sqrt{2}}{2}(-1+i)\right) = (z^2 + i) = (z^2 - z_6).$$

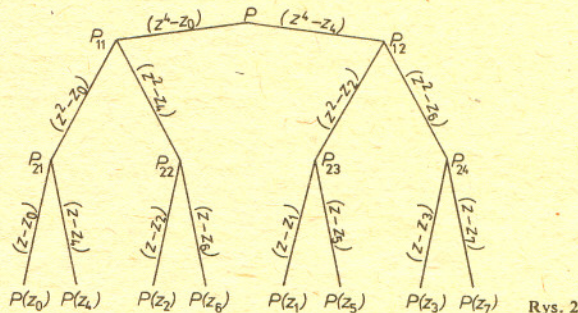
Tak więc, jeżeli podzielimy wielomian P przez wielomiany $(z^2 - 1), (z^2 + 1), (z^2 - i), (z^2 + i)$, a następnie otrzymane reszty przez odpowiednie dwumiany $(z - z_k)$, to w ten sposób możemy obliczyć łatwo wszystkie wartości $P(z_k)$, $k = 0, 1, \dots, 7$.

Zauważmy jednak dalej, że dzielenie przez $(z^2 - z_0), (z^2 - z_4), (z^2 - z_2)$ i $(z^2 - z_6)$ można wykonać rozumując podobnie. Mianowicie pogrupujmy parami te wielomiany tak, aby iloczyn $(z^2 - z') \cdot (z^2 - z'')$ był dwumianem postaci $(z^2 + a)$, co w dalszym ciągu jest możliwe:

$$(z^2 - z_0)(z^2 - z_4) = (z^2 - 1)(z^2 + 1) = (z^4 - 1) = (z^4 - z_0),$$

$$(z^2 - z_2)(z^2 - z_6) = (z^2 - i)(z^2 + i) = (z^4 + 1) = (z^4 - z_4).$$

I znowu dzieląc P przez $(z^4 - z_0)$ i $(z^4 - z_4)$ otrzymamy dwie reszty stopnia trzeciego, które dzieląc przez dwumiany $(z^2 - z_k)$, $k = 0, 4, 2, 6$ dadzą reszty stopnia pierwszego, a z tych reszt można już obliczyć $P(z_k)$ dla $k = 0, 1, \dots, 7$, w sposób uprzednio opisany. Całość tego postępowania wygodnie jest przedstawić w postaci drzewa (rys. 2).



Rys. 2

To jeszcze nie koniec „chwytów” technicznych prowadzących do algorytmu FFT. Otóż dzielenie wielomianu P przez dwumiany $(z^2 - z_0)$ i $(z^2 - z_4)$ jest bardzo proste. Wystarczy zauważyć, że reszta z dzielenia wielomianu

$$P(z) = a_0 + a_1 z + \dots + a_{2k-1} z^{2k-1}$$

przez dwumian $(z^2 - c)$ jest postaci

$$(*) r(z) = (a_0 + ca_k) + (a_1 + ca_{k+1})z + \dots + (a_{k-1} + ca_{2k-1})z^{k-1}.$$

Faktycznie, stopień wielomianu r jest mniejszy niż k oraz

$$P(z) = (a_k + a_{k+1}z + \dots + a_{2k-1}z^{k-1})(z^k - c) + r(z).$$

Mamy zatem resztę z dzielenia wielomianu P przez $(z^k - z_0)$ postaci

$$(a_0 + a_4) + (a_1 + a_5)z + (a_2 + a_6)z^2 + (a_3 + a_7)z^3$$

oraz resztę z dzielenia wielomianu P przez $(z^k - z_4)$ postaci

$$(a_0 - a_4) + (a_1 - a_5)z + (a_2 - a_6)z^2 + (a_3 - a_7)z^3.$$

Dzielenie tak otrzymanych reszt przez dwumiany $(z^2 - z_k)$, $k = 0, 4, 2, 6$, wykonujemy podobnie, tzn. stosując wzór (*). Ogólny schemat postępowania w przypadku $n = 8$ przedstawiony jest poniżej, natomiast przykład wykonania takich obliczeń dla konkretnego wielomianu w długiej tabelce.

