

Q1

```
1  #ifndef ANIMALS_HPP
2  #define ANIMALS_HPP
3
4  #include <iostream>
5
6  class Animal {
7  public:
8      void play();
9  };
10
11  class Cat : public Animal {
12  public:
13      void play();
14      void meow();
15  };
16
17  class Dog : public Animal {
18  public:
19      void play();
20      void bark();
21  };
22
23  #endif
```

```
● root@Galaxy-Book-Noam:~$ ./animals
Le chat joue
Le chien joue
Le chat miaule
Le chien aboie
```

J'ai bien mes 3 méthode play, la compilation fonctionne correctement

Q2

```
class Animal {
public:
    void play();
    void eat();
};

class Cat : public Animal {
public:
    void play();
    void meow();
};

class Dog : public Animal {
public:
    void play();
    void bark();
    void eat();
};
```

```
> void Animal::play() { ...
}

void Animal::eat() {
    std::cout << "L'animal mange" << std::endl;
}

> void Cat::play() { ...
}

> void Cat::meow() { ...
}

> void Dog::play() { ...
}

> void Dog::bark() { ...
}

void Dog::eat() {
    std::cout << "Le chien mange salement" << std::endl;
}
```

Q3

```
32 int main(){
33
34     Cat c; Dog d;
35     c.play(); d.play();
36     c.meow(); d.bark();
37     c.eat(); d.eat();
38     return 0;
39 }
```

PROBLÈMES SORTIE CONSOLE DE DÉBOGAGE

● root@Galaxy-Book-Noam:~/Desktop/P00_1
Le chat joue
Le chien joue
Le chat miaule
Le chien aboie
L'animal mange
Le chien mange salement

La méthode eat de l'instance cat affiche seulement "l'animal mange" ce qui signifie que cela appelle la méthode eat de la classe Animal si je ne surcharge pas la méthode de Animal

Q4

```
27
28 void Dog::eat() {
29     Animal::eat();
30     std::cout << "Le chien mange salement" << std::endl;
31 }
```

PROBLÈMES SORTIE CONSOLE DE DÉBOGAGE TERMINAL PORTS

Le chat joue
Le chien joue
Le chat miaule
Le chien aboie
L'animal mange
L'animal mange
Le chien mange salement

On voit bien que sans changer le main, la méthode animal::eat() est appelée car "L'animal mange" est appelé 1x pour cat mais aussi une fois pour dog en plus d'afficher le message précédent

