

# Zadanie D - Operacje tablicowe

Punktów do uzyskania: 10

Język programowania: C

## Opis zadania

Działanie zadania polega na wczytywaniu trzysnakowych kodów operacji (z ewentualnymi argumentami oddzielnymi pojedynczymi spacjami), aż do odczytania kodu operacji kończącej działanie programu oraz implementacją opisanych dalej operacji. Operacje manipulują w pamięci komputera dwuwymiarową strukturą liczb całkowitych typu `int`, złożoną ze zmiennej liczby wierszy o zmiennej liczbie elementów. Numeracja wierszy zawsze rozpoczyna się wartością 0 i zakłada, że wszystkie niepuste wiersze występują na początku. Podobnie, numeracja elementów w wierszu zaczyna się od wartości 0, a kolejne element tworzą spójny początkowy ciąg kolumn.

## Specyfikacja operacji

- **AFR**  $w\ e_0 \dots e_{w-1}$   
Na początek struktury dodaj niepusty wiesz liczący  $w$  elementów o podanych kolejno wartościach.
- **ALR**  $w\ e_0 \dots e_{w-1}$   
Na koniec struktury dodaj niepusty wiesz liczący  $w$  elementów o podanych wartościach.
- **AFC**  $h\ e_0 \dots e_{h-1}$   
Do  $h$  pierwszych wierszy struktury dodaj na początek kolumnę złożoną z podanych elementów. Jeżeli  $h$  jest większe od ilości wierszy, stwórz nowe wiersze z początkowymi pojedynczymi elementami jak podane w argumentach operacji.
- **ALC**  $h\ e_0 \dots e_{h-1}$   
Do  $h$  pierwszych wierszy struktury dodaj na koniec wiersza elementy podane w parametrach. Jeżeli  $h$  jest większe od ilości wierszy, stwórz nowe wiersze z początkowymi pojedynczymi elementami jak podane w argumentach operacji.
- **IBR**  $r\ w\ e_0 \dots e_{w-1}$   
Wstaw przed wierszem o numerze  $r$  nowy wiersz o dodatniej liczbie elementów  $w$ , złożony z podanych elementów  $e_0 \dots e_{w-1}$ . W przypadku braku wiersza o numerze  $r$  operacja nie przynosi żadnych skutków.
- **IAR**  $r\ w\ e_0 \dots e_{w-1}$   
Wstaw po wierszu o numerze  $r$  nowy wiersz o dodatniej liczbie elementów  $w$  złożony z elementów  $e_0 \dots e_{w-1}$ . W przypadku braku wiersza o numerze  $r$  operacja nie przynosi żadnych skutków.
- **IBC**  $c\ h\ e_0 \dots e_{h-1}$   
W początkowych  $h$  wierszach przed kolumną o numerze  $c$  wstaw nową kolumnę złożoną z elementów  $e_0 \dots e_{h-1}$ . Po-

nadto:

- Wartość  $h$  jest dodatnia.
- Jeżeli wiersz liczy mniej elementów niż numer wstawianej kolumny, elementy dodawane są na koniec wiersza.
- W przypadku liczby wierszy  $h$  mniejszej od liczby dodawanych elementów, operacja dodaje nowe wiersze, zaś elementy dodawanej kolumny stają się ich pierwszymi elementami.
- Wiersze o numerze większym od liczby elementów dodawanej kolumny nie zmieniają swojej zawartości.
- **IAC**  $c\ h\ e_0 \dots e_{h-1}$   
W początkowych  $n$  wierszach po kolumnie o numerze  $c$  wstaw nową kolumnę złożoną z elementów  $e_0 \dots e_{h-1}$ . Ponadto:
  - Wartość  $h$  jest dodatnia.
  - Jeżeli wiersz liczy mniej elementów niż numer wstawianej kolumny, elementy dodawane są na koniec wiersza.
  - W przypadku liczby wierszy  $h$  mniejszej od liczby dodawanych elementów, operacja dodaje nowe wiersze, zaś elementy dodawanej kolumny stają się ich pierwszymi elementami.
  - Wiersze o numerze większym od liczby elementów dodawanej kolumny nie zmieniają swojej zawartości.
- **SWR**  $r\ s$   
Zamień wartościami wiersze o nieujemnych numerach  $r$  oraz  $s$ . W przypadku nieobejmowania przez strukturę wierszy danych argumentami operacja nie powoduje żadnych zmian.
- **SWC**  $c\ d$   
Zamień wartościami kolumny o nieujemnych numerach  $c$  oraz  $d$ . Jeżeli wiersz nie zawiera elementu o podanym numerze kolumny działanie dla danego wiersza nie powoduje żadnych zmian.
- **DFR**  
Skasuj pierwszy wiersz struktury, a jeżeli struktura jest pusta, operacja nie przynosi żadnych skutków.
- **DLR**  
Skasuj ostatni wiersz struktury, a jeżeli struktura jest pusta, działanie nie przynosi żadnych skutków.
- **DFC**  
Skasuj pierwszą kolumnę wszystkich wierszy, zaś wiersze tracące elementy usuń ze struktury. Jeżeli struktura jest pusta, działanie nie przynosi żadnych skutków.
- **DLC**  
Skasuj ostatnie elementy wszystkich wierszy, zaś wiersze tracące elementy usuń ze struktury. Jeżeli struktura jest pusta, działanie nie przynosi żadnych skutków.
- **RMR**  $r$   
Usuń wiersz o nieujemnym numerze  $r$ , zaś w przypadku braku wiersza o podanym numerze, operacja nie przynosi żadnych skutków.

