

Zestaw C-21:

Organizacja plików: Wszystkie pliki oddawane do sprawdzenia należy zapisać we wspólnym folderze o nazwie będącej numerem indeksu, umieszczonym na pulpicie.

Oddajemy tylko źródła programów (pliki o rozszerzeniach .cpp i .h)!!!

Zad. 1: Napisać bibliotekę udostępniającą:

- typ Sznur będący dynamiczną listą łączoną przechowującą liczby całkowite;
- funkcję wypisz wypisującą zawartość sznura;
- funkcję wstaw dodającą na końcu sznura element zawierający liczbę podaną jako parametr;
- funkcję usun usuwającą ze sznura wszystkie elementy o wartościach większych niż średnia z wartości sznura znajdujących się w co M-tym elemencie tego sznura począwszy od wartości na pozycji M (M jest liczbą całkowitą dodatnią będącą parametrem funkcji, w przypadku gdy brak jest elementów do obliczenia powyższej średniej, sznur pozostawiany jest bez zmiany). Pamięć zajmowana przez usuwane elementy ma zostać zwolniona. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp);

oraz program testujący działanie biblioteki poprzez pobranie sznura (funkcja wstaw; użytkownik podaje liczby, które mają być wstawione do sznura i decyduje o zakończeniu podawania), wypisanie jego zawartości (funkcja wypisz; przykład: dla zestawu liczb wejściowych 4,7,5,1,6,6,7,1,4,2,9,1,1 sznur uzyskuje postać 4,7,5,1,6,6,7,1,4,2,9,1,1), a następnie, po pobraniu od użytkownika wartości M, usunięcie elementów przechowujących wartości większe niż średnia z co M-tego elementu w sznurze (funkcja usun) i ponowne wypisanie jego zawartości (przykład: dla M=2 sznur 4,7,5,1,6,6,7,1,4,2,9,1,1 uzyska postać 1,1,2,1,1, a dla M=3 ten sam sznur zostanie zredukowany do postaci 4,1,1,4,2,1,1).

Zad. 2: Rozszerzyć bibliotekę z zad.1 o funkcję przeniesTrojki, która modyfikuje sznur poprzez przepięcie na koniec sznura wszystkich rozłącznych trójek sąsiednich elementów o różnych wartościach (rozłączne trójki są wybierane tak, że pierwszy element sznura jest pierwszym elementem pierwszej trójki, a w przypadku sznura o długości nie podzielnej przez 3 jego ostatnie elementy nie wchodzą w skład żadnej trójki). Przepinanie ma być wykonywane tak, aby kolejność przepinanych trójek była zachowana, ale kolejność elementów w każdej przenoszonych trójce została odwrócona. Przykład: sznur 3,6,6,4,7,5,7,3,2,9,7,7,4,2 ma zostać przekształcony do postaci 3,6,6,9,7,7,4,2,5,7,4,2,3,7. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zad. 3: Rozszerzyć bibliotekę z zad.1 o funkcję wymienPary, modyfikującą dany sznur oraz drugi sznur S2 podany jako parametr, poprzez przepięcie między nimi co drugiej rozłącznej pary sąsiednich elementów w następujący sposób: druga pełna para elementów z „tego” sznura (oznaczonego w dalszym opisie przez S1) ma zostać przepięta w miejsce drugiej pełnej pary elementów sznura S2 i odwrotnie, analogicznie z elementami czwartych pełnych par w obu sznurach, szóstych, ósmych, itd., przy czym elementy nadmiarowych par w sznurach nie są przenoszone (przykłady: dla S1=1,2,7,7,1,3,2,4,5 i S2=3,2,3,6,3,4,8,6,5,6,2 wynikiem będzie S1=1,2,3,6,1,3,8,6,5 i S2=3,2,7,7,3,4,2,4,5,6,2, dla S1=4,2,3,1 i S2=1,2,3 sznury te pozostaną bez zmian). Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zestaw C-22:

Organizacja plików: Wszystkie pliki oddawane do sprawdzenia należy zapisać we wspólnym folderze o nazwie będącej numerem indeksu, umieszczonym na pulpicie.

Oddajemy tylko źródła programów (pliki o rozszerzeniach .cpp i .h)!!!