3	1	2	1	2	1	1	1
5	2	3	2	1	0	1	0
8	4	2	0	1+i	0	1-i	0
12	4	2	2	1+i	1+i	1-i	1-i

$$F(3, 1, 2, 1, 2, 1, 1, 1) = [12, 1+i, 2, 1-i, 4, 1+i, 2, 1-i]$$

współczynniki wielomianu P							
a_0	a_1	a_2	a_3	a_4	a_5	a_6	a_7
współczynniki wielomianu P_{11}				współczynniki wielomianu P_{12}			
$b_0 = a_0 + a_4$	$b_1 = a_1 + a_5$	$b_2 = a_2 + a_6$	$b_3 = a_3 + a_7$	$b_4 = a_0 - a_4$	$b_5 = a_1 - a_5$	$b_6 = a_2 - a_6$	$b_7 = a_3 - a_7$
Współczynniki wielomianu P_{21}		Współczynniki wielomianu P_{22}		Współczynniki wielomianu P_{23}		Współczynniki wielomianu P_{24}	
$c_0 = b_0 + b_2$	$c_1 = b_1 + b_3$	$c_2 = b_0 - b_2$	$c_3 = b_1 - b_3$	$c_4 = b_4 + ib_6$	$c_5 = b_5 + ib_7$	$c_6 = b_4 - ib_6$	$c_7 = b_5 - ib_7$
$P(z_0)$	$P(z_4)$	$P(z_2)$	$P(z_6)$	$P(z_1)$	$P(z_5)$	$P(z_3)$	$P(z_7)$
$d_0 = c_0 + c_1$	$d_1 = c_0 - c_1$	$d_2 = c_2 + ic_3$	$d_3 = c_2 - ic_3$	$d_4 = c_4 + \frac{\sqrt{2}}{2}(1+i)c_5$	$d_5 = c_4 - \frac{\sqrt{2}}{2}(1+i)c_5$	$d_6 = c_6 + \frac{\sqrt{2}}{2}(-1+i)c_7$	$d_7 = c_6 - \frac{\sqrt{2}}{2}(-1+i)c_7$

Ile operacji arytmetycznych na liczbach zespolonych trzeba wykonać w celu obliczenia $F(a)$ stosując algorytm FFT? Zauważmy, że dla $n = 2^m$ musimy wyznaczyć m kolejnych wierszy reprezentujących współczynniki odpowiednich reszt. Łatwo sprawdzić, że wyznaczenie takiego wiersza z poprzedniego wymaga n dodawań i n mnożeń. Zatem w sumie trzeba wykonać $2 \cdot 2^m \cdot m = 2 \cdot n \cdot \log(n)$ operacji arytmetycznych na liczbach zespolonych. Jest to kluczowa własność tego algorytmu. Mianowicie wyznaczenie $F(a)$ za pomocą algorytmu FFT wymaga wykonania tylko rzędu $n \log(n)$ operacji arytmetycznych, podczas gdy algorytmy tradycyjne wymagające dyskretną transformację Fouriera wymagają wykonania rzędu n^2 operacji arytmetycznych. Wróćmy teraz do pytania, po co w ogóle chcemy wyznaczać dyskretną transformację Fouriera. Otóż pomocne są one niezwykle w obliczaniu spłotu wektorów. Cóż to jest spłot wektorów? Znać zapewne tę operację, jakkolwiek być może sam termin jest Wam nieznanym. Niech P i Q będą dwoma wielomianami stopnia $n-1$:

$$P(z) = a_0 + a_1z + \dots + a_{n-1}z^{n-1}, \quad Q(z) = b_0 + b_1z + \dots + b_{n-1}z^{n-1}.$$

Mnożąc te wielomiany otrzymamy wielomian stopnia $2n-2$

$$P(z)Q(z) = c_0 + c_1z + \dots + c_{2n-2}z^{2n-2}, \quad \text{gdzie}$$

$$c_0 = a_0b_0, \quad c_1 = a_0b_1 + a_1b_0, \quad \dots, \quad c_{2n-2} = a_{n-1}b_{n-1}.$$

Otóż wektor $c = [c_0, c_1, \dots, c_{2n-2}]$ nazywamy spłotem wektorów $a = [a_0, a_1, \dots, a_{n-1}]$, $b = [b_0, b_1, \dots, b_{n-1}]$, co oznacza się najczęściej następująco: $c = a * b$.

