```cpp
#include <iostream>
#include <string>
#include <fstream>
using namespace std;
//Defining an Account class
class Account{
        public:
//              setting salary value
                float salary = 2000.00;

};
//Using inheritance
class Engineer: public Account{
        public:
                float allowance = 200.00;

};
//using a struct
struct softwareEngineer{
//      defining pointers to be used later
        int *mobileDeveloperBonus;
        int *webDeveloperBonus;
        int *desktopDeveloperBonus;
//      Using an array of integers
        int recommendedAges[8] = {3, 9, 16, 28, 32, 50, 51, 52};
//      Using floats
        float mobileDeveloperTax = 60.04;
        float webDeveloperTax = 45.87;
        float destopDeveloperTax = 54.23;
};
//using a function that does not return a value
void printDeveloperBonus(int bonus, int* developerBonus){
        developerBonus = &bonus;
        cout << "The address stored in this pointer is: " << developerBonus  << endl;
        cout << "The value of this pointer is: " << *developerBonus  << endl;

}
//Using a function that returns a float value
float netSalary(float salary, float bonus, float allawance, float tax){
        float netSalary;
        netSalary = salary + bonus + allawance - tax;
        return netSalary;

}
```

```cpp
//using a function that does not return any value (Just printing the main menu)
void displayMenu(){
        cout << endl <<"-----------------Menu--------------------" << endl;
        cout<< endl << "1. Get the address and value stored in a pointer"<< endl;
        cout << "2. Check recommended ages of the book" << endl;
        cout << "3. Get the price of the book" << endl;
        cout << "4. Create a text file with employes names" << endl;
        cout << "5. Read text file" << endl;
        cout << "6. Exit" << endl;
}
//using a function that does not return any value (Just printing the Software Engineer menu)
void displaySoftwareEngineers(){
        cout << "-----------------Select a type of book--------------------" << endl;
        cout<< endl << "1. kid books"<< endl;
        cout << "2. adult books" << endl;
        cout << "3. Magazines" << endl;
}
//creating a data file that returns a boolen
bool createFile(){
//      using an array of strings
        string softwareEngineers[5] = {"Jane 601-789-2454", "John 258-943-2156", "Hellen
601-523-5202", "Lawrence 563-892-0921", "Matheo 789-253-4503"};
        fstream fw;
//      creating a text file
        fw.open("engineers.txt", ios::out);
        if(fw.is_open()){
//              using a loop to write data to a file
                for(int i = 0; i < 5; i++){
                        fw << softwareEngineers[i] << "\n";
                }
                fw.close();
                return true;
        }else{
                return false;
        }


}
//a function for reading data in a text file
void readFile(){
        string engineer;
        fstream textFile;
//      openning the text file
        textFile.open("engineers.txt", ios::in);
```

```cpp
                cout << "List of Employes Names and Phone Numbers." << endl;
//        Printing data from a text file
        if(textFile.is_open()){
                string data;
                int count = 1;
//                using a while loop to read and print datya from a text file
                while(getline(textFile, data)){
                        cout << count << ". "<< data << "\n";
                        count = count + 1;
                }
                textFile.close();
        }else{
                cout << "error";
        }
}
//The main function
int main() {
//        Defining a class object
        Engineer e1;
//        initializing an object of a struct
        struct softwareEngineer engineer;
//        defining float and int variables to be used later
        float salary;
        float allowance;
        float bonus;
        float tax;
        int mobileBonus = 180;
        int webBonus = 150;
        int desktopBonus = 190;
        int option;
        int engineerOption;

        salary = float(e1.salary);
        allowance = float(e1.allowance);



        menu:
//        Calling the function to display the main menu
        displayMenu();
        cout << "Select an option: ";
        cin >> option;
//        if option 1 (Get the address and value stored in a pointer) is selected
        if(option == 1){
```

```cpp
                displaySoftwareEngineers();
                cout << "Select an a book type: ";
                cin >> engineerOption;
                if(engineerOption == 1){
                        printDeveloperBonus(mobileBonus, engineer.mobileDeveloperBonus);
                }else if(engineerOption == 2){
                        printDeveloperBonus(webBonus, engineer.webDeveloperBonus);
                }else if(engineerOption == 3){
                        printDeveloperBonus(desktopBonus, engineer.desktopDeveloperBonus);
                }
                goto menu;


        }
//      if option 2 is selected from the main menu
        else if(option == 2){
                cout << "Recommended ages for types of books are: " ;
//              using a for loop to get data from an array
                for(int i = 0; i < 8; i++){
                        cout << engineer.recommendedAges[i] << ", ";
                }
                goto menu;


        }
//      if option 3 is selected from the main menu
        else if(option == 3){
                displaySoftwareEngineers();
                cout << "Select an book type: ";
                cin >> engineerOption;
                if(engineerOption == 1){
                        bonus = float(mobileBonus);
                        tax = engineer.mobileDeveloperTax;
                        cout << endl << "The net salary for this type of book are : $" << netSalary
(salary, bonus, allowance, tax)<< endl;

                }else if(engineerOption == 2){
                        bonus = float(webBonus);
                        tax = engineer.webDeveloperTax;
                        cout << endl << "The net salary for this type of book are : $" << netSalary
(salary, bonus, allowance, tax)<< endl;

                }else if(engineerOption == 3){
                        bonus = float(desktopBonus);
                        tax = engineer.destopDeveloperTax;
```

```cpp
                    cout << endl << "The net salary for this type of book are : $" << netSalary
(salary, bonus, allowance, tax)<< endl;

            }
            goto menu;
        }
//      if option 4 is selected from the main menu
        else if(option == 4){
//              if createFile returns true
                if(createFile()){
                        cout << "File created successfully!! " ;
                }
//              if it returns false
                else{
                        cout << "Error while creating a text file!! " ;
                }
                goto menu;

        }
//      if option 5 is selected from the main menu (read file content)
        else if(option == 5){
                readFile();
                goto menu;

        }
//      if option 6 is selected (exit the system)
        else if(option == 6){
                exit(0);

        }




}
```