

Q1

```
#ifndef HUMAN_HPP
#define HUMAN_HPP

class Human{};

#endif
```

Q2

```
#include "animal.hpp"
#include "human.hpp"

class Human;

> void Animal::play() { ...
> void Animal::eat() { ...
> void Cat::play() { ...
> void Cat::meow() { ...
> void Dog::play() { ...
> void Dog::bark() { ...
> void Dog::eat() { ...

void Dog::play_with_human(Human &h) {
    std::cout << "Le chien joue avec un humain" << std::endl;
}
```

```
class Dog : public Animal {
public:
    void play();
    void bark();
    void eat();
    void play_with_human(Human &h);
};
```

Q3

```
int main(){

    Cat c; Dog d;
    c.play(); d.play();
    c.meow(); d.bark();
    c.eat(); d.eat();

    Human my_human;
    d.play_with_human(my_human);
    return 0;
}
```

```
root@Galaxy-Book-Noam:~/Desk
Le chat joue
Le chien joue
Le chat miaule
Le chien aboie
L'animal mange
L'animal mange
Le chien mange salement
Le chien joue avec un humain
```

Q4

```
class Animal {
public:
    void play();
    void eat();
    void play_with_human(Human &h);
};

class Cat : public Animal {
public:
    void play();
    void meow();
    void play_with_human(Human &h);
};
```

```
> void Animal::play_with_human(Human &h) { ...
> void Cat::play() { ...
> void Cat::meow() { ...
> void Cat::play_with_human(Human &h) { ...
```

```
6 int main(){
7
8     Cat c; Dog d;
9     c.play(); d.play();
10    c.meow(); d.bark();
11    c.eat(); d.eat();
12
13    Human my_human;
14    c.play_with_human(my_human);
15    d.play_with_human(my_human);
16
17    return 0;
18 }
```

PROBLÈMES SORTIE CONSOLE DE DÉBOGAGE

● root@Galaxy-Book-Noam:~/Desktop/P00\_P  
Le chat joue  
Le chien joue  
Le chat miaule  
Le chien aboie  
L'animal mange  
L'animal mange  
Le chien mange salement  
Les chats sont autonomiaous  
Le chien joue avec un humain

Q5

```
✓ void animal_day(Animal &animal, Human &human) {
    animal.eat();
    animal.play();
    animal.eat();
    animal.play_with_human(human);
}
```

```
int main(){
    Cat c; Dog d;
    Human my_human;

    std::cout << "Journée typique pour le chat :" << std::endl;
    animal_day(c, my_human);

    std::cout << "\nJournée typique pour le chien :" << std::endl;
    animal_day(d, my_human);

    return 0;
}
```

Q6

```
root@Galaxy-Book-Noam:~/Desktop/
Journée typique pour le chat :
L'animal mange
L'animal joue
L'animal mange
L'animal joue avec un humain

Journée typique pour le chien :
L'animal mange
L'animal joue
L'animal mange
L'animal joue avec un humain
```

Cela affiche animal au lieu de chat et chiens, les méthode sont statiques et non virtuels

Q7

```
class Animal {
public:
    virtual void play();
    virtual void eat();
    virtual void play_with_human(Human &h);
};
```

```
root@Galaxy-Book-Noam:~/Desktop/
Journée typique pour le chat :
L'animal mange
Le chat joue
L'animal mange
Les chats sont autonomes

Journée typique pour le chien :
L'animal mange
Le chien mange salement
Le chien joue
L'animal mange
Le chien mange salement
Le chien joue avec un humain
```

Q8

```
Journée typique pour le chat :
L'animal mange
Le chat joue
L'animal mange
L'animal joue avec un humain
```

Si on enlève la méthode play\_with\_human de cat, la méthode play\_with\_human d'animal est appelé à la place