

FCIS – Ain Shams University

Credit Hour Programs (CHP)

Final Assessment

Academic year: 2019/2020

Academic Semester: Spring 2020



**Credit Hour Programs**  
Faculty of Computer & Information Sciences

- Research thesis replaces the final written exam
- The research is individual
- Each part/component of the research has a weight
- Research submission is online in electronic format
- Due date will be the date announced by faculty/program administration
- Research should be submitted in announced Student Template
- Plagiarism checking will be applied. Research is subject to rejection in such case

## RESEARCH THESIS

<b>Program</b>	CHP
<b>Level</b>	1
<b>Course Code</b>	CIS150
<b>Course Title</b>	Structured Programming

<b>Instructor(s)</b>	Dr. Yasmine Afify Dr. Salsabil Amin
----------------------	--

<b>Research Title:</b>	Telecom Company
------------------------	-----------------

THIS PAGE IS THE DOCUMENT COVER PAGE  
YOU ARE **NOT ALLOWED** TO WRITE YOUR NAME, OR REVEAL YOUR IDENTITY  
THROUGH ANY OF THE DOUCMENT PAGES  
YOU ARE ONLY ALLOWED TO FILL IN THE GIVEN INFORMATION

CHP - Spring 2020

## 1. Data Model

1) struct client, data, Date: make this struct because collection of data of different types.

Use structure client to take data of users to enter app.

Use structure data to take data of every line from client.

Use structure Date to set start date and renewal date.

2) int user ID, int start date, int renewal date, int Day , int Month , int Year;

Because int stores whole numbers without decimal.

int transferred balance; local variable to declared inside function, available only within that function, int stores whole number without decimal

3)int line [3]; Because int stores whole numbers without decimal, used for integer data types is **int**, and take array because collection of the same type of data and 3 lines have same data type.

4)int value [5]: Because int stores integer number, array because array collection of the same type of data, this array collection five of number of balance in card to recharge balance

5) string email, string name, string address, String package type; because string is array of character.

6) char password; stores single Characters, letter, number, or ASCII value.

char ch; local variable to declared inside function, available only within that function, stores a single character, letter, number, or ASCII value , reason of calling ch to take password from client.

7) long int mobile number; long int take the numbers that consist of 11 digit.

8) client a, users [3]; global variable to declared outside all functions, hence defined for all functions.

9) Float balance; because float stores fraction number contain decimal.

10) long int mobile number; global variable to declared outside all functions, hence defined for all functions, long int take the numbers that consist of 11-digit.

