



# Zadanie: DZI

## Działka 2 [B]

PA 2017, runda 4. Dostępna pamięć: 128 MB. Limit czasu: 12 s.

23.11.2017

**Uwaga!** To jest zadanie rozproszone. Zanim zaczniesz je rozwiązywać, zapoznaj się z informacjami dotyczącymi tego typu zadań dostępnymi w serwisie internetowym zawodów.

Bajtazar dawno temu kupił spory obszar pola. Pole to było podzielone na kwadraciki jednostkowe, wśród których każde było albo użytkowe, albo nieużytkowe. Bajtazar pragnął rozpocząć uprawę na prostokątnej działce będącej częścią tego pola i złożonej tylko z kwadracików użytkowych\*.

Od tego czasu minęło już prawie piętnaście lat. Bajtazar niedawno sprzedał stary obszar pola i kupił nowy, o wiele większy. Chciałby teraz na zakupionym terenie stworzyć dowolną prostokątną działkę złożoną jedynie z kwadracików użytkowych. Na ile sposobów może to zrobić?

## Komunikacja z biblioteką

Aby móc rozwiązywać zadanie, należy na początku programu dopisać dyrektywę

```
#include "dzialka.h"
```

Dodana biblioteka udostępnia następujące funkcje:

- `int GetFieldHeight()`; – zwraca wysokość zakupionego pola.
- `int GetFieldWidth()`; – zwraca szerokość zakupionego pola.
- `int IsUsableCell(int r, int c)`; – dla podanych  $0 \leq r < \text{GetFieldHeight}()$ ,  $0 \leq c < \text{GetFieldWidth}()$  zwraca 1, jeśli kwadracik o współrzędnych  $(r, c)$  jest użytkowy lub 0 w przeciwnym przypadku.

## Wyjście

Dokładnie jedna instancja powinna wypisać jedną liczbę całkowitą – liczbę możliwych umiejscowień działki, to jest prostokąta złożonego jedynie z kwadracików użytkowych.

## Ograniczenia

- $1 \leq \text{GetFieldHeight}(), \text{GetFieldWidth}() \leq 75\,000$ .
- Każdy test będzie uruchomiony na 100 instancjach komputerów.
- Podany limit pamięci dotyczy pojedynczej instancji.
- Limit liczby wiadomości wychodzących z pojedynczej instancji: 1000.
- Limit łącznego rozmiaru wiadomości wychodzących z pojedynczej instancji: 8 MB.
- Wywołanie funkcji bibliotecznych zajmuje średnio nie dłużej niż 0.04 mikrosekundy.

## Testy przykładowe

Po wysłaniu rozwiązania od razu poznasz wyniki sprawdzania na poniższych testach przykładowych, uruchomionych na 100 instancjach.

- 0a: test przykładowy, opisany na następnej stronie.
- 0b: pole jest prostokątem o wysokości 2000 i szerokości 1536. Pole  $(r, c)$  jest użytkowe wtedy i tylko wtedy, gdy  $r + c$  **nie jest** wielokrotnością 100.
- 0c: pole jest kwadratem o boku 75 000. Pole  $(r, c)$  jest użytkowe dla wszystkich  $r \neq c$ .

\*Zainteresowanych odsyłamy do zadania *Działka*, które pojawiło się na IX Olimpiadzie Informatycznej. Można je znaleźć pod linkiem [szkopul.edu.pl/portal/problemset/oi/9](http://szkopul.edu.pl/portal/problemset/oi/9) albo w *Niebieskiej książeczce IX OI* dostępnej na stronie [oi.edu.pl](http://oi.edu.pl).

## Przykładowe wywołanie programu

Dla poniższego przebiegu programu:

Wywołanie funkcji	Zwrócona wartość
GetFieldHeight()	3
GetFieldWidth()	3
IsUsableCell(0, 0)	1
IsUsableCell(0, 1)	1
IsUsableCell(0, 2)	1
IsUsableCell(1, 0)	0
IsUsableCell(1, 1)	1
IsUsableCell(1, 2)	0
IsUsableCell(2, 0)	1
IsUsableCell(2, 1)	1
IsUsableCell(2, 2)	1

poprawnym wynikiem jest:

16

## Eksperymenty

W dziale *Pliki* możesz znaleźć przykładową bibliotekę testującą. Składa się ona z plików `dzialka.h` oraz `dzialka.c`. Znajduje się tam też przykładowy (błędny) program `dzi_wrong.cpp` korzystający z interfejsu udostępnianego przez bibliotekę.

Aby skompilować Twój program `dzi.cpp` z biblioteką, należy wydać następującą komendę (zakładamy, że `rpa` jest zdefiniowane jak w poradniku do zadań rozproszonych):

```
rpa build --source dzi.cpp --library dzialka.c
```

Tak skompilowany program wczytuje test z przykładowego wejścia. Pierwszy wiersz pliku wejściowego zawiera dwie liczby:  $n = \text{GetFieldHeight}()$  oraz  $m = \text{GetFieldWidth}()$ . Kolejnych  $n$  wierszy zawiera  $m$ -literowe ciągi znaków;  $j$ -ty znak w  $i$ -tym wierszu opisuje zawartość kwadracika  $(i, j)$  i powinien być równy 1, jeśli kwadracik jest używalny, zaś 0 w przeciwnym przypadku. Dla ułatwienia, test przykładowy został przetłumaczony do powyższego formatu (plik `dzi0.in` w archiwum powyżej).