

Zadanie B - Operacje tablicowe

Punktów do uzyskania: 7

Opis zadania

Zadanie polega na implementacji trzech operacji modyfikujących jednowymiarową tablicę cykliczną. Cykliczność oznacza, że elementem poprzedzającym pierwszy element jest element ostatni, zaś elementem następującym po elemencie ostatnim jest element pierwszy. W konsekwencji numeracja elementów jest cykliczna, zatem używany indeks może być dowolną liczbą całkowitą, zaś faktyczny indeks jest wyliczany poprzez pozycję w cyklu względem faktycznych elementów tablicy. Dla przykładu, przy tablicy pięcioelementowej (o indeksach od 0 do 4), wartości indeksu wynoszące -10, -5, 5, 10 (ogólnie postaci 5*k*) oznaczają indeks pierwszego elementu (indeks 0), zaś wartości indeksu -6, -1, 9 (ogólnie postaci 5*k*+4) oznaczają indeks ostatniego elementu (indeks 4).

Specyfikacja operacji

• **Odwracanie**

Operacja polega na odwróceniu kolejności elementów wszystkich kolejnych fragmentów o zadanej długości i zadany indeks pierwszego elementu. Ewentualnie pozostały fragment krótszy niż zadana długość fragmentu nie jest odwracany.

Dla przykładu, dla dwunastoelementowej tablicy o wartościach:

0.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.
11	12	13	14	15	16	17	18	19	20	21	22

operacja zadana indeksem pierwszego elementu fragmentu wynoszącym 22 (faktycznym indeksem 10) i długością fragmentu wynoszącą 5 prowadzi do tablicy:

0.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.
11	22	21	18	17	16	15	14	19	20	13	12

• **Zamiana parami**

Operacja zamienia każde dwa kolejne sąsiednie fragmenty o zadanej długości i zadany cyklicznie indeksem pierwszego elementu. Ewentualnie pozostały fragment niedopełniający długością możliwości zamiany z sąsiednim fragmentem pozostaje bez zmian. Przykładowo, dla dwunastoelementowej tablicy o wartościach:

0.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.
11	12	13	14	15	16	17	18	19	20	21	22

operacja zadana wartością indeksu pierwszego elementu fragmentu wynoszącą 9, długością fragmentu wynoszącą 4 prowadzi

do tablicy:

0.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.
15	20	21	22	11	16	17	18	19	12	13	14

• **Przesunięcie**

Operacja w każdym kolejnym fragmencie o zadanej długości i zadanym indeksie pierwszego elementu powoduje cykliczne przesunięcie elementów o zadaną ilość pozycji. Dodatnia wartość przesunięcia oznacza przesunięcie w kierunku rosnących indeksów, zaś ujemna wartość przesunięcia oznacza przesunięcie w kierunku malejących indeksów. Przesunięciu podlega również ewentualnie pozostały fragment krótszy niż zadana długość.

Dla przykładowej dwunastoelementowej tablicy o elementach:

0.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.
11	12	13	14	15	16	17	18	19	20	21	22

operacja zadana wartością indeksu pierwszego elementu fragmentu wynoszącą -9 (czyli dla faktycznej tablicy indeksem 3), długością fragmentu wynoszącą 5 i wartością przesunięcia wynoszącą -7 prowadzi do tablicy:

0.	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.
20	13	12	16	17	18	14	15	21	22	11	19

Wejście

- Dane są wczytywane ze standardowego wejścia.
- Pierwsza linia danych wejściowych zawiera dodatnia liczbę całkowitą określającą ilość sesji działań, po której następują dane każdej sesji.
- Dane sesji obejmują:
 - Linie złożoną z dodatniej liczby całkowitej określającej ilość elementów tablicy z następującymi po niej liczbami całkowitymi w ilości równej pierwszej danej liczbie i stanowiącymi elementy tablicy.
 - Umieszczone w osobnych liniach określenia rodzaju operacji wraz z parametrami zależnymi od operacji.
 - Pojedynczą literę F w jednej linii dla oznaczenia końca operacji.
- Określenia operacji obejmują:
 - Dla operacji odwracania literę R z następującą dowolną liczbą całkowitą określającą indeks pierwszego elementu fragmentu i następującą całkowitą liczbą nieujemną określającą długość fragmentu.
 - Dla operacji przesunięcia literę C z następującą dowolną liczbą całkowitą określającą indeks pierwszego elementu fragmentu, następującą całkowitą liczbą nieujemną określającą długość fragmentu i następującą kolejną dowolną liczbą całkowitą określającą przesunięcie.

- Dla operacji zamiany literę S z następującą dowolną liczbą całkowitą określającą indeks pierwszego elementu fragmentu i następującą całkowitą liczbą nieujemną określającą długość fragmentu.

Wyjście

- Dane wypisywane są na standardowe wyjście.
- Dla każdej sesji operacji program wypisuje zawartość tablicy przed rozpoczęciem działań i po ich zakończeniu.
- Wypisanie tablicy oznacza jedną linię z wartościami oddzielnymi pojedynczą spacją.

Dodatkowe uwarunkowania

- Pierwsza linia kodu źródłowego MUSI w komentarzu w standardzie języka C++ (dwa znaki ukośnika) zawierać imię i nazwisko autora rozwiązania.
- Jedynym dozwolonym do włączenia plikiem nagłówkowym jest plik iostream.
- **W implementacji każdej sesji możliwe jest użycie tylko jednej tablicy, może to być JEDYNIIE tablica z wczytanymi danymi, o długości wynikającej z ilości danych. Tym samym zabronione jest używanie jakichkolwiek pomocniczych tablic, a nawet używanie elementów tablicy spoza wczytanego zakresu.**
- W całym kodzie źródłowym zabronione jest używanie:
 - Słów **for**, **break** oraz **continue**.
 - **Własnych podprogramów**.
 - Słów **string**, **struct**, **class**.
 - Pamięci dynamicznej.
- Jakakolwiek próba obejścia powyższych warunków skutkuje dyskwalifikacją rozwiązania i wyzerowaniem punktacji po upływie terminu oddania zadania.

Przykład wejścia i odpowiadającego wyjścia

wejście	wyjście
2	10 11 12 13 14 15
6 10 11 12 13 14 15	15 12 11 10 13 14
R -10 10	20 21 22 23 24 25 26 27
C -3 4 -7	26 23 24 21 22 27 20 25
R 5 7	
S -3 6	
S -8 7	
F	
8 20 21 22 23 24 25 26 27	
C 1 4 18	
R 7 10	
S -15 6	
F	