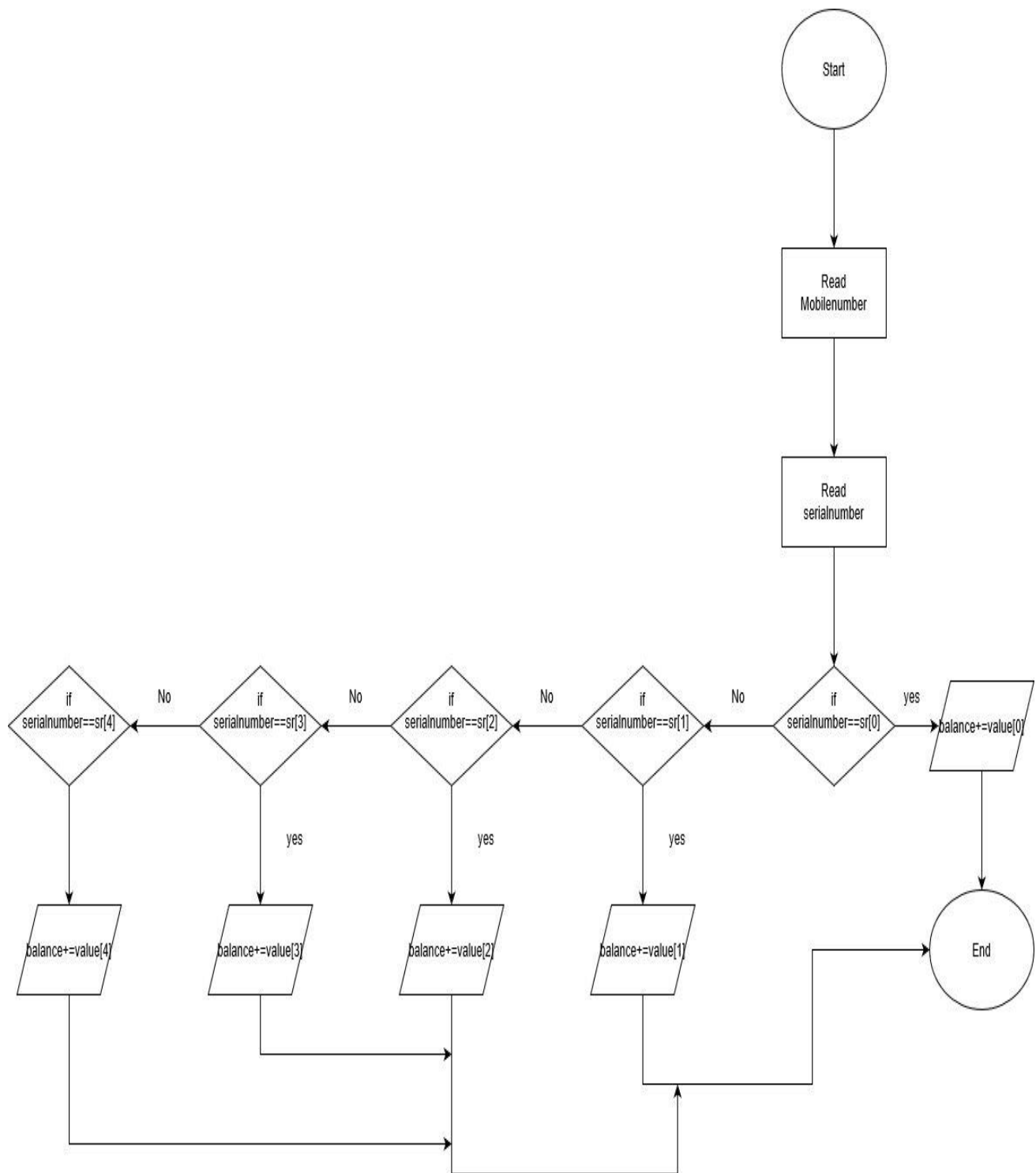
11)long int serial number, long int number transferred to the balance; local variable to declared inside function, available only within that function, long int take the numbers that consist of 11 digit.

12)long int sr [5]: long int take the numbers that consist of 11 digit , sr [5] (array) because array collection of the same type of data , this array collection five from serial number card to recharge balance

13) string name of new package type, string message; local variable to declared inside function, available only within that function, string because string is array of character.

14) bool is special, bool is num; local variable to declared inside function, available only within that function, bool because Boolean stores true or false value.

## 2. Logical Model (Algorithm):



### **3. Process Model (Functions)**

- 1)void change package type (string, long int) : use this function to change package type for client
- 2)void renew subscription (long int) : use this function to renew subscription “package type” for client
- 3)void make call (long int): use this function to make call another mobile number
- 4)void send message (string): use this function to send message another mobile number
- 5)void recharge balance (long int): use this function to recharge balance by entering a card serial number with a specific amount of money for client
- 6)void transfer balance (long int , int): use this function to transfer balance to another user on the same network.

### **4. Coding Style**

- 1) using meaning full name in line 30 replace var from x to mobile number.
- 2) using meaning full name in line 33 replace var from y to Name of new package type.
- 3) using meaning full name in line 107 replace var from t to Message.
- 4) using meaning full name in line 122 replace var from n to serial number.
- 5) using meaning full name in line 141 replace var from p to number transferred to the balance.
- 6) using meaning full name in line 144 replace var from c to transferred balance.
- 7) using meaning full name in line 191 replace var from u to number of lines.
- 8) using meaning full name in line 200 replace var from g to the numbers entered.

