

Uppgiftsbeskrivning av kompetenskontroll, moment 1;

Bankomat med signup och signin funktioner + interaktion med saldo

Uppgiftsbeskrivning;

Ni skall koda ett program som agerar bankomat.

I denna bankomat skall användaren kunna sätta in samt ta ut pengar och även se sitt saldo. Saldot skall även alltid uppvisas efter en handling som har förändrat det, exempelvis uttag eller insättning. För att komma åt denna information måste personen ange sitt användarnamn samt sitt lösenord.

Om personen inte har ett konto måste personen istället välja alternativet “Create New Account” där användaren då får ange användarnamn samt lösenord och då mera blir registrerad användare. Efter detta skall användaren nu kunna logga in och nyttja nämnda funktioner ovan.

Till en början byggs detta med runtime memory så att användarens saldo bara sparas under användningen, men vi kommer sedan bygga vidare med exempelvis databasfunktion för lagring och flertalet andra funktioner.

Moment 1;

I skapandet av programmets första del kommer vi nyttja en enda .cpp fil för att hålla koden. Vi kommer att nyttja structs, strings, functions, vectors och user input för att uppnå funktionerna som nämndes ovan. I stort sett allt vi vidrört hittills. Ni får även ett exempel på grundstrukturen på nästkommande sida.

Detta är delmoment 1 som sedan kommer utökas med funktionaliteter såsom lokal databas, mer grafiskt bemötande och liknande.

Syftet med detta är att förbereda er inför kursens slutuppgift.

Försatt struktur;

För att underlätta er struktur i detta arbete men även underlätta för oss utbildare att kunna vara ett bra stöd i vidareutvecklingen och rättningen så tilldelas ni nedan försatta namn på funktionerna ni skall nyttja;

```
#include <iostream>
#include <iomanip>
#include <string>
#include <vector>
using namespace std;

struct AutoTellerMachine
// to represent each customer who uses the ATM program
{
    void CreateNewAccount(string newUsername, string newPassword);
    void AccountLogin(string loginUsername, string loginPassword);
    void DepositMoney(double depositAmount);
    void WithdrawMoney(double withdrawalAmount);
    void SetAccountLogin(int setAccountLocation);
    void SetLastMoneyMovement(int accountID, double amount);
    void SetBeginningBalance(int accountID);
    void SetLastOperation(int accountID, char userInput);
    void AccountMenu();
    int GetAccountLogin() const;
    double GetLastMoneyMovement(int accountID) const;
    double GetAccountBalance(int accountID) const;
    double GetBeginningBalance(int accountID) const;
    char GetLastOperation(int accountID) const;
    string GetUsername(int accountID) const;

    int loggedInAccountLocation;
    double accountBalance;
    double beginningBalance;
    double lastMoneyMovement;
    char lastOperation;
    string username;
    string password;
};
```