

Programowanie zachłanne

Poniższe zadania rozwiązuje się (mniej lub bardziej) zachłannie. Dla każdego z nich, oprócz programu, należy uzasadnić poprawność rozwiązania zachłannego.

- [CLR, p. 17.1] Problem wyboru zajęć (historyjka o kinomaniu i jak największej liczbie filmów do obejrzenia).
- Dany jest zbiór odcinków na prostej. Jaka jest minimalna liczba gwoździ, za pomocą których można je przybić do prostej? Każdy odcinek musi być przybity przynajmniej jednym gwoździem.
- [CLR, ćw 17.2-4] Problem tankowania na stacjach benzynowych.
- Napisz procedurę `vector<int> silnie(int n)`, która dla danej dodatniej liczby całkowitej n znajduje najkrótszy taki ciąg dodatnich liczb całkowitych $[x_1; x_2; \dots; x_k]$, że $x_1! + x_2! + \dots + x_k! = n$. Na przykład, dla `silnie(42)` poprawnym wynikiem jest n.p. `{3, 4, 3, 3}`.
- [PCh] Dany jest ciąg nawiasów, otwierających i zamykających. Napisz procedurę `nawiasy`, która obliczy minimalną liczbę nawiasów które należy obrócić, tak aby uzyskać poprawne wyrażenie nawiasowe. Jeżeli nie jest to możliwe, to poprawny wynik wynosi -1 .

```
enum nawias {Otwierajacy, Zamykajacy};
int nawiasy(vector<enum nawias>)
```

- W trakcie wakacji n informatyków pojechało razem na obóz informatyczny. Zamieszkali oni w domkach kempingowych stojących wzdłuż prostej drogi. Firma telekomunikacyjna chce im zapewnić dostęp do Internetu rozstawiając wzdłuż drogi nadajniki sieci bezprzewodowej. Jeden nadajnik może obsługiwać co najwyżej k domków znajdujących się od niego w odległości co najwyżej r . Napisz procedurę `int wakacje(int k, int r, vector<int>)`, która na podstawie liczb k i r , oraz list współrzędnych domków obliczy minimalną liczbę potrzebnych nadajników.
- Ciąg $[x_1; x_2; \dots]$ nazwiemy naprzemiennym, jeżeli:
 - $x_1 < x_2 > x_3 < x_4 > \dots$, lub
 - $x_1 > x_2 < x_3 > x_4 < \dots$.
 Ciąg jednoelementowy i pusty są również naprzemienne. Napisz procedurę `int naprzemienny(vector<int>)`, która mając dany ciąg liczb całkowitych obliczy długość najdłuższego jego naprzemiennego podciągu. Na przykład, `naprzemienny({4, 8, 6, 3, 4, 5, 3, 7, 9, 5, 7}) == 8`.
- Rozważamy ciągi liczb całkowitych (x_1, x_2, \dots, x_k) konstruowane zgodnie z następującymi regułami:
 - $x_1 = 1$,
 - $x_{i+1} = x_i + f_i$ dla pewnej liczby Fibonacciego f_i .
 Napisz procedurę `vector<int> ciag(int n)`, która dla danej dodatniej liczby całkowitej n wyznaczy najkrótszy taki ciąg (x_1, x_2, \dots, x_k) , że $x_k = n$.

- [XII OI, zadanie *Lot na marsa*] Bajtazar postanowił polecieć na Marsa, aby zwiedzić istniejące tam stacje badawcze. Wszystkie stacje na Marsie leżą na okręgu. Bajtazar ląduje w jednej z nich, a następnie porusza się za pomocą specjalnego pojazdu, który jest napędzany odpowiednim paliwem. Litry paliwa starcza na metr jazdy. Zapasy paliwa są jednak niewielkie – różne jego ilości znajdują się w różnych stacjach. Bajtazar może tankować paliwo na stacji, na której w danym momencie się znajduje, nie więcej jednak, niż dostępna tam jego ilość (pojemność baku jest nieograniczona). Musi mu to wystarczyć na dojazd do następnej stacji. Bajtazar musi zdecydować, gdzie powinien wylądować, tak żeby mógł zwiedzić wszystkie stacje. Na koniec Bajtazar musi wrócić do stacji, w której wylądował. W czasie podróży Bajtazar musi poruszać się po okręgu, stając w wybranym jednym z dwóch kierunków.
- Firma X podjęła się szeregu zobowiązań, z których wykonaniem zalega. Wykonanie każdego z nich zajmuje określoną liczbę dni. Za każdy dzień opóźnienia wykonania każdego z zobowiązań firma płaci określone kary. Napisz procedurę `harmonogram`, która obliczy optymalną kolejność wykonywania zaległych zobowiązań. Procedura ta otrzymuje ciąg par postaci: *(liczba dni potrzebnych do wykonania, kara za dzień zwłoki)*. Jej wynikiem powinien być ciąg tychże par w kolejności, w jakiej należy je wykonywać.
- [V OI, zadanie *Suma ciągu jedynekowego* (uproszczone)] Dana jest liczba $n \geq 0$ oraz $0 \leq s \leq \frac{n(n+1)}{2}$. Podaj podzbiór zbioru $\{1, \dots, n\}$, którego suma elementów jest równa s . Czy jest to zbiór o minimalnej liczbie elementów?
- Napisz procedurę `vector<int> podzial(int n)`, która dla danej nieujemnej liczby całkowitej przedstawi ją jako sumę jak największej liczby (dodatnich) różnych liczb Fibonacciego. Na przykład, `podzial(42) = {1, 2, 5, 13, 21}`.
- [IV OI, zadanie *Lotniska*] W n miastach (ponumerowanych od 1 do n) znajdują się lotniska. Każde lotnisko ma określoną przepustowość, tj. liczbę połączeń z innymi miastami. Połączenia takie są zwrotne, czyli połączenie z A do B jest

