

Napišite sučelje i implementaciju za strukture `pjesak` i `ulica`.

Struktura `pjesak` predstavlja pješaka. Svaki pješak ima naziv (`string`), vrijeme ulaska i izlaska s ulice (zadani kao `sat` i `minuta`, oba `int`) te mjesto ulaska i izlaska s ulice (zadani kao broj kilometara od početka ulice (`int`)).

Struktura `ulica` predstavlja ulicu koja sadrži kolekciju podataka tipa `pjesak`. Ukupan broj pješaka na ulici nikad neće biti veći od 1000.

Sučelje (deklaraciju) za obje strukture spremite u datoteku `ulica.h`, a implementaciju u datoteku `ulica.cpp`.

Strukture moraju imati sljedeće elemente (smijete dodati i elemente po želji, a neke ćete i morati dodati):

Struktura i funkcije	Opis
pjesak	
<code>pjesak(string naziv, int h1, int m1, int x1, int h2, int m2, int x2)</code>	Stvara pješaka (konstruktor) koje je na ulicu ušao u <code>h1</code> sati i <code>m1</code> minuta na kilometru <code>x1</code> , a izašao u <code>h2</code> sati i <code>m2</code> minuta na kilometru <code>x2</code> . Možete pretpostaviti da je prvo vrijeme manje od drugog i da je <code>x1</code> manji od <code>x2</code> .
<code>string getNaziv()</code>	Vraća naziv pješaka.
<code>float prosjecnaBrzina()</code>	Vraća prosječnu brzinu pješaka tijekom hoda po ulici u km/h.
ulica	
<code>ulica(int d)</code>	Stvara ulicu duljine <code>d</code> koja u sebi nema pješaka.
<code>int brojPjesaka()</code>	Vraća broj pješaka.
<code>pjesak getPjesak(int index)</code>	<code>index</code> je broj između 0 i brojaPješaka-1. Vraća određenog pješaka prema redoslijedu dodavanja u kolekciju.
<code>bool dodajPjesaka(pjesak p)</code>	Dodaje pješaka <code>p</code> kolekciji, ukoliko u kolekciji ima manje od 1000 pješaka i ako je mjesto ulaska i izlaska pješaka unutar granica ulice i vraća <code>true</code> , a <code>false</code> inače (pogledajte primjer dolje).
<code>bool ukloniPjesaka(string naz)</code>	Uklanja pješaka naziva <code>naz</code> , ako postoji u kolekciji, i vraća <code>true</code> , a <code>false</code> inače. Možete pretpostaviti da su nazivi svih pješaka na ulici međusobno različiti.
<code>ulica vrijeme(int h, int m)</code>	Vraća ulicu koja počinje i završava gdje i početna ulica i koja se sastoji samo od pješaka koji su se u vrijeme <code>h:m</code> nalazili na ulici. Pješaci trebaju biti u onom redoslijedu u kojem su bili u početnoj ulici.

<pre>int promijeniUlicu(int x1, int x2)</pre>	<p>Mijenja duljinu ulice tako da počinje na kilometru x_1, a završava na kilometru x_2. Također treba ukloniti sve pješake kojima je mjesto ulaska ili izlaska s ulice izvan nove dionice. Funkcija treba vratiti broj uklonjenih pješaka.</p>
-----------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Primjer klijentskog programa

```
#include "ulica.h"
#include <iostream>
#include <iomanip>

using namespace std;

void ispisiPjesake(ulica u) {
    for (int i = 0; i < u.brojPjesaka(); ++i) {
        pjesak p = u.getPjesak(i);
        cout << p.getNaziv() << ": ";
        cout << fixed << setprecision(2) << p.prosjecnaBrzina() << "
km/h" << endl;
    }
}

int main() {
    ulica U1(123);
    cout << U1.dodajPjesaka(pjesak("Marko", 12,0,0,13,0,14)); // 1
    cout << U1.dodajPjesaka(pjesak("Pero", 9,56,50,14,10,123)); // 1
    cout << U1.dodajPjesaka(pjesak("Ivan", 15,10,31,17,1,68)); // 1
    cout << U1.dodajPjesaka(pjesak("Filip", 8,0,45,10,1,124)); // 0 - izvan
granica ulice
    cout << U1.dodajPjesaka(pjesak("Luka", 9,5,50,12,55,121)); // 1
    cout << endl;

    ispisiPjesake(U1);
    /*
    Marko: 14.00 km/h
    Pero: 17.24 km/h
    Ivan: 20.00 km/h
    Luka: 18.52 km/h
    */

    cout << U1.brojPjesaka() << endl; // 4
    U1.ukloniPjesaka("Luka"); // postoji u kolekciji
    U1.ukloniPjesaka("Oliver"); // ne postoji u kolekciji
    cout << U1.brojPjesaka() << endl; // 3

    ispisiPjesake(U1.vrijeme(12, 0));
    /*
    Marko: 14.00 km/h
    Pero: 17.24 km/h
    */

    ispisiPjesake(U1.vrijeme(7, 59)); // nista
```

```

cout << U1.promijeniUlicu(50, 123) << endl; // 2
ispisiPjesake(U1);

/*
Pero: 17.24 km/h
*/

cout << U1.promijeniUlicu(51, 65) << endl; // 1
ispisiPjesake(U1);

cout << U1.promijeniUlicu(20, 123) << endl; // 0
ispisiPjesake(U1);

cout << U1.dodajPjesaka(pjesak("Zvonko", 15,25,19,16,55,100)); // 0
cout << U1.dodajPjesaka(pjesak("Davor", 14,45,51,17,34,122)); // 1
cout << endl;
ispisiPjesake(U1);
/*
Davor: 25.21 km/h
*/

return 0;
}

```

Opće napomene

- Struktura, funkcije i datoteke koje šalžete moraju se zvati *točno* onako kako je zadano u zadatku. Pazite na mala i velika slova!
- Trebate poslati samo sučelje i implementaciju. U datotekama koje šalžete *ne smije* se nalaziti funkcija `main()`!
- Nijedna funkcija *ne smije* ništa učitavati s tipkovnice ili neke datoteke, niti išta ispisivati na ekran ili u neku datoteku.
- Svaki od `main`-ova pomoću kojih testiramo ispravnost vašeg programa neće pozivati sve gore navedene funkcije. Stoga, ako neku od funkcija ne znate napisati ipak možete dobiti koji bod (u tom slučaju tu funkciju nemojte navesti niti u `.h` niti u `.cpp` datoteci ili napravite neku trivijalnu implementaciju).

Ispravnost implementacijâ koje napišete bit će provjerena tako da ćemo mi napisati razne klijentske programe koji će deklarirati nekoliko varijabli zadane strukture, i na njima pozivati funkcije koje ste trebali napisati. Ako se poslani programi ne budu uspješno povezivali (*linkali*) s našim klijentskim programima, smatrat će se neispravnima. Neki klijentski programi provjeravat će samo neke jednostavnije funkcije, dok će neki provjeravati sve funkcije koje trebate napisati. Provjera je potpuno automatska, tako da je od presudne važnosti da se pridržavate specifikacije. Nepridržavanje lako može uzrokovati osvojenih 0 bodova iz zadaće! Naravno, za provjeru radi li implementacija prije nego što je pošaljete, preporučuje se da je testirate pomoću nekog klijentskog programa. No taj klijentski program ne šalžete!