



Rozwiązanie zadania F 847.

Niech natężenie światła po przejściu przez pierwszy polaryzator wynosi I_1 . Zgodnie z prawem Malusa natężenie światła po przejściu przez drugi polaryzator wynosi $I_2 = I_1 \cos^2 \varphi$. Kąt między osiami polaryzacji drugiego i trzeciego polaryzatora wynosi $\frac{\pi}{2} - \varphi$. Natężenie światła za trzecim z polaryzatorów wynosi więc

$$\begin{aligned} I_3 &= I_2 \cos^2 \left(\frac{\pi}{2} - \varphi \right) = \\ &= I_1 \cos^2 \varphi \cos^2 \left(\frac{\pi}{2} - \varphi \right) = \\ &= \frac{I_1}{4} \sin^2(2\varphi). \end{aligned}$$

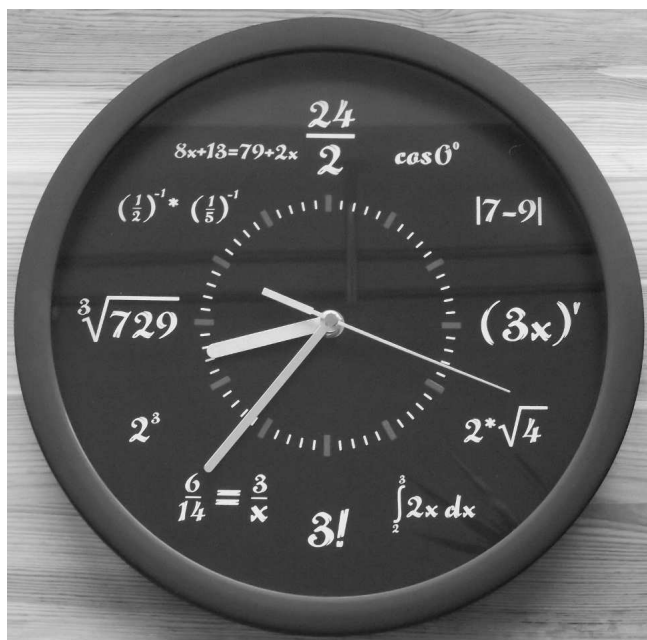
Natężenie światła będzie największe dla $\varphi = 45^\circ$, a najmniejsze (wygaszenie) dla $\varphi = 0$ i $\varphi = \pi/2$, to znaczy dla dodatkowego polaryzatora ustawionego równoległe do osi polaryzacji pierwszego albo ostatniego z polaryzatorów.

jest dowolna. Możemy je łączyć parami według dowolnego planu w postaci drzewa binarnego. Wykonując operacje w odpowiedniej kolejności, możemy uniknąć generowania dużych wyników pośrednich obliczeń. Niestety, okazuje się, że wybór optymalnego kształtu drzewa złączenia jest problemem NP-trudnym. Nie znamy deterministycznego algorytmu o wielomianowej złożoności czasowej. Trzeba więc przeszukać całą przestrzeń możliwych rozwiązań lub zastosować algorytmy sztucznej inteligencji. Jest to cena, jaką płacimy za prostotę modelu danych. Samych kształtów pełnych drzew binarnych o n liściach jest $\binom{2^n}{n} / (n + 1)$, a trzeba jeszcze przypisać poszczególnym liściom konkretne tabele. W systemie R firmy IBM zastosowano przeszukujący wszystkie plany algorytm o złożoności $O(n2^n)$. Niestety, nawet dla małych n (rzędu kilkunastu) jest to liczba zbyt duża, by taki algorytm mógł być zastosowany w praktyce. Dlatego współczesne systemy zarządzania bazami danych przeszukują tę przestrzeń heurystycznie, najczęściej za pomocą symulowanego wyzarcia lub algorytmów genetycznych. A przecież obliczenie odpowiedzi na zapytanie wymaga nie tylko złączania tabel...

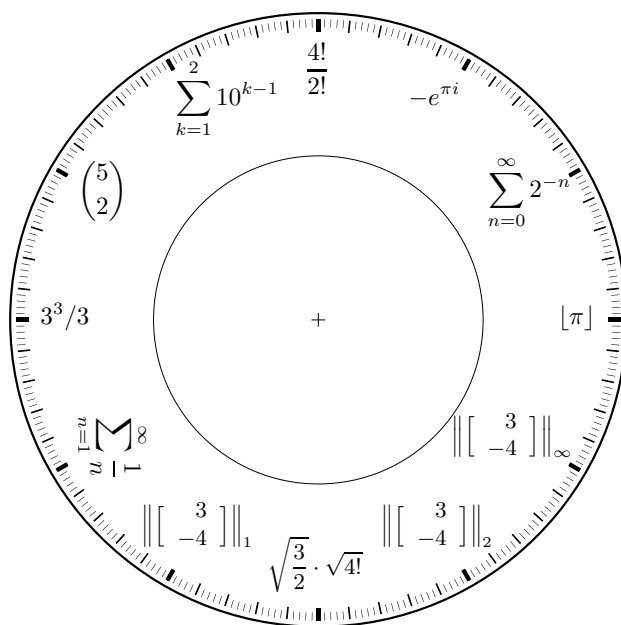
Mam nadzieję, że udało mi się wykazać, iż worek z danymi, taki, jaki zastosowano w USOS, jest niezwykle skomplikowanym tworem. Nie poruszyłem przy tym wielu zagadnień, takich jak indeksy (drzewiaste i haszowane struktury danych służące do wyszukiwania konkretnych wierszy w tabelach), (pół)automatyczne strojenie baz danych (w tym doboru indeksów), kwestia uprawnień użytkowników, zarządzanie fizycznymi nośnikami pamięci, archiwizacja i retencja danych itd. Konstrukcja bazy danych ma wiele wspólnego z konstrukcją cepa. Jest przysłowiowo – ale tylko pozornie – prosta. Każdy może zbudować sobie swój cep, a nawet bazę danych. Ale czy Twoja baza danych będzie działała wydajnie i bez utraty danych przez pół minuty czy kilka lat? Czy bijak Twojego cepa odłączy się od dźwierzaka po trzech czy po milionie uderzeń?

Zegar

W prezencie od znajomych dostałem zegar (rys. 1). Powiesiłem go sobie na ścianie i zacząłem wymyślać... inne zegary. Moja propozycja jest na rysunku 2.



Rys. 1



Rys. 2

Czytelników zachęcam do projektowania własnych tarcz zegarowych. Oczekujemy na nie do 1 marca 2014 roku pod adresem delta@mimuw.edu.pl. Najlepszy projekt (liczą się wybrane wyrażenia, mogą być też wzięte z fizyki, astronomii lub informatyki) zrealizuję (będzie działał) i wyślę autorowi.

Przemysław KICIAK