Zad. 1: Napisać bibliotekę udostępniającą:

- typ Sznur będący dynamiczną listą łączoną przechowującą liczby całkowite;
- funkcję wypisz wypisującą zawartość sznura;
- funkcję wstaw wstawiającą do sznura element zawierający liczbę podaną jako parametr tak, aby sznur był uporządkowany niemalejąco (zakładając, że sznur „na wejściu” jest już uporządkowany w taki sposób);
- funkcję usun rozpatrującą rozłączne czwórki elementów sznura i usuwającą całą czwórkę jeśli jej trzeci element jest podzielny przez pierwsze dwa, albo tylko ostatni element czwórki w przeciwnym przypadku. W przypadku sznura o długości nie podzielnej przez 4 końcowe elementy (nie wchodzące w skład żadnej czwórki – czwórki wybierane są tak, że pierwszy element sznura jest początkowym elementem pierwszej czwórki) nie są modyfikowane. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp);

oraz program testujący działanie biblioteki poprzez pobranie sznura (funkcja wstaw; użytkownik podaje liczby, które mają być wstawione do sznura i decyduje o zakończeniu podawania), wypisanie jego zawartości (funkcja wypisz; przykład: dla zestawu liczb wejściowych 1,3,6,1,3,7,2,6,3,1,1 sznur uzyskuje postać 1,1,1,1,2,3,3,3,6,6,7), a następnie usunięcie w sznurze rozłącznych czwórek elementów lub ostatnich elementów tych czwórek (funkcja usun) i ponowne wypisanie jego zawartości (przykład: sznur 1,1,1,1,2,3,3,3,6,6,7 zostanie przekształcony do postaci 2,3,3,6,6,7, sznur 2,3,6,7,8,8,8,9,9,10,90,100 stanie się pusty).

Zad. 2: Rozszerzyć bibliotekę z zad.1 o funkcję przenieszKrokiem, która modyfikuje sznur poprzez przecięcie wszystkich elementów tego sznura znajdujących się na co K-tej pozycji w sznurze począwszy od elementu na pozycji K (K jest liczbą całkowitą dodatnią będącą parametrem funkcji), naprzemiennie na początek lub koniec sznura: pierwszy na początek, następny na koniec, kolejny na początek, itd. Przepinanie elementów ma odwrócić ich kolejność względem tej w oryginalnym sznurze w ramach grupy przepinanej na początek oraz niezależnie w ramach grupy przepinanej na koniec sznura (porządek elementów w każdej parze też ma zostać zachowany). Przykład: dla sznura przechowującego liczby 1,1,2,2,5,5,5,5,6,6,6,8,8,9,9,9 i K=2 dostaniemy wynik 9,6,5,1,1,2,5,5,6,6,8,9,9,8,5,2, dla K=20 powyższy sznur pozostanie bez zmiany, dla K=1 uzyska postać 9,8,6,6,5,5,2,1,9,9,8,6,5,5,2,1. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zad. 3: Rozszerzyć bibliotekę z zad.1 o funkcję wymienGrupy, modyfikującą dany sznur oraz drugi sznur S2 podany jako parametr w następujący sposób: pierwsza grupa sąsiednich elementów z „tego” sznura (oznaczonego w dalszym opisie przez S1) przechowujących jednakowe wartości jest zamieniana z pierwszą grupą sąsiednich jednakowych elementów sznura S2, następnie trzecia grupa elementów o jednakowych wartościach z S1 jest zamieniana z trzecią taką grupą sznura S2, dalej analogicznie dla grup piątych, siódmych, itd. Pojedyncze elementy o unikalnych wartościach w danym sznurze też stanowią grupy. Nadmiarowa końcówka sznura o większej liczbie grup nie jest przenoszona. Przykład: dla sznurów S1=1,2,2,4,4,4,4,5,6,6,7 i S2=2,2,3,3,3,4,6,6,6,8,8 wynikiem będzie S1=2,2,2,2,4,5,8,8,7 i S2=1,3,3,3,4,4,4,4,6,6,6,6, dla S1=2,2 i S2=1,3,5 wynikiem będzie S1=1 oraz S2=2,2,3,5). Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zestaw C-23:

Organizacja plików: Wszystkie pliki oddawane do sprawdzenia należy zapisać we wspólnym folderze o nazwie będącej numerem indeksu, umieszczonym na pulpicie.

