

Zadanie 1 (10 pkt. na pracowni, później 5 pkt.). Napisz program, który będzie drukował na standardowym wyjściu prosty kalendarz na wybrany miesiąc.

Program powinien wczytywać ze standardowego wejścia trzy liczby całkowite dodatnie oznaczające rok, miesiąc, i dzień tygodnia, od którego ten miesiąc się zaczyna (1 – poniedziałek, 7 – niedziela). Wydrukować należy kolejne numery dni miesiąca (odpowiednio pooddzielane spacjami), z tygodniami w osobnych wierszach. Na przykład dla wejścia "2200 2 3" właściwym wyjściem jest

```
      1  2  3  4  5
 6   7  8  9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28
```

Jako dodatkowe utrudnienie najpierw wpisz te liczby do tablicy (rozmiaru 7×6 – nie przejmuj się tym, że miesiące zazwyczaj "mieszczą się" w 5, a czasami nawet w 4 tygodniach), ustawiając w komórkach "spoza miesiąca" jakąś wybraną specjalną wartość (np. 0 lub ujemną); dopiero na podstawie tej tablicy wydrukuj wyjście. Za rozwiązanie bez tego utrudnienia można dostać najwyżej 6 pkt. na pracowni (później 3 pkt.).

Uwagi i wskazówki: Rozwiązując to zadanie na pracowni możesz nie przejmować się poprawnością wczytania liczby przez funkcję `scanf`, ale wciąż należy sprawdzić, czy (poprawnie wczytana) liczba jest poprawna, tj. że miesiąc nie jest większy niż 12, a dzień tygodnia – niż 7. Do wypisywania przyda się specyfikator formatu `%2d`.

(W rzeczywistości rok i miesiąc oczywiście jednoznacznie określają dzień tygodnia, od którego ten miesiąc się zaczyna – osoby zainteresowane mogą poszperać za hasłem "kalendarz wieczny".)

Zadanie 2 (10 pkt.). Napisz program, który dla zadanej na wejściu naturalnej liczby $N \geq 12$ odpowie na pytanie: dla jakiej liczby naturalnej $M \leq N$ jest najwięcej trójkątów pitagorejskich (tj. prostokątnych o bokach całkowitej długości) o obwodzie M , i ile jest tych trójkątów?

Jeśli to maksimum osiągane jest dla kilku liczb M , wypisz dowolną z nich. Dla $N = 1000$ maksimum wynosi 8 i osiągane jest tylko dla liczby 840, a obliczenia raczej naiwnego programu wykonywane są na przeciętnym komputerze w mgnieniu oka, więc to nie jest zadanie na nie wiadomo jakie boksowanie się z matematyką. (Ale zawsze można się chwilę poboksować i wymyślić ciekawsze rozwiązanie, czemu nie.)

Zadanie 3 - Generowanie ciągu

Dana jest liczba n (nie większa od miliona) oraz $n-1$ zależności w postaci „-1” lub „1”. Chcemy na tej podstawie zbudować n -elementowy ciąg, zgodnie z następującymi zasadami:

- wszystkie liczby w ciągu muszą być naturalne dodatnie,
- jeżeli i -ta zależność to „-1” liczba na pozycji i musi być większa niż liczba na pozycji $i+1$ (ciąg jest w tym miejscu rosnący),

- jeżeli i -ta zależność to „1”, liczba na pozycji i musi być mniejsza niż liczba na pozycji $i+1$ (ciąg jest w tym miejscu malejący),
- suma elementów ciągu musi być jak najmniejsza.

Wejście

W pierwszej linii wejścia znajduje się dodatnia liczba naturalna $n \leq 1000000$. Druga linia wejścia składa się z $n-1$ oddzielonych spacjami 1 i -1.

Wyjście

Jedyna linia wyjścia powinna składać się z n oddzielonych spacjami liczb naturalnych dodatnich tworzących ciąg zgodnie z zasadami zadania.

Przykłady

A

Wejście:

5

-1 1 1 1

Wyjście

2 1 2 3 4

B

Wejście:

12

-1 -1 1 -1 -1 1 1 1 -1 -1 -1

Wyjście

3 2 1 3 2 1 2 3 4 3 2 1

C

Wejście

2

1

Wyjście

1 2