Gdybyśmy chcieli wyznaczyć spłot $a * b$ wprost z definicji, to musielibyśmy wykonać około $2n^2$ operacji arytmetycznych, co każdy z Was może łatwo sprawdzić. W celu wyznaczenia spłotu można posłużyć się dyskretną transformacją Fouriera. Rozszerzmy mianowicie wektory a i b dołączając do nich n współrzędnych o wartościach 0, tj. $a = [a_0, a_1, \dots, a_{n-1}, 0, \dots, 0]$, $b = [b_0, b_1, \dots, b_{n-1}, 0, \dots, 0]$. Spłot $a * b$ może mieć też $2n$ współrzędnych, jeżeli ostatnią ustalimy jako 0. Zachodzi wówczas bardzo ważne twierdzenie o splocie:

$$F(a * b) = F(a)F(b),$$

gdzie iloczyn po prawej stronie nie jest iloczynem skalarnym

wektorów $F(a)$ i $F(b)$, ale iloczynem „po współrzędnych” (tzn. wynik nie jest skalarem, ale wektorem o współrzędnych będących iloczynami odpowiednich współrzędnych wektorów $F(a)$ i $F(b)$). Aby wyznaczyć spłot $a * b$, obliczamy najpierw $F(a)$ i $F(b)$, następnie mnożymy te wektory „po współrzędnych” i wreszcie wyznaczamy transformację odwrotną $F^{-1}(F(a)F(b))$. Pierwsze dwie i ostatnia operacja mają koszt rzędu $n \log(n)$, a mnożenie „po współrzędnych” ma koszt rzędu n operacji arytmetycznych.

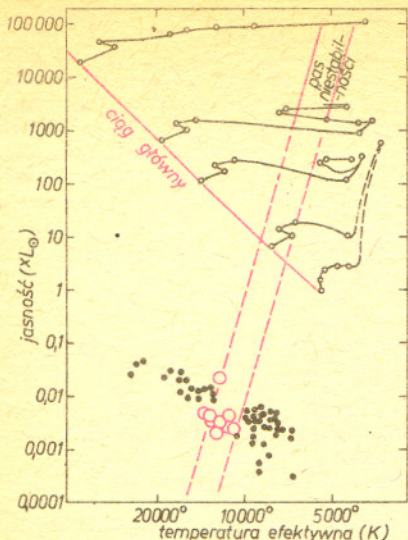
Na zakończenie naszego spotkania z szybką transformacją Fouriera podam jeszcze kilka ważnych faktów dotyczących samej transformacji, jak i jej zastosowań.

Niektórzy Czytelnicy mogą mieć wątpliwości, czy istnieje łatwy sposób znajdowania tej cudownej permutacji początkowej ciągu z_0, z_1, \dots, z_{n-1} (patrz dolny wiersz w rys. 2). Jest to permutacja rzeczywiście cudowna, albowiem na każdym piętrze drzewa mamy do czynienia tylko z dwumianami postaci $z^k - c$, gdzie c jest pewnym pierwiastkiem n -tego stopnia z jedności. Otóż istnieje łatwy sposób wyznaczenia tej permutacji. Mianowicie, niech $n = 2^m$ i rozważmy dowolną liczbę k , $0 \leq k < n$. Taką liczbę można przedstawić w układzie dwójkowym jako $b_0b_1 \dots b_{m-1}$ ($b_i = 0$ lub 1). Niech teraz $rev(k)$ będzie liczbą z tego samego przedziału, o reprezentacji dwójkowej

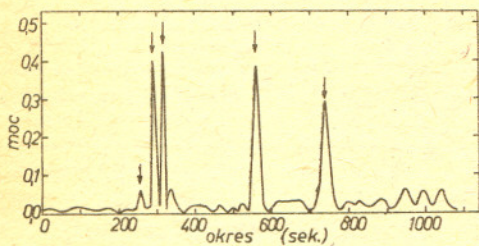
odwrotnej, tzn. $b_{m-1} \dots b_1b_0$. Permutacja przyporządkowująca liczbie k liczbę $rev(k)$ jest szukaną permutacją. Dla $n = 8$ mamy permutację (000), (100), (010), (110), (001), (101), (011), (111), a więc 0, 4, 2, 6, 1, 5, 3, 7 dziesiętnie. Pozostawiamy Czytelnikowi sprawdzenie, że permutacja rev ma żądane własności.

Twierdzenie o splocie można zinterpretować w następujący sposób. Mamy dane dwa wielomiany P i Q . Wyznaczamy ich wartości w $2n$ punktach $\omega^0, \omega^1, \dots, \omega^{2n-1}$ (tu oczywiście ω jest pierwiastkiem stopnia $2n$), co odpowiada wyznaczeniu $F(a)$ i $F(b)$. Następnie mnożymy parami ich wartości w tych punktach $(F(a) \cdot F(b))$. Ostatecznie mając takie $2n$ wartości szukamy wielomianu stopnia co najwyżej $2n-1$, który w tych punktach przyjmuje wyznaczone wartości $F^{-1}(F(a) \cdot F(b))$. Ponieważ istnieje dokładnie tylko jeden wielomian o tej własności, który w punktach $\omega^0, \omega^1, \dots, \omega^{2n-1}$ przyjmuje wartości $P(\omega^i)Q(\omega^i)$ dla $i = 0, \dots, 2n-1$, zatem musi to być iloczyn wielomianów $P(z)Q(z)$. A więc użyta przez nas do pomnożenia dwóch wielomianów liczba działań jest rzędu $n \cdot \log n$, podczas gdy przy „normalnym” mnożeniu jest ona rzędu n^2 .