Oddajemy tylko źródła programów (pliki o rozszerzeniach .cpp i .h)!!!

Zad. 1: Napisać bibliotekę udostępniającą:

- typ `Sznur` będący dynamiczną listą łączoną przechowującą liczby całkowite;
- funkcję `wypisz` wypisującą zawartość sznura;
- funkcję `wstaw` wstawiającą do sznura element zawierający liczbę podaną jako parametr tak, aby sznur był uporządkowany nierosnąco (zakładając, że sznur „na wejściu” jest już uporządkowany w taki sposób);
- funkcję `usun` rozpatrującą rozłączne trójki elementów sznura podanego jako parametr i usuwającą pierwszy element trójki jeżeli wartość znajdująca się w elemencie środkowym trójki jest o 1 mniejsza od wartości w pierwszym elemencie tej trójki, lub usuwającą pierwszy i ostatni element w trójce w przeciwnym przypadku. W przypadku sznura o długości nie podzielnej przez 3 końcowe elementy (nie wchodzące w skład żadnej trójki – trójki wybierane są tak, że pierwszy element sznura jest początkowym elementem pierwszej trójki) nie są usuwane. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp);

oraz program testujący działanie biblioteki poprzez pobranie sznura (funkcja `wstaw`; użytkownik podaje liczby, które mają być wstawione do sznura i decyduje o zakończeniu podawania), wypisanie jego zawartości (funkcja `wypisz`; przykład: dla zestawu liczb wejściowych 1,3,6,1,3,7,2,6,3,1,1 sznur uzyskuje postać 7,6,6,3,3,3,2,1,1,1,1), a następnie usunięcie w sznurze wybranych elementów z rozłącznych trójek (funkcja `usun`) i ponowne wypisanie jego zawartości (przykład: sznur 7,6,6,3,3,3,2,1,1,1,1,1 zostanie przekształcony do postaci 6,6,3,1,1,1,1, a sznur 7,5,5,5,4,4,4,2,1 zostanie przekształcony do postaci 5,4,4,2).

Zad. 2: Rozszerzyć bibliotekę z zad.1 o funkcję `rozrzucPary`, która modyfikuje sznur poprzez przepięcie wszystkich rozłącznych par o różnicy pierwszej i drugiej wartości w parze mniejszej od nieujemnej całkowitej wartości N będącej parametrem funkcji, naprzemiennie na początek lub koniec sznura: pierwszą na początek, następną na koniec, kolejną na początek, itd. (pary wybierane są tak że pierwszy element sznura jest pierwszym elementem pierwszej pary, a przypadku sznura nieparzystej długości jego ostatni element nie jest elementem żadnej pary). Przepinanie par ma zachować ich kolejność względem tej w oryginalnym sznurze w ramach grupy przepinanej na początek oraz niezależnie w ramach grupy przepinanej na koniec sznura (porządek elementów w każdej parze też ma zostać zachowany). Przykład: dla sznura przechowującego liczby 9,9,9,8,8,6,6,6,5,5,5,5,2,2,1,1 i $N=2$ dostaniemy wynik 9,9,6,6,5,5,8,6,5,2,1,9,8,6,5,2,1, dla $N=0$ powyższy sznur pozostanie bez zmiany, dla $N=5$ uzyska postać 9,9,8,6,6,5,5,2,1,9,8,6,6,5,5,2,1. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zad. 3: Rozszerzyć bibliotekę z zad.1 o funkcję `przeniesGrupy`, modyfikującą dany sznur oraz drugi sznur $S2$ podany jako parametr w następujący sposób: grupy sąsiednich elementów z „tego” sznura (oznaczonego w dalszym opisie przez $S1$) przechowujących jednakowe wartości i mające parzystą długość mają zostać przepięte do $S2$, a grupy jednakowych elementów sznura $S2$ mające nieparzystą długość mają zostać przepięte do sznura $S1$. Przepinanie ma zostać wykonane tak, aby sznury były nadal uporządkowane nierosnąco (przykład: dla sznurów $S1=7,6,6,5,4,4,4,2,2,1$ i $S2=8,8,6,6,6,4,4,4,2,2$ wynikiem będzie $S1=7,6,6,6,5,4,4,4,1$ i $S2=8,8,6,6,4,4,4,4,2,2,2,2$, dla $S1=5,5,5,4,3,3,3$ i $S2=7,7,7,6,1$ wynikiem będzie $S1=7,7,7,6,5,5,5,4,3,3,3,1$ oraz pusty sznur $S2$). Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zestaw C-33:

