**Korišteni uzorci dizajna :**

**Builder** – korišten kako bi odvojio opcionalne i potrebne parametre, te kako prevelik broj argumenata u konstruktoru ne bi doveo do krivog niza argumenata. Također, pri čitanju datoteke *DZ\_1\_uroni.txt* kreiram „početno“ stanje objekta, a kasnije kreiram konačno stanje objekta (sa sastavom ronioca za svaki uron). Koristi se i za kreiranje Ronioc objekta.

**Factory Method** – korišten za agencije (ne znamo unaprijed kojoj agenciji pripada ronioc te koju agenciju treba obavijestiti), institucije (isto kao i za agencije), te algoritme (koje također ne znamo unaprijed, dobijemo ih kao parametre prilikom pokretanja).

**Observer –** korišten za obavještavanje ronioca o planu ronjenja. Također, korišten je za obavještavanje pripadajuće agencije i institucije (ovisno o svakom roniocu).

**Visitor –** korišten kako bi dodali novu funkcionalost za klase „Ronioc“ i „Uron“, s tim da ne mijenjamo postojeće klase „Ronioc“ i „Uron“ – osim implementiranja Visitable sučelja.

**Singleton –** korišten je kako bi imali jednu instancu ronilačkog kluba, u kojeg sam spremao listu ronioca i listu urona (iz odgovarajućih datoteka). Također sam spremio i objekt Random klase (te postavio sjeme), kako bi ga mogao koristiti i izvan *main* metode.

**Promjene u odnosu na 1. zadaću:**

* Svi algoritmi bi trebali raditi, te sada koriste Factory Method umjesto Abstract Factory-a
* Manje izmjene u kodu

**Opis vlastite funkcionalnosti :**

Implementacija Visitor uzorka dizajna, koja dodaje nove funkcionalnosti vezane za klase „Ronioc“ i „Uron“. Dodane funkcionalnosti:

* Računanje prosječnog broja godina (za one ronioce koji rone)
* Prikaz broja ronioca prema rasponu godina (20-29, 30-39, itd)
* Računanje prosječnog broja ronioca u uronima