- 9) using meaning full name in line 229 replace var from s to number of operation. [1]
- 10) use comment “Read client's data to enter the App “in line 5.
- 11) use comment “ Read mobile number's data to provide the services available to the customer ” in line 21.
- 12) use comment “clients use this function to change package type of mobile number” in line 31 .
- 13) use comment “clients use this function to renew subscription package type” in line 43
- 14) use comment “check or Ensure that the number entered by the client is inside the App” in line 49.
- 15) use comment “This function helps the client to make a call” in line 93.
- 16) use comment “check the line from which the call was made in order to deduct the balance from it” in line 97.
- 17) use comment “This function helps the client to send a message” in line 105.
- 18) use comment “check the line from which the send a message was made in order to deduct the balance from it” in line 110.
- 19) use comment “This function helps the client to recharge balance in any lines” in line 118.
- 20) use comment ”check serial number which is entered with what is inside the program” in line 125
- 21) use comment “this function helps the client to transfer balance in any lines ” in line 139.
- 22) use comment “check any line from which the balance is transferred” in line 147.
- 23) use comment “client can choose any services from App” in line 232.
- 24) use comment “check Does the user like to complete or not?” in line 263.
- [2-3]

- 25) use function guideline in line 31 >>> In order to make it clear to the customer that the number that the package renews on is specified
- 26) use function guideline in line 93, 105 >>> In order to make it clear to the customer, 0.25 piasters was deducted from the line from which he sent a message or made a call.
- 27) use function guideline in line 118 >>>In order to make it clear that the entered serial number is compared to the numbers stored inside the program, and in the absence of it, it has not been shipped.
- 28) use function guideline in line 139>>> In order to make it clear that the balance is transferred from one customer to another customer on the same network and the value is deducted from the line that carries the amount entered in case there is more than one line. [4]
- 32) using formatting in line 4.
- 33) using formatting in line 14.
- 34) using formatting in line 29.
- 35) using formatting in line 42.
- 36) using formatting in line 61.
- 37) using formatting in line 64
- 38) using formatting in line 67.
- 39) using formatting in line 71.
- 40) using formatting in line 77.
- 41) using formatting in line 87.
- 42) using formatting in line 92.
- 43) using formatting in line 104.
- 44) using formatting in line 117.
- 45) using formatting in line 133.
- 46) using formatting in line 135.
- 47) using formatting in line 138.
- 48) using formatting in line 155.

- 49) using formatting in line 165.
- 50) using formatting in line 168.
- 51) using formatting in line 226.
- 52) using formatting in line 237.
- 53) using formatting in line 241.
- 54) using formatting in line 245.
- 55) using formatting in line 249.
- 56) using formatting in line 253.
- 57) using formatting in line 257.
- 58) using formatting in line 261.
- 59) using formatting in line 265.
- 60) using formatting in line 276. [5]

## 5. Implementation

```
1. #include <iostream>
2. #include <conio.h>
3. using namespace std;
4.
5. struct client          // Read client's data to enter the App
6. {
7.     int userID;
8.     string email;
9.     string name;
10.    char password;
11.    string address;
12.    int line[3];
13. }clienta;
14.
15. struct Date
16. {
17.     int Day;
18.     int Month;
19.     int Year;
20. };
21. struct data            // Read mobilenumber's data to provide the services available to the
    customer
22. {
23.     long int mobilenumber;
24.     string packagetype;
25.     float balance;
26.     Date Dateofstart;
27.     Date Dateofrenwel;
28. }users[3];
29.
```



```

30. long int Mobilenumber;
31. void changepakettype()           // clients use this function to change package type of mobile
    number
32. {
33.     string Nameofnewpackage;
34.     cout << "Enter Mobilenumber" "/n";
35.     cin >> Mobilenumber;
36.     cout << "Enter Name of new pakettype";
37.     cin >> Nameofnewpackage;
38.     for (int i = 0; i < clienta.line[i]; i++)
39.         cin >> users[i].pakettype;
40.     cout << Mobilenumber << "/n" << Nameofnewpackage << endl;
41. }
42.
43. void renewsubscription()         // clients use this function to renew subscription package ty
    pe
44. {
45.     cout << "Enter Mobilenumber";
46.     cin >> Mobilenumber;
47.     for (int i = 0; i < clienta.line[i]; i++)
48.     {
49.         if (clienta.line[i] == Mobilenumber) // check or Ensure that the number entered by
            the client is inside the App
50.         {
51.             if (users[i].pakettype == "high")
52.             {
53.                 if (users[i].balance >= 100)
54.                 {
55.                     users[i].balance -= 100;
56.
57.                 }
58.                 cout << "operation successful";
59.             }
60.         }
61.
62.         else
63.             cout << "error";
64.
65.         if (users[i].pakettype == "middle")
66.         {
67.
68.             if (users[i].balance >= 50)
69.             {
70.                 users[i].balance -= 50;
71.
72.             }
73.             cout << "operation successful";
74.         }
75.         else
76.             cout << "error";
77.
78.         if (users[i].pakettype == "low")
79.         {
80.             if (users[i].balance >= 15)
81.             {
82.                 users[i].balance -= 15;
83.                 cout << "operation successful";
84.             }
85.             cout << "operation successful";
86.         }
87.
88.         else
89.             cout << "error";
90.     }
91. }
92.

