## **Theory Assignment 2 - Entity Relationship Diagram (UML Notation)**

Andressa Pessoa de A. Machado | April 9, 2020 Main + bool RUNNING = true; + char user input; Bank + int account index; + double deposit amount; + double withdraw amount; - max number of accounts = 16 + double balance total; - int number\_of\_accounts - forward list<Account\*> accounts + Bank bank: Account\* chequing1; + Bank(){}; Account\* chequing2; + ~Bank(){}; Account\* chequing3; + void add an account(Account\*): Account\* saving1; + int get num of accounts(); Account\* saving2; + Account\* get account(int); Account\* saving3; + void monthly operation(); + int main(void) {} + void delete accounts(); + void print list of accounts(); Bank has n accounts Account - int account number # double balance + Account(int account number, double balance): account\_number(account\_number), balance(balance){}; + virtual ~Account() {}; + virtual void monthly\_operation() = 0; + int get account number(); + double get balance(); + virtual double withdraw(double) = 0; + double deposit(double); ChequingAccount ChequingAccount - const double additional\_withdraw\_fee = 2.0; - const double interest = 0.5 / 100; - const int withdraw max = 4; - int withdraw counter = 0; + SavingAccount(int account number, double balance): Account(account number, balance) {}; + ChequingAccount(int account number, double balance) : Account(account number, balance) {}; + ~SavingAccount() {}; + ~ChequingAccount() {}; + double withdraw(double); + void monthly operation(); + void monthly operation(); + double withdraw(double);