Organizacja plików: Wszystkie pliki oddawane do sprawdzenia należy zapisać we wspólnym folderze o nazwie będącej numerem indeksu, umieszczonym na pulpicie.

Oddajemy tylko źródła programów (pliki o rozszerzeniach .cpp i .h)!!!

Zad. 1: Napisać bibliotekę udostępniającą:

- typ Sznur będący dynamiczną listą łączoną przechowującą liczby całkowite;
- funkcję wypisz wypisującą zawartość sznura;
- funkcję wstaw wstawiającą do sznura element zawierający liczbę podaną jako parametr tak, aby sznur był uporządkowany nierosnąco (zakładając, że sznur „na wejściu” jest już uporządkowany w taki sposób);
- funkcję usun rozpatrującą rozłączne czwórki elementów sznura podanego jako parametr i usuwającą całą czwórkę jeśli dwa środkowe jej elementy mają różną wartość, albo dwa środkowe elementy czwórki w przeciwnym przypadku. W przypadku sznura o długości nie podzielnej przez 4 końcowe elementy (nie wchodzące w skład żadnej czwórki) nie są modyfikowane (czwórki wybierane są tak, że pierwszy element sznura jest początkowym elementem pierwszej czwórki). Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp);

oraz program testujący działanie biblioteki poprzez pobranie sznura (funkcja wstaw; użytkownik podaje liczby, które mają być wstawione do sznura i decyduje o zakończeniu podawania), wypisanie jego zawartości (funkcja wypisz; przykład: dla zestawu liczb wejściowych 1,3,6,1,3,7,2,6,3,1,1 sznur uzyskuje postać 7,6,6,3,3,3,2,1,1,1,1), a następnie usunięcie w sznurze rozłącznych czwórek elementów lub środkowych elementów tych czwórek (funkcja usun) i ponowne wypisanie jego zawartości (przykład: sznur 7,6,6,3,3,3,2,1,1,1,1, zostanie przekształcony do postaci 7,3,1,1,1, sznur 9,9,7,6,6,6,5,5 stanie się pusty, sznur 5,5,5,5,4,4,2,1 zostanie przekształcony do postaci 5,5,4,2,1).