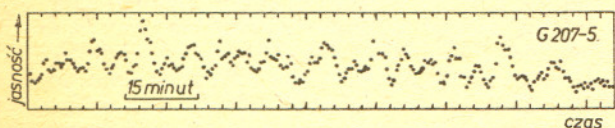
Kilka słów o zastosowaniach szybkiej transformacji Fouriera. Autorzy podstawowego podręcznika ze złożoności obliczeniowej A. Aho, J. Hopcroft, J. Ullman (*Projektowanie i analiza algorytmów komputerowych*, PWN, 1983) piszą, że transformacja ta jest w informatyce wszechobecna. Jest to niewątpliwie prawda. Pokazaliśmy, jak ją stosować do obliczania spłotów wektorów i iloczynu wielomianów. Większość operacji na wielomianach wykonuje się za pomocą tej transformacji. Stosuje się ją także dla arytmetyki wieloprecyzyjnej (gdzy liczby trzeba reprezentować w komputerze za pomocą bardzo długiego ciągu bitów). Wiele zadań numerycznych rozwiązuje się za pomocą tej transformacji, na przykład niektóre zadania prowadzące od układów równań różniczkowych cząstkowych do układów równań liniowych. Czy Fourier mógł przewidzieć, że jego transformacja znajdzie zastosowanie w tomografii komputerowej? Ale nasi Czytelnicy po przeczytaniu niniejszego artykułu będą już wiedzieć, że tomografia komputerowa (nagrodzona zresztą Noblem w dziedzinie medycyny), która pozwala w niesłychanie precyzyjny a jednocześnie prawie nieszkodliwy sposób dokonać wewnętrznej obserwacji pacjenta, też korzysta z algorytmu FFT.



Rys. 1. Diagram Hertzsprunga-Russella. Pas niestabilności przecina drogę ewolucyjną gwiazd (zaznaczone krętymi liniami). Stygnące białe karły przesuwać się w kierunku prawego dolnego rogu rysunku przecinając przedłużenie pasa niestabilności. Te, które pulsują, zaznaczone są kolorem.



Rys. 2. Wykres zmian jasności gwiazdy G 207-9.



Rys. 3. Transformacja Fouriera zmian jasności gwiazdy G 207-9.

Dzisiejsze „Patrz w niebo” chcemy poświęcić jeszcze jednej ilustracji zastosowania szybkiej transformacji Fouriera (FFT) opisaney w tym numerze.

Otóż każdemu wiadomo, że wiele gwiazd zmienia swoją jasność. Znaczna ich część znajduje się w tzw. pasie niestabilności na diagramie Hertzsprunga-Russella. Są to regularnie zmienne cefeidy. Zimą 1964 roku Arlo Landolt z Uniwersytetu w Luizjanie (USA) odkrył bardzo słabą zmienną gwiazdę, której nie udało się wówczas sklasyfikować. Wkrótce okazało się, że gwiazda ta będąc białym karłem leży na przedłużeniu wspomnianego pasa niestabilności. W ciągu następných dwudziestu lat odkryto kilkunastu kolejnych członków nowo powstałej grupy taksonomicznej. Mówi się o nich, że są to obiekty typu ZZ Ceti. Wszystkie znajdują się w wąskim przedziale jasności całkowitych i temperatur powierzchniowych. Ponadto przedziału tego nie okupują żadne inne gwiazdy o atmosferach wodorowych, co może świadczyć o tym, że każdy „wodorowy” biały karzeł musi w trakcie stygnięcia przejść przez fazę ZZ Ceti.

