Exercise 6: Account Balance

In this exercise, you will be required to use pointers and pointer to functions to update the balance in an account

We'll cover the following

Part 1

Pointers to Update Account Balance

Problem Statement

In this part you are required to:

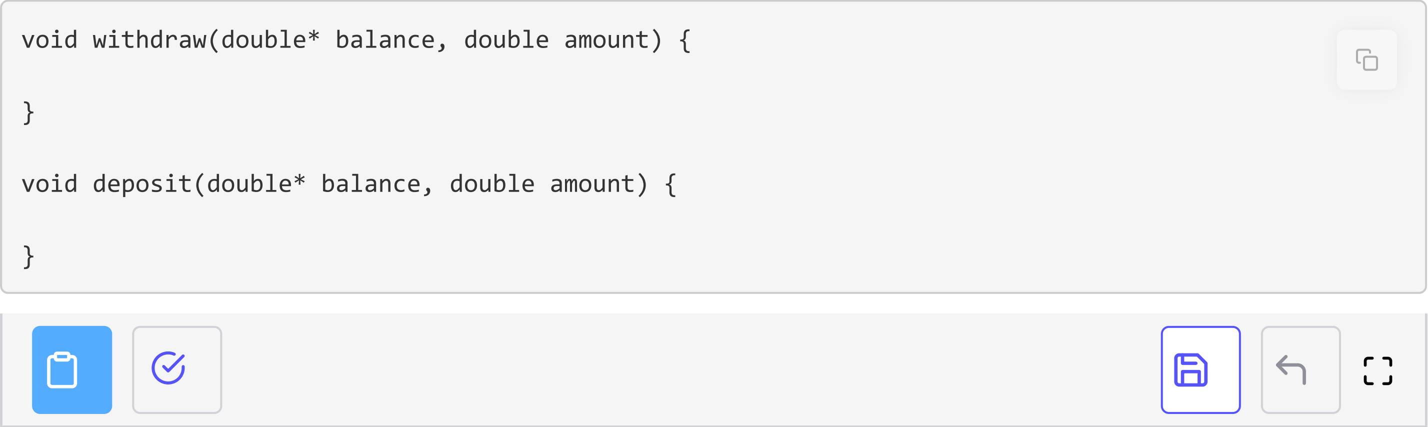
- Make functions
  - **Withdraw**:
    - Is a `void` function.
    - Takes **two** *parameters*:
      - a *pointer* to the *balance* in *account*.
      - the *amount* to be *withdrawn* from account, also of *type* `double`.
    - Add a **check** so that the *amount* is only *withdrawn* if it is **less** then **total** *balance* in the *account*. Otherwise it should *display*: **"You are broke. You don’t have enough money. Go away"**
  - **Deposit**:
    - Takes **two** *parameters*:
      - a *pointer* to the *balance* in *account*.
      - the *amount* to be *deposited* in the account, also of *type* `double`.

the *amount* to be *deposited* in the account, also of *type* `double`.

- The function should **add** the required *amount* into the account.

Write your code below. It is recommended that you try solving the exercise yourself before viewing the solution.

Good Luck!



```
void withdraw(double* balance, double amount) {

}

void deposit(double* balance, double amount) {

}
```

Part 2

Pointers To Functions

Problem Statement

Now let’s implement the above example but by using **Pointers to Functions**.

In this part you are required to:

- Declare a **pointer** to functions
  - **Withdraw**:
  - **Deposit**:

> **Note:** The code for both `Withdraw` and `Deposit` functions have already been prepended (not visible to you) in the code widget below hence you don’t need to rewrite them as you’ve done that above already.