Zad. 2: Rozszerzyć bibliotekę z zad.1 o funkcję rozrzucSrodek, która modyfikuje sznur poprzez wpięcie do początkowej części sznura złożonej z K elementów i końcowej części sznura złożonej z K elementów (gdzie K jest liczbą całkowitą dodatnią będącą parametrem funkcji) wszystkich elementów z pozostałej, środkowej części sznura w następujący sposób: pierwszy element z środkowej części sznura (tj. części nie należącej do jego „K-końcówek”) jest wpinany przed pierwszym elementem jego „K-początku” (czyli przed pierwszym elementem sznura), drugi element z środkowej części sznura - przed pierwszym elementem jego „K-końca”, trzeci – za pierwszym elementem „K-początku”, czwarty – za pierwszym elementem „K-końca”, piąty – za drugim elementem „K-początku”, szósty – za drugim elementem „K-końca” itd.; przy czym wszystkie elementy stojące na nieparzystych pozycjach środkowej części i nie przepięte wcześniej mają zostać umieszczone za ostatnim elementem „K-początku” (czyli za K-tym elementem pierwotnego sznura) z zachowaniem swojej oryginalnej kolejności, a wszystkie elementy stojące na parzystych pozycjach środkowej części i nie przepięte wcześniej mają zostać umieszczone za ostatnim elementem „K-końca” (czyli za ostatnim elementem pierwotnego sznura), również z zachowaniem swojej oryginalnej kolejności. Sznur o długości mniejszej niż $2K+1$ pozostawiany jest bez zmiany. Przykład: dla sznura przechowującego liczby 9,8,7,6,5,4,3,2,1,0 i $K=3$ dostaniemy wynik 6,9,4,8,7,5,2,3,1,0, dla $K=2$ powyższy sznur uzyska postać 7,9,5,8,3,6,1,4,0,2, a dla $K=1$ – postać 8,9,6,4,2,7,0,5,3,1. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zad. 3: Rozszerzyć bibliotekę z zad.1 o funkcję przeniesMGrupy, modyfikującą dany sznur oraz drugi sznur S2 podany jako parametr w następujący sposób: grupy sąsiednich elementów z „tego” sznura (oznaczonego w dalszym opisie przez S1) przechowujących jednakowe wartości i mające długość podzielną przez M (gdzie M jest liczbą całkowitą dodatnią będącą parametrem funkcji) mają zostać przepięte do S2, a grupy jednakowych elementów sznura S2 mające długość nie podzielną przez M mają zostać przepięte do sznura S1. Przepinanie ma zostać wykonane tak, aby sznury były nadal uporządkowane nierosnąco (przykład: dla $M=2$ i sznurów $S1=7,6,6,5,4,4,4,2,2,1$ i $S2=8,8,6,6,6,4,4,4,2,2$ wynikiem będzie $S1=7,6,6,6,5,4,4,4,1$ i $S2=8,8,6,6,4,4,4,2,2,2$, dla $S1=5,5,5,4,3,3,3$ i $S2=7,7,7,6,1$ wynikiem będzie $S1=7,7,7,6,5,5,5,4,3,3,3,1$ oraz pusty sznur S2). Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zestaw C-24:

Organizacja plików: Wszystkie pliki oddawane do sprawdzenia należy zapisać we wspólnym folderze o nazwie będącej numerem indeksu, umieszczonym na pulpicie.

Oddajemy tylko źródła programów (pliki o rozszerzeniach .cpp i .h)!!!

Zad. 1: Napisać bibliotekę udostępniającą:

- typ Sznur będący dynamiczną listą łączoną przechowującą liczby całkowite;
- funkcję wypisz wypisującą zawartość sznura;
- funkcję wstaw wstawiającą do sznura element zawierający liczbę podaną jako parametr tak, aby sznur był uporządkowany nierosnąco (zakładając, że sznur „na wejściu” jest już uporządkowany w taki sposób);
- funkcję usunZDodatkiem rozpatrującą rozłączne pary elementów sznura i usuwającą całą parę jeśli zawiera równe wartości. Dodatkowo, jeżeli element następujący bezpośrednio po usuniętej parze zawiera tę samą wartość co usuwane, to też jest usuwany (usuwamy jeden dodatkowy element; w tym przypadku nie jest on uwzględniany w kolejnej parze). Pary wybierane są tak, że pierwszy element sznura jest początkowym elementem pierwszej pary. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp);

oraz program testujący działanie biblioteki poprzez pobranie sznura (funkcja wstaw; użytkownik podaje liczby, które mają być wstawione do sznura i decyduje o zakończeniu podawania), wypisanie jego zawartości (funkcja wypisz; przykład: dla zestawu liczb wejściowych 1,5,6,1,3,7,2,7,3,1,3,1 sznur uzyskuje postać 7,7,6,5,3,3,3,2,1,1,1,1), a następnie usunięcie w sznurze rozłącznych par identycznych elementów z ewentualnym dodatkowym trzecim (funkcja usunZDodatkiem) i ponowne wypisanie jego zawartości (przykład: sznur 7,7,6,5,3,3,3,2,1,1,1,1 zostanie przekształcony do postaci 6,5,2,1, sznur 9,9,9,6,6,6,5,5 stanie się pusty, sznur 5,5,5,5,4,4,4,2,1 zostanie przekształcony do postaci 5,4,2,1).