Gwiazdy tego typu są zmienne, jednak krzywe ich zmienności wcale nie wydają się być regularne. Jeśli popatrzeć na rysunek obok, to oczywiście widać wyraźnie maksima i minima jasności, ale już okres ich nie da się łatwo wyznaczyć (spróbujcie!). Przychodzi tu z pomocą FFT. Wykonując transformację Fouriera krzywej zmian jasności otrzymujemy diagram, na którego osi poziomej może być np. okres zmian jasności, a na osi pionowej amplituda tych zmian. FFT wykonana dla obserwacji gwiazdy G 207-9 przedstawionej powyżej daje, podobnie jak dla innych gwiazd typu ZZ Ceti, przedziwny rezultat: okazuje się, że gwiazda zmienia swą jasność z kilkoma, a czasem kilkunastoma okresami jednocześnie. Okresy te i ich wzajemne położenia są kopalnią informacji na temat struktury wewnętrznej gwiazd, ich mas, składu chemicznego, a nawet szybkości obrotu i szybkości stygnięcia tych obiektów.

dr Tomasz CHLEBOWSKI

Data Wielkanocy

K. F. Gauss podał następującą regułę obliczania daty Wielkanocy. Niech a, b, c oznaczają odpowiednio reszty z dzielenia numeru roku przez 19, 4, 7. Przyjmijmy tabelkę wielkości pomocniczych x i y :

dla lat	1583—1699	$x = 22,$	$y = 2$
	1700—1799	23	3
	1800—1899	23	4
	1900—2099	24	5

Niech teraz reszta z dzielenia $(19a + x):30$ wynosi d , a reszta z dzielenia $(26 + 4c + 6d + y):7$ wynosi e . Wtedy datą gregoriańską Wielkanocy jest

$$(22 + d + e) \text{ marca, gdy } d + e \leq 9,$$

$$(d + e - 9) \text{ kwietnia, gdy } d + e > 9.$$

Jednak jeśli $d = 28$ oraz $a < 10$ i otrzymamy datę 25 lub 26 kwietnia, to Wielkanoc wypada 18 kwietnia.

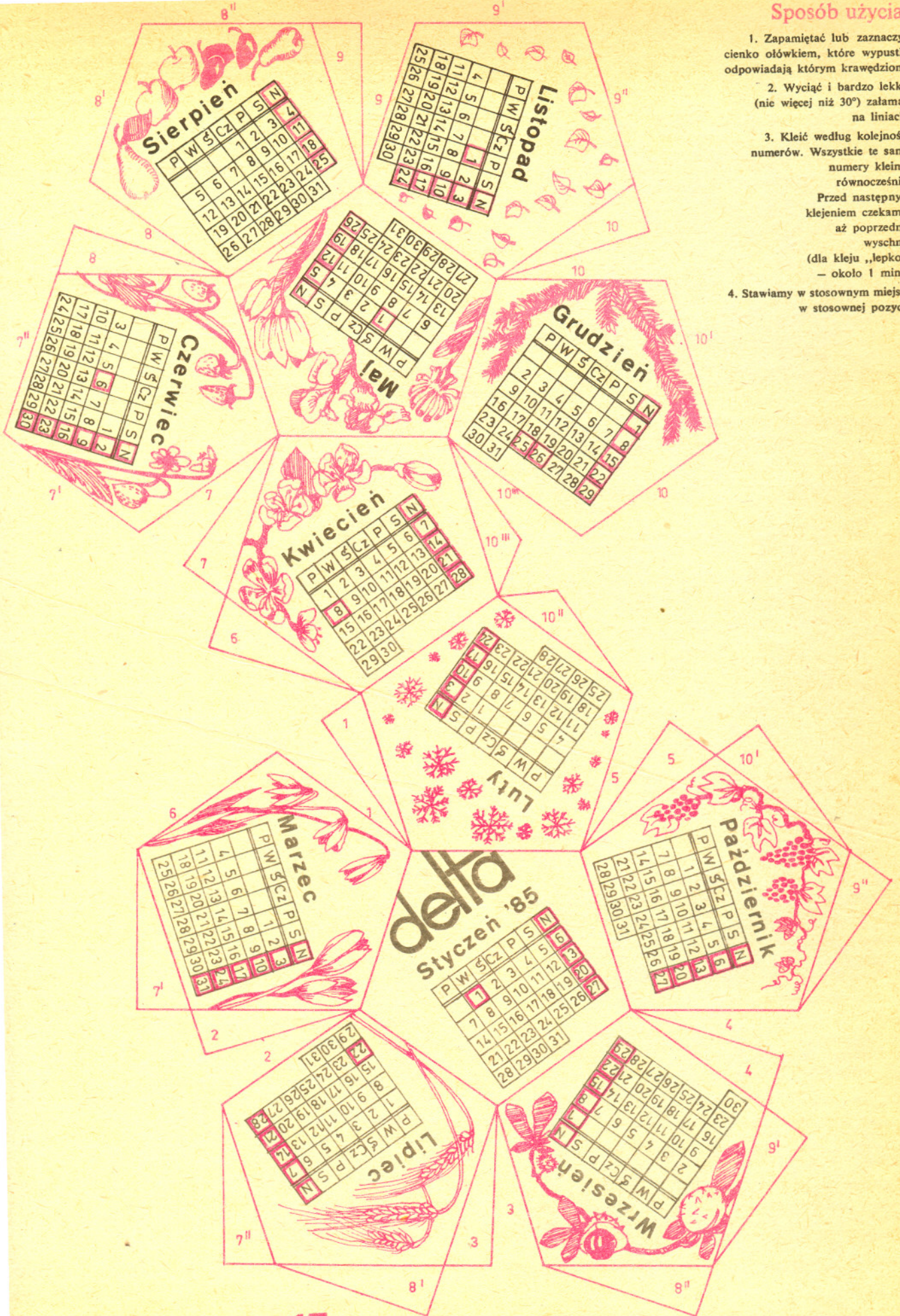
Wiadomo, że Wielkanoc mamy w pierwszą niedzielę po pierwszej wiosennej pełni Księżyca, więc jakaś regularność oczywiście w tym jest, wątpliwe jednak, by ktokolwiek (nawet z astronomów) próbował teraz samodzielnie udowodnić poprawność tej procedury.

