Schronisko

1. Klasa Schronisko  
   - Schronisko może posiadać 25 psów i 15 kotów. Dynamiczna alokacja  
   - Posiada metodę zwierzę\* schronisko::oddaj\_zwierzę(). Tu wykorzystam template

- Posiada metodę void schronisko::weź\_zwierzę(zwierzę \* \_oddane) Tu też  
- Decyduje o możliwości adopcji – tworzy listę dostępnych zwierząt vector<zwierzę>

- Usypianie zwierząt, których nikt nie wziął 5 razy.

1. Klasa zwierzę

- Ma świadomość o tym kto jest jego właścicielem.  
- Atrybuty: poziom\_agresji, rasa.  
- Metody zwracające wartości atrybutów.  
- Metoda możliwość ataku Tu polimorfizm z ataku na gryzienie lub drapanie

1. Klasa pies  
   - Dziedziczy po klasie zwierzę.  
   - Dodatkowe atrybuty: wielkość, czy\_siersc\_uczula.  
   - Dodatkowa metoda: możliwość ugryzienia.

- Metody zwracające wartości dodatkowych atrybutów.

1. Klasa kot  
   - Dziedziczy po klasie zwierzę.  
   - Dodatkowe atrybuty: czy\_to\_kot\_domowy.  
   - Dodatkowa metoda: możliwość podrapania.

- Metody zwracające wartości dodatkowych atrybutów.

1. Klasa klient  
   - Klient może posiadać przy sobie 1 zwierzę – do starania się o adopcje lub oddania jeśli jest miejsce w schronisku.  
   - Atrybuty: czy\_chce\_adoptowac, inne\_zwierzęta, wielkość\_mieszkania, czy\_szuka konkretnej rasy, uczulenie.  
   - Posiada metody zwracające wartości atrybutów.  
   - Posiada metodą wypróbuj.  
   - Posiada metodę void klient:weź\_zwierzę(zwierzę\* \_adoptowane)

- Posiada metodę zwierzę\* klient::oddaj\_zwierzę()  
  
  
Z pliku odczytywane będą psy, koty i klienci. Stringi, streamstreamy

Do pliku dopisywane będą kolejno oddawane zwierzęta i nowi właściciele.