Zad. 2: Rozszerzyć bibliotekę z zad.1 o funkcję przeniesParami, która modyfikuje sznur poprzez przepięcie naprzemiennie na początek i koniec sznura par utworzonych z K elementów „na lewo od środka sznura” i K elementów „na prawo od środka sznura” (gdzie K jest liczbą całkowitą dodatnią będącą parametrem funkcji; jeśli elementów „na lewo od środka” i „na prawo od środka” jest mniej niż K funkcja używa ich tyle ile jest), w następujący sposób: para złożona z pierwszego elementu „lewej K-grupy” i pierwszego elementu „prawej K-grupy” jest wpinana na początku sznura, para złożona z drugiego elementu „lewej K-grupy” i drugiego elementu z „prawej K-grupy” jest wpinana na końcu sznura, itd. Przepinanie par ma zachować kolejność elementów względem tej w oryginalnym sznurze. Przykład: dla sznura przechowującego liczby 9,8,7,6,5,4,3,2,1 i K=3 dostaniemy wynik 8,4,6,2,9,5,1,7,3, dla K=8 powyższy sznur uzyska postać 9,4,7,2,5,8,3,6,1; dla sznura 7,5,4,3,2,1 i K=2 wynikiem będzie sznur postaci 5,3,7,1,4,2. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zad. 3: Rozszerzyć bibliotekę z zad.1 o funkcję wymienTrojki, modyfikującą dany sznur oraz drugi sznur S2 podany jako parametr, poprzez przepięcie między nimi co drugiej rozłącznej trójki sąsiednich elementów w następujący sposób: pierwsza pełna trójka elementów z „tego” sznura (oznaczonego w dalszym opisie przez S1) ma zostać przepięta w miejsce pierwszej pełnej trójki elementów sznura S2 i odwrotnie, analogicznie z elementami trzecich pełnych trójek w obu sznurach, piątych, siódmych, itd., przy czym elementy nadmiarowych trójek w sznurach nie są przenoszone (przykład: dla sznurów S1=7,6,6,5,4,4,4,2,2,1,1,1 i S2=8,8,6,6,6,4,4,3,3,2,2 wynikiem będzie S1=8,8,6,5,4,4,4,3,2,1,1,1 i S2=7,6,6,6,6,4,4,2,3,2,2 dla S1=5,5,4,4,3,3,3,2,1 i S2=7,7,7,6,1 wynikiem będzie S1=7,7,7,4,3,3,3,2,1 i S2=5,5,4,6,1). Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zestaw C-25:

Organizacja plików: Wszystkie pliki oddawane do sprawdzenia należy zapisać we wspólnym folderze o nazwie będącej numerem indeksu, umieszczonym na pulpicie.

Oddajemy tylko źródła programów (pliki o rozszerzeniach .cpp i .h)!!!

Zad. 1: Napisać bibliotekę udostępniającą:

- typ Sznur będący dynamiczną listą łączoną przechowującą liczby całkowite;
- funkcję wypisz wypisującą zawartość sznura;
- funkcję wstaw dodającą na końcu sznura element zawierający liczbę podaną jako parametr;
- funkcję usunCiagi usuwającą ze sznura ciągi sąsiednich elementów sznura w następujący sposób: omijamy K elementów sznura (lub tyle ile jest, gdy jest mniej), następnie usuwamy najdłuższy możliwy ciągły fragment zawierający same wartości nieparzyste (być może pusty), omijamy kolejnych K elementów, usuwamy najdłuższy możliwy ciągły fragment złożony z liczb nieparzystych, itd. (K jest liczbą całkowitą dodatnią będącą parametrem funkcji). Pamięć zajmowana przez usuwane elementy ma zostać zwolniona. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp);

oraz program testujący działanie biblioteki poprzez pobranie sznura (funkcja wstaw; użytkownik podaje liczby, które mają być wstawione do sznura i decyduje o zakończeniu podawania), wypisanie jego zawartości (funkcja wypisz; przykład: dla zestawu liczb wejściowych 4,7,5,1,6,6,1,7,3,7,2,9,1,1,1 sznur uzyskuje postać 4,7,5,1,6,6,1,7,3,7,2,9,1,1,1), a następnie po pobraniu od użytkownika wartości K, usunięcie w sznurze ciągów wartości nieparzystych z pominięciem K elementów przed kolejnymi usuwanymi ciągami (funkcja usunCiagi) i ponowne wypisanie jego zawartości (przykład: dla K=3 sznur 4,7,5,1,6,6,1,7,3,7,2,9,1,1,1 uzyska postać 4,7,5,6,6,1,2,9,1, dla K=2 sznur 9,9,9,4,4,4,4,5,4,7,7,1,1,2 zostanie zredukowany do 9,9,4,4,4,4,7,2, dla K=1 sznur 1,1,1,1,1 zostanie zredukowany do 1, dla K=3 sznur 1,2,3,4,5,6,8 pozostanie bez zmiany).

