Concept Map Address this before (5 minutes and you don’t even need to go over their answers), and if you have time after answering the questions below, then revisit your concept map to make any changes based on the new information you learned!

As a class, discuss the following:
Given the following code, discuss the following and write code to prove your answers.

```cpp
struct card {
    int rank;
    string suit;
};
class cardgame {
    protected:
        card *deck;
    int num_cards;
    public:
        cardgame();
        ~cardgame();
};
class gofish : public cardgame {
    private:
        static const int max_players=4;
    public:
        gofish();
        ~gofish();
};
```
If you create a child object, i.e. gofish g;

- What is inherited and not inherited?

- What is accessible and not accessible?

- When is the cardgame constructor and gofish constructor called, and what is the order?

- When is the cardgame destructor and gofish destructor called, and what is the order?

- What is upcasting and downcasting? Which is allowed without an explicit typecast and what does it do?

- What if you have many cardgames and do not know which game the user will want to play, e.g. solitaire, blackjack, rummy? How will you handle this?

- What keyword do you need on the destructor and why?