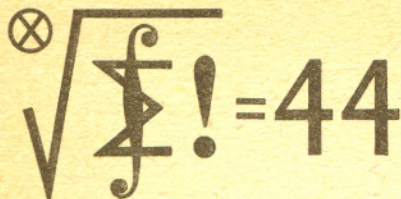
T. K.



Sposób użycia:

1. Zapamiętać lub zaznaczyć cienko ołówkiem, które wypustki odpowiadają którym krawędziom.
2. Wyciąć i bardzo lekko (nie więcej niż 30°) załamać na liniach.
3. Kleić według kolejności numerów. Wszystkie te same numery kleimy równocześnie. Przed następnym klejeniem czekamy, aż poprzednie wyschnie (dla kleju „lepol” – około 1 min.).
4. Stawiamy w stosownym miejscu w stosownej pozycji.





Skrót regulaminu

Każdy może nadsyłać rozwiązania zadań z numeru n w terminie do końca miesiąca $n+2$. Szkice rozwiązań zamieszczamy w nr $n+4$. Można nadsyłać rozwiązania trzech, dwóch lub jednego zadania (każde na oddzielnej kartce), można to robić co miesiąc lub z dowolnymi przerwami. Oceniamy zadania w skali od 0 do 1 z dokładnością do 0,1. Ocenę mnożymy przez

$$4 - 3 \cdot \frac{\text{suma ocen za rozwiązanie danego zadania}}{\text{liczba osób, które nadesłały choć jedno rozwiązanie z numeru}}$$

i tyle punktów otrzymuje nadsyłający. Po zgromadzeniu 44 punktów (w dowolnym czasie) zostaje on członkiem Klubu, a nadwyżka punktów jest zaliczana do ponownego udziału. Trzykrotne członkostwo — to tytuł Weterana.

Szczegółowy regulamin został wydrukowany w nr 1/1984.

Redaguje dr Marcin E. KUCZMA

Zadania nr 100, 101, 102

Termin nadsyłania rozwiązań: 28 II 1985

100. Dwa koła współśrodkowe podzielono na 100 równych sektorów (każde). Na każdym z kół pewne 51 sektorów pomalowano na czerwono, a pozostałe 49 sektorów na niebiesko. Udowodnić, że można tak obrócić jedno koło względem drugiego, żeby co najmniej 52 sektory jednego koła nałożyły się na sektory drugiego o tym samym kolorze (czerwone na czerwone, niebieskie na niebieskie).

101. Czworokąt wypukły $ABCD$ ma tę własność, że koła wpisane w trójkąty ABC, BCD, CDA, DAB mają równe promienie. Dowieść, że $ABCD$ jest prostokątem.

102. Wykazać, że istnieje nieskończenie wiele liczb naturalnych n takich, że

$$\binom{n}{1} - \binom{n}{3} + \binom{n}{5} - \binom{n}{7} + \dots = (\sqrt{2})^{n-1}.$$

Scharakteryzować rozmieszczenie wszystkich takich liczb w ciągu wszystkich liczb naturalnych.

Zadanie 102 przysłał pan Krzysztof Trautman z Warszawy.

Uwaga amatorzy zadań z fizyki !