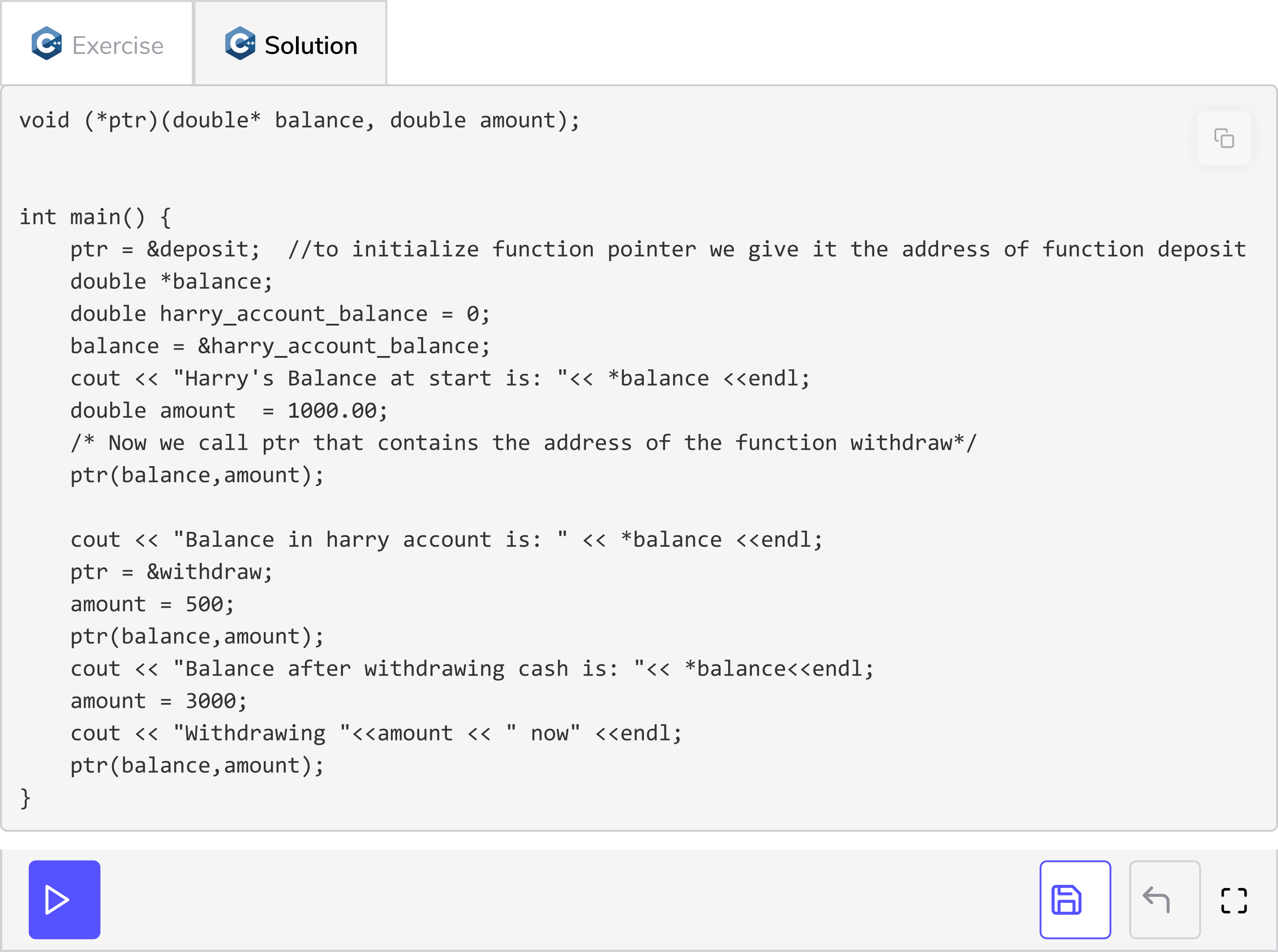
- *Call* both these *functions* in the `main` according to the *instructions* given in the code widget below.

Hint

- *Declare* and *use* a **pointer** named `balance` in your code below to update and keep track of the **total** *balance* in the *account*.

Write your code below. It is recommended that you try solving the exercise yourself before viewing the solution.

Good Luck!



```
void (*ptr)(double* balance, double amount);


int main() {
    ptr = &deposit;  //to initialize function pointer we give it the address of function deposit
    double *balance;
    double harry_account_balance = 0;
    balance = &harry_account_balance;
    cout << "Harry's Balance at start is: "<< *balance <<endl;
    double amount  = 1000.00;
    /* Now we call ptr that contains the address of the function withdraw*/
    ptr(balance,amount);

    cout << "Balance in harry account is: " << *balance <<endl;
    ptr = &withdraw;
    amount = 500;
    ptr(balance,amount);
    cout << "Balance after withdrawing cash is: "<< *balance<<endl;
    amount = 3000;
    cout << "Withdrawing "<<amount << " now" <<endl;
    ptr(balance,amount);
}
```