Zad. 2: Rozszerzyć bibliotekę z zad.1 o funkcję usunNadmiar, która modyfikuje sznur w następujący sposób: dla każdej wartości występującej w sznurze usuniętych ma zostać tyle końcowych elementów przechowujących tę wartość, aby zostało ich w sznurze nie więcej niż N (gdzie N jest liczbą całkowitą dodatnią będącą parametrem funkcji). Pamięć zajmowana przez usuwane elementy ma zostać zwolniona. Przykład: dla sznura 1,7,9,3,0,1,5,5,2,4,3,1,2,9,3,2,3 i N=2 dostaniemy wynik 1,7,9,3,0,1,5,5,2,4,3,2,9, dla N=5 powyższy sznur nie zmieni się, a dla N=1 uzyska postać 1,7,9,3,0,5,2,4. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zad. 3: Rozszerzyć bibliotekę z zad.1 o funkcję przeniesCoKty, modyfikującą dany sznur (oznaczony w dalszym opisie przez S1) oraz drugi sznur podany jako parametr (oznaczony dalej przez S2). Modyfikacja ma być wykonywana w następujący sposób: co K-ty element sznura S1, zaczynając od K-tego, ma zostać przepięty do końcowej części sznura S2 złożonej z M (lub tylu ile jest gdy jest mniej) elementów (gdzie K i M są liczbami całkowitymi dodatnimi będącymi parametrami funkcji), przy czym pierwszy z przepinanych elementów sznura S1 ma być wpięty za ostatnim elementem „M-końcówki” S2 (czyli za ostatnim elementem S2), drugi przepinany element S1 ma być wpięty przed ostatnim elementem oryginalnej „M-końcówki”, trzeci – przed jej przedostatnim elementem itd.; a M+1-szy i kolejne przepinane elementy są wpinane przed pierwszym z elementów „M-końcówki” S2 z odwróceniem kolejności względem tej jaką miały w S1 (przykłady: dla sznurów S1=1,2,3,4,5,6,7,8,9,0,7, S2=8,2,4,6,1,5,3,2,4, K=2 i M=3 wynikiem będą S1=1,3,5,7,9,7, S2=8,2,4,6,1,5,0,8,3,6,2,4,4,2, dla sznurów S1=1,3,2, S2=2,2,1,2,1, K=1 i M=1 wynikiem będą pusty sznur S1 i S2= 2,2,1,2,2,3,1,1, dla sznurów S1=1,2,3,4,5,6,7, S2=9, K=2 i M=4 wynikiem będą S1=1,3,5,7 i S2=6,4,9,2). Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zestaw C-26:

Organizacja plików: Wszystkie pliki oddawane do sprawdzenia należy zapisać we wspólnym folderze o nazwie będącej numerem indeksu, umieszczonym na pulpicie.

Oddajemy tylko źródła programów (pliki o rozszerzeniach .cpp i .h)!!!

Zad. 1: Napisać bibliotekę udostępniającą:

- typ Sznur będący dynamiczną listą łączoną przechowującą liczby całkowite;
- funkcję wypisz wypisującą zawartość sznura;
- funkcję wstaw dodającą na końcu sznura element zawierający liczbę podaną jako parametr;
- funkcję usunStaleCiagi usuwającą ze sznura ciągi sąsiednich elementów sznura w następujący sposób: omijamy M elementów sznura (lub tyle ile jest, gdy jest mniej), następnie usuwamy najdłuższy możliwy i równocześnie mający co najmniej M elementów ciągły fragment zawierający takie same wartości (jeśli stały fragment ma mniej niż M elementów to jest omijany), omijamy kolejnych M elementów, usuwamy najdłuższy możliwy ciągły stały fragment o długości co najmniej M, itd. (M jest liczbą całkowitą dodatnią będącą parametrem funkcji). Pamięć zajmowana przez usuwane elementy ma zostać zwolniona. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp);