Za miesiąc rusza liga fizyczna Klubu 44.

Pierwsze zadania z fizyki w numerze 1/1985.

Również w numerze styczniowym zamieścimy

obszerne omówienie ligi matematycznej

oraz dłuższą tabelę ligową

z kilkudziesięcioma nazwiskami.

Rozwiązania zadań z numeru 8/1984

Przypominamy treść zadań:

88. Jeden z kątów trójkąta T ma miarę 120° . Niech T' będzie trójkątem, którego wierzchołkami są punkty przecięcia dwusiecznych kątów trójkąta T z przeciwległymi bokami. Dowieść, że trójkąt T' jest prostokątny.

89. Który z płaskich przekrojów sześcianu ma największe pole?

90. Znaleźć wszystkie rozwiązania równania $x^2 + y^2 + z^2 = x^2 y^2$ w liczbach całkowitych.

88. Oznaczmy wierzchołki trójkątów T i T' odpowiednio przez A, B, C, A', B', C' . Odcinki $\overline{AA'}$, $\overline{BB'}$, $\overline{CC'}$ są dwusiecznymi kątów trójkąta T , $\sphericalangle BCA = 120^\circ$. Niech $\overline{C'B''}$ będzie dwusieczną kąta $AC'C$ ($B'' \in AC$). Na mocy twierdzenia o podziale boku przez dwusieczną zachodzą proporcje: $AB' : B'C = AB : BC$, $AB'' : B''C = AC' : C'C$. Zbudujmy na boku \overline{BC} (na zewnątrz trójkąta T) trójkąt równoboczny BCD . Wówczas $\overline{BD} \parallel \overline{C'C}$, więc $AC' : C'C = AB : BD = AB : BC$. Stąd i z poprzednich proporcji wynika, że punkt B'' pokrywa się z B' , czyli $\overline{C'B'}$ jest dwusieczną kąta $AC'C$. Analogicznie

$\overline{C'A'}$ jest dwusieczną kąta $BC'C$. Wobec tego $\sphericalangle B'C'A' = 90^\circ$.

89. Wybierzmy jedną parę ścian równoległych danego sześcianu — nazwijmy je górną i dolną — i rozpatrzmy przekrój sześcianu płaszczyzną π , tworzącą z płaszczyznami tych ścian kąt φ . Niech H oznacza wielokąt powstały w przekroju sześcianu płaszczyzną π , a H' — rzut tego wielokąta na płaszczyznę ściany dolnej. H' jest częścią wspólną tej ściany oraz pasa płaszczyzny ograniczonego dwiema prostymi równoległymi, z których jedna jest krawędzią przecięcia płaszczyzny π z płaszczyzną ściany dolnej, a druga jest rzutem krawędzi przecięcia π z płaszczyzną ściany górnej (rys. 1). Odległość d między tymi prostymi, długość a krawędzi sześcianu i kąt φ związane są zależnością: $\text{tg } \varphi = a/d$. Przy ustalonym φ — a więc i ustalonym d — obierzmy takie położenie płaszczyzny π , by pole H' było maksymalne. Obliczenie tego pola było treścią zadania ligowego 76 (rozwiązanie w numerze 6/1984); równa się ono

$$S_{\max}(H') = \begin{cases} d(\sqrt{2}a - d/2) & \text{gdy } 0 \leq d \leq a/\sqrt{2} \\ (a^2 + d^2)/2 & \text{gdy } a/\sqrt{2} \leq d \leq a \\ a^2 & \text{gdy } d \geq a \end{cases}$$

Ponieważ $S(H') = S(H)\cos\varphi$, więc przy ustalonym φ pole $S(H)$ jest maksymalne wtedy, gdy $S(H')$ jest maksymalne. Rozpatrując trójkąt prostokątny o przyprostokątnych a i d widzimy, że $\cos\varphi = d(a^2 + d^2)^{-1/2}$.

Stąd

$$S_{\max}(H) = \begin{cases} (\sqrt{2}a - d/2)(a^2 + d^2)^{1/2} & \text{gdy } 0 \leq d \leq a/\sqrt{2} \\ (2d)^{-1}(a^2 + d^2)^{3/2} & \text{gdy } a/\sqrt{2} \leq d \leq a \\ a^2 d^{-1}(a^2 + d^2)^{1/2} & \text{gdy } d \geq a \end{cases}$$

Gdy φ zmienia się od 0° do 90° , to d przebiega wartości od ∞ do 0. Obliczona wartość $S_{\max}(H)$ jest funkcją ciągłą zmiennej d . Badając jej pochodną w przedziałach różniczkowalności stwierdzamy, że jest ona malejąca w $(0, a/\sqrt{2})$, rosnąca w $(a/\sqrt{2}, a)$, malejąca w (a, ∞) . Dla $d = 0$ i $d = a$ przyjmuje wartość maksymalną równą $\sqrt{2}a^2$. Odpowiada to przekrojowi sześcianu płaszczyzną przechodzącą przez równoległe przekątne dwóch przeciwległych ścian (rys. 2).

