CPP07

* Templates :

↳ can define template fct°, classes, struct, etc → code patron

```
template < typename T >                    instancier     max<int>(a, b) << std::endl; → Explici
~~int~~ T const& max (const T &x, const T &y) {   →        max (a, b) ...  → Implicit
    return ( x >= y ? x : y );
}                                                                  .tpp
```

* Default Type :

↳ allow to pass default values to type variables in template

template < typename T = float > → if not specified type, use float (~~inputed in param~~ or empty <>)

< >

complete → < bool, bool >

no specializ ~~complete~~ → vs < T, U >

partial → < T, int >

* Specialization :

↳ partial or complete specialization (new syntax) by leaving other type like int

↳ call a specific template if type matches instead of generic template