oraz program testujący działanie biblioteki poprzez pobranie sznura (funkcja wstaw; użytkownik podaje liczby, które mają być wstawione do sznura i decyduje o zakończeniu podawania), wypisanie jego zawartości (funkcja wypisz; przykład: dla zestawu liczb wejściowych 4,7,5,1,6,6,1,7,7,7,2,9,1,1,1,1,1 sznur uzyskuje postać 4,7,5,1,6,6,1,7,7,7,2,9,1,1,1,1,1), a następnie po pobraniu od użytkownika wartości M, usunięcie w sznurze stałych ciągów o długości co najmniej M z pominięciem M elementów przed kolejnymi usuwanymi ciągami (funkcja usunStaleCiagi) i ponowne wypisanie jego zawartości (przykład: dla M=3 sznur 4,7,5,1,6,6,1,7,7,7,2,9,1,1,1,1,1 uzyska postać 4,7,5,1,6,6,1,2,9,1, dla M=2 sznur 9,9,9,4,4,4,4,4,4,5,4,7,2,1,1,3 zostanie zredukowany do 9,9,9,4,4,5,4,2,1,1,3, dla M=1 sznur 1,1,1,1,1 zostanie zredukowany do 1).

Zad. 2: Rozszerzyć bibliotekę z zad.1 o funkcję usunNadmiarowe, która modyfikuje sznur w następujący sposób: dla każdej wartości występującej w sznurze usuniętych ma zostać tyle początkowych elementów przechowywujących tę wartość, aby nie zostało ich w sznurze więcej niż N (będącej parametrem funkcji). Pamięć zajmowana przez usuwane elementy ma zostać zwolniona. Przykład: dla sznura 1,7,9,3,0,1,5,5,5,2,4,3,1,2,9,3,2,3 i N=2 dostaniemy wynik 7,9,0,1,5,5,4,1,2,9,3,2,3, dla N=5 powyższy sznur nie zmieni się, a dla N=1 uzyska postać 7,0,5,4,1,9,2,3. Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.

Zad. 3: Rozszerzyć bibliotekę z zad.1 o funkcję przeniesCoKty, modyfikującą dany sznur (oznaczony w dalszym opisie przez S1) oraz drugi sznur podany jako parametr (oznaczony dalej przez S2). Modyfikacja ma być wykonywana w następujący sposób: co K-ty element sznura S1, zaczynając od K-tego, ma zostać przepięty do końcowej części sznura S2 złożonej z M (lub tylu ilu jest gdy jest mniej) elementów (gdzie K i M są liczbami całkowitymi dodatnimi będącymi parametrami funkcji), przy czym pierwszy z przepinanych elementów sznura S1 ma być wpięty za ostatnim elementem „M-końcówki” S2 (czyli za ostatnim elementem S2), drugi przepinany element S1 ma być wpięty przed ostatnim elementem oryginalnej „M-końcówki”, trzeci – przed jej przedostatnim elementem itd.; a M+1-szy i kolejne przepinane elementy są wpinane przed pierwszym z elementów „M-końcówki” S2 z odwróceniem kolejności względem tej jaką miały w S1 (przykłady: dla sznurów S1=1,2,3,4,5,6,7,8,9,0,7, S2=8,2,4,6,1,5,3,2,4, K=2 i M=3 wynikiem będą S1=1,3,5,7,9,7, S2=8,2,4,6,1,5,0,8,3,6,2,4,4,2, dla sznurów S1=1,3,2, S2=2,2,1,2,1, K=1 i M=1 wynikiem będą pusty sznur S1 i S2= 2,2,1,2,2,3,1,1, dla sznurów S1=1,2,3,4,5,6,7, S2=9, K=2 i M=4 wynikiem będą S1=1,3,5,7 i S2=6,4,9,2). Funkcja nie może używać tablic, pomocniczych sznurów ani struktur danych dostępnych w bibliotece standardowej (wektorów, list, kolejek, itp). Wykorzystać funkcję w programie testującym.