90. Pokażemy, że jedynym rozwiązaniem jest $x = y = z = 0$. Załóżmy, że (x, y, z) jest rozwiązaniem różnym od $(0, 0, 0)$. Niech k będzie największą nieujemną liczbą całkowitą taką, że każda z liczb x, y, z dzieli się przez 2^k . Wówczas $x = 2^k u, y = 2^k v, z = 2^k w$ i co najmniej jedna z liczb u, v, w jest nieparzysta. Z danego równania wynika, że $u^2 + v^2 + w^2 = 4^k u^2 v^2$. Lewa strona tej równości daje przy dzieleniu przez 4 resztę 1, 2 lub 3 — w zależności od tego, ile jest składników nieparzystych. Zatem $k = 0$, czyli co najmniej jedna z liczb x, y, z jest nieparzysta. Gdyby x i y były obie parzyste, z musiałoby też być parzyste. Wobec tego nieparzysta jest jedna z liczb x, y i z równości $z^2 + 1 = (x^2 - 1)(y^2 - 1)$ wynika, że $z^2 + 1$ dzieli się przez 8 — co jest niemożliwe.

W następnym numerze *Delty* siedemnasta strona będzie już „płaska”. Większość anaglifów w numerach 4–12/1984 wykonał dr Krzysztof S. Nowiński, wspomagany przez komputer PDP 11/45 z plotterem, udostępniony nam przez Centrum Astronomiczne im. Mikołaja Kopernika.

Rozwiązanie konkursu *Małej Delty* z numeru 7/1984

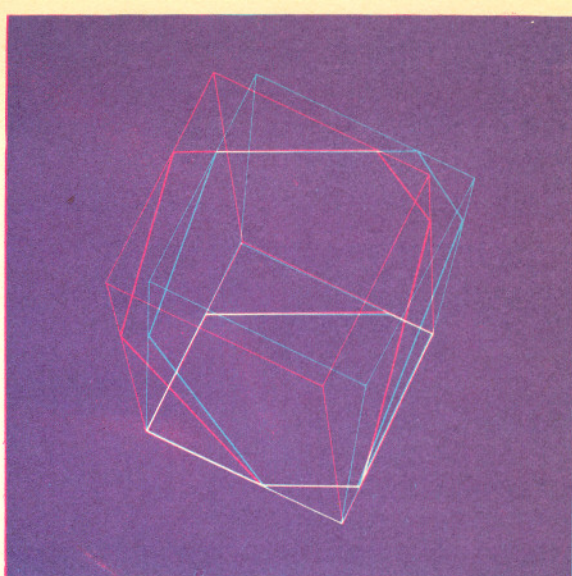
Prawidłowe rozwiązania nadesłało 12 Czytelników.

W wyniku losowania nagrody otrzymali

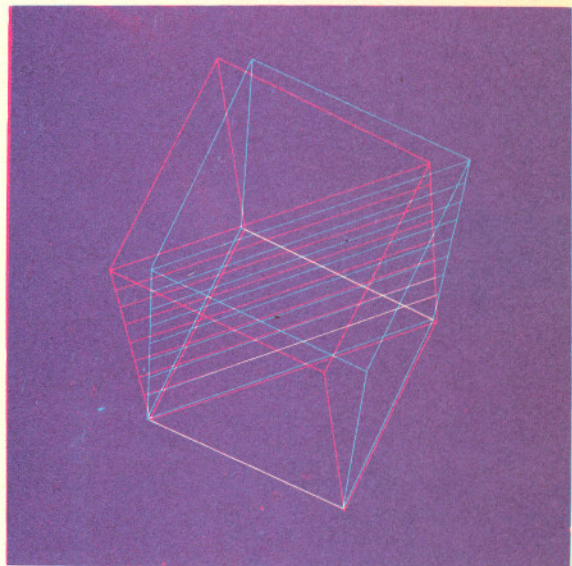
Daniel Zdunek z Otwocka

i Zbigniew Śliwa z Kamiennej Góry.

Nagrody zostały wysłane pocztą.



Rys. 1



Rys. 2