- jednocześnie połączeniem z B do A . Podaj połączenia realizujące **dokładnie** podane przepustowości, lub stwierdź, że się nie da. Dane mają postać ciągu uporządkowanego niemalejąco wg przepustowości miast.
- o wersja łatwiejsza: między dwoma miastami może być wiele połączeń,
 - o wersja trudniejsza: między dwoma miastami może być co najwyżej jedno połączenie.
14. Dana jest lista par liczb postaci (x, y) , $x < y$, opisujących (domknięte) odcinki alejki jakie mogą oświetlić poszczególne latarnie, oraz para liczb (a, b) , $a < b$, opisująca odcinek alejki, który chcemy oświetlić. Napisz procedurę `int ile(vector<pair<int, int>> t, int a, int b)`, która obliczy minimalną liczbę latarni, jakie są konieczne do oświetlenia danego odcinka alejki. Przykład: `ile({(1, 4), (7, 9), (-2, 3), (4, 9), (9, 12), (0, 5), (3, 7)}, (-1, 11)) == 4`
15. [X Ol, zadanie *Czekolada*] Dana jest tabliczka czekolady, którą należy połamać na części. Każde przełamanie kawałka czekolady jest obciążone pewnym kosztem. Koszt ten nie zależy od wielkości kawałka czekolady, ale od prostej, wzdłuż której łamiemy. Dla wszystkich prostych, na których leżą krawędzie części czekolady mamy stałe określające koszty łamania. Wyznacz minimalny koszt połamania czekolady na części.
16. [III Ol, zadanie *Wieża*] Na szachownicy $n \times n$ należy ustawić n wież tak, żeby żadne dwie się nie szachowały, a ponadto każda z tych wież ma wyznaczony prostokąt, w którym ma stać. Znajdź takie ustawienie, lub stwierdź, że się nie da.
17. [XI Ol, zadanie *Most*] Dana jest grupa osób, które chcą przejść nocą przez most. Mają oni jedną latarkę, która pozwala przejść jednej lub dwóm osobom. Każda osoba potrzebuje określonego czasu na przejście przez most. (Możesz założyć, że dane są posortowane.) Jeśli dwie osoby idą razem, to potrzebują tyle czasu, co wolniejsza z nich. Jaki jest minimalny czas potrzebny do przejścia wszystkich?
18. [XII Ol, zadanie *Samochodziki*; OSC, optymalna strategia wymiany stron] Jasio jest trzylatkiem i bardzo lubi bawić się samochodzikami. Jasio ma n różnych samochodzików. Wszystkie samochodziki leżą na wysokiej półce, do której Jasio nie dosięga. W pokoju jest mało miejsca, więc na podłodze może znajdować się jednocześnie co najwyżej k samochodzików. Jasio bawi się jednym z samochodzików leżących na podłodze. Oprócz Jasia w pokoju cały czas przebywa mama. Gdy Jasio chce się bawić jakimś innym samochodzikiem i jeśli ten samochodzik jest na podłodze, to sięga po niego. Jeśli jednak samochodzik znajduje się na półce, to musi mu go podać mama. Mama podając Jasiowi jeden samochodzik, może przy okazji zabrać z podłogi dowolnie wybrany przez siebie inny samochodzik i odłożyć go na półkę (tak, aby na podłodze nie brakło miejsca). Mama bardzo dobrze zna swoje dziecko i wie dokładnie, którymi samochodzikami Jasio będzie chciał się bawić. Dysponując tą wiedzą mama chce zminimalizować liczbę przypadków, gdy musi podawać Jasiowi samochodzik z półki. W tym celu musi bardzo rozważnie odkładać samochodziki na półkę. Napisz procedurę `samochodziki`, która dla danej liczby k oraz ciągu samochodzików, którymi zechce się bawić Jasio, obliczy minimalną liczbę razy, kiedy mama będzie musiała Jasiowi podać samochodzik.
19. [XV Ol, zadanie *Plakatowanie*] Rozważamy następującą łamigłówkę. Dana jest tablica nieujemnych liczb całkowitych $a = \{a_1, \dots, a_n\}$. Jeśli $a_i = a_{i+1} = \dots = a_{j-1} = a_j$ (dla $1 \leq i \leq j \leq n$), to w jednym ruchu można zmniejszyć wszystkie liczby a_i, a_{i+1}, \dots, a_j o dowolną dodatnią liczbę, nie większą niż a_i (tak żeby po zmniejszeniu nadal były nieujemne). Napisz procedurę: `ruchy`, która dla danej tablicy a wyznaczy najkrótszą sekwencję ruchów potrzebnych do wyzerowania wszystkich liczb a_1, \dots, a_n .

Last modified: Saturday, 21 January 2023, 1:26 PM

Contact us



Follow us

Contact site support

You are logged in as Witold Formański (Log out)

Data retention summary

Get the mobile app

Get the mobile app