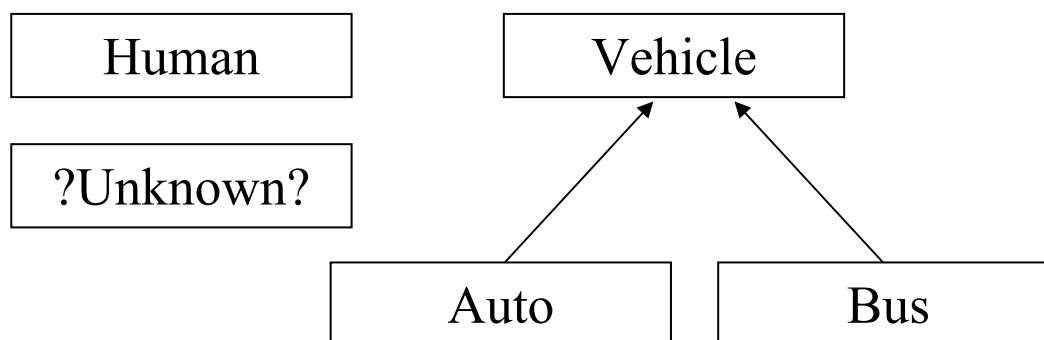


10. laboratorijas darbs

Ir novērotājs, kas kontrolē kustību ielā un reģistrē četru klašu objektus: Human, Vehicle, Auto un Bus. Visus citus objektus viņš interpretē kā *nezināmus* objektus.



Attiecīgās klases deklarē *bez* atribūtiem un metodēm. Objektu veidošanai programmā izmanto *konstruktorus pēc noklusēšanas*.

Objektu masīvs atrodas programmas galvenajā klasē, `main(...)` metodē:

```
Object [] Street = {new Human(), ...};
```

Objektu testēšanai izmanto trīs metodes no galvenās klases. Visas trīs metodes tiks izsauktas no `main(...)` metodes. Objektu masīvu nodod kā parametru:

1. **public static void** check1(Object [] O) {
 int V=0, A=0, B=0, H=0, U=0; // skaitītāji
2. **public static void** check2(Object [] O) {
 int V=0, A=0, B=0, H=0, U=0; // skaitītāji
3. **public static void** check3(Object [] O) {
 HashMap<String, Integer> Result =
 new HashMap<String, Integer>(); // vārdnīca

Piezīme: trešajā gadījumā precīzi klasificē *visus* objektus (variants Unknown nav pieņemams). Objekta klases vārds ir *vārdnīcas atslēga*.

Izpildīt objektu klasifikāciju, izmantojot trīs dažādus RTTI paņēmienus. Pēdējā gadījumā izveidot noskaņojamo *rezultātu vārdnīcu*. Šajā gadījumā ir tikai viens **if...else** operators.

Iespējamie rezultāti:

Instance of ...

Vehicle: 1, Auto:1, Bus:2, Human: 3, Unknown object: 1.

Is instance ...

Vehicle: 1, Auto:1, Bus:2, Human: 3, Unknown object: 1.

HashMap + Get class ...

```
{Bus=2, java.lang.Integer=1, Auto=1, Vehicle=1, Human=3}
```

Piezīme: sākumā vārdnīca ir *tukša*.

Vārdnīcas atslēgas eksistēšanas pārbaude:

```
if (Result.containsKey(Key))
```