- **RMC**  $c$   
Ze wszystkich wierszy usuń kolumnę o numerze  $c$ . Wiersze nieposiadające elementu o numerze kasowanej kolumny nie zmieniają zawartości, zaś wiersze stojące się w wyniku operacji pustymi są usuwane ze struktury.
- **ISB**  $r\ c\ h\ w\ e_{0,0} \dots e_{0,w-1} \dots e_{h-1,0} \dots e_{h-1,w-1}$   
W strukturę rozpoczynając w wierszu o numerze  $r$  i kolumnie o numerze  $c$  wstaw blok elementów liczący  $h$  wierszy i  $w$  kolumn, o elementach danych w linii argumentu kolejno wierszami  $e_{0,0} \dots e_{0,w-1}, \dots, e_{h-1,0}, \dots, e_{h-1,w-1}$ . Ponadto:
  - Jeżeli ostatni numer wiersza struktury jest mniejszy od numeru wiersza danego argumentem, dane kolejnych wierszy stają się nowo dodawanymi wierszami po ostatnim wierszu.
  - Jeżeli ilość dodawanych wierzy z uwzględnieniem numeru wiersza danego argumentem przekracza ilość aktualnych wierszy dodawane są kolejne nowe wiersze o początkowych kolumnach.
  - Jeżeli wiersz nie zawiera kolumny o numerze podanej argumentem zawartość wiersza argumentów jest dodawana na koniec danego wiersza.
- **RMB**  $r\ h\ c\ w$   
Usuwa blok elementów obejmujący wiersze począwszy od nieujemnego numeru  $r$  w dodatniej ilości  $h$  i kolumny począwszy od nieujemnego numeru  $c$  w dodatniej ilości  $w$ . Wiersze tracące wszystkie elementy są usuwane ze struktury.
- **PRT**  
Wyświetla na standardowym wyjściu zawartość struktury. Każdy wiersz wyświetlany jest w osobnej linii, zaś elementy są oddzielone pojedynczą spacją.
- **END**  
Kończy działanie programu, zwalniając całą pobieraną pamięć.

## Dodatkowe uwarunkowania

- Nazwą pliku źródłowego musi być `source.c`.
- Pierwsza linia kodu źródłowego MUSI w komentarzu zawierać imię i nazwisko autora rozwiązania.
- Jedynymi dozwolonymi do włączenia plikami nagłówkowymi są pliki `stdio.h` oraz `stdlib.h`.
- Zabronione jest używanie słów kluczowych `struct` oraz `union`.
- Zakazane jest używanie zmiennych globalnych.
- Zakazane jest używanie znaków kwadratowych nawiasów oraz ich zastępników.
- Cała pamięć alokowana dynamicznie musi być zwrócona.
- Wczytywane argumenty operacji są zawsze wartościami odpowiedniego typu, pozwalającymi wykonać działanie przy warunkach określonych specyfiką operacji.
- Białe znaki wyjścia nie mają znaczenia dla sprawdzania poprawności działania.