```

```

93. void makecall()    //This function helps the client to make a call
94. {
95.     cout << "calling";
96.     cin >> Mobilenumber;
97.     for (int i = 0; i < clienta.line[i]; i++)    //check the line from which the call was ma
de in order to deduct the balance from it
98.     {
99.         if (users[i].balance >= 0.25)
100.             users[i].balance -= 0.25;
101.     }
102.     cout << "balance-=0.25" << endl;
103. }
104.
105. void sendmessage()    //This function helps the client to send a message
106. {
107.     string Message;
108.     cout << "type Message";
109.     cin >> Message;
110.     for (int i = 0; i < clienta.line[i]; i++)    //check the line from which the send a messa
ge was made in order to deduct the balance from it
111.     {
112.         if (users[i].balance >= 0.25)
113.             users[i].balance -= 0.25;
114.     }
115.     cout << "balance -=0.25" << endl;
116. }
117.
118. void rechargebalance()    //This function helps the client to rechargebalance in any lines
119. {
120.     long int sr[5] = { 1515151515,4141414141,6398521478,6352417854,4521789652 };
121.     int value[5] = { 15,50,100,150,200 };
122.     long int serialnumber;
123.     cout << "Enter serial number";
124.     cin >> serialnumber;
125.     for (int i = 0; i < 4; i++)    // check serial number which is entered with what is inside
the program
126.     {
127.         if (sr[i] == serialnumber)
128.         {
129.             for (int j = 0; j < clienta.line[i]; j++)
130.             {
131.                 users[j].balance += value[i];
132.             }
133.
134.         }
135.
136.     }
137. }
138.
139. void transferbalance()    //This function helps the client to transfer balance in any lines
140. {
141.     long int numbertransferredtothebalance;
142.     int transfeerredbalance;
143.     cout << "Enter the number transferred to the balance " << "endl" << " Enter Transferred
balance ";
144.     cin >> numbertransferredtothebalance;
145.     cin >> transfeerredbalance;
146.
147.     for (int i = 0; i < clienta.line[1]; i++)    //check any line from which the balance is t
ransferred
148.     {
149.         if (users[i].balance >= transfeerredbalance)
150.         {
151.             users[i].balance -= transfeerredbalance;
152.         }

```

```

153.     }
154. }
155.
156. int main()
157. {
158.     Date MyDate;
159.     cout << "Enter userID: ";
160.     cin >> clienta.userID;
161.     cout << "Enter e-mail: ";
162.     cin >> clienta.email;
163.     cout << "Enter name:";
164.     cin >> clienta.name;
165.
166.     char ch;
167.     bool isSpecial = false, isNum = false;
168.
169.     cout << "Enter password :";
170.     while (true)
171.     {
172.         ch = _getch();
173.         if (ch == '\r')
174.         {
175.             break;
176.         }
177.         if (ch == '$' || ch == '%' || ch == '_' || ch == '#' || ch == '@')
178.         {
179.             isSpecial = true;
180.         }
181.         if (ch >= 47 && ch <= 58)
182.         {
183.             isNum = true;
184.         }
185.     }
186.     if (isSpecial && isNum)
187.     {
188.         cout << endl << "password is valid" << endl;
189.         cout << "Enter address:";
190.         cin >> clienta.address;
191.         int numberoflines;
192.         cout << "Enter number of lines :";
193.         cin >> numberoflines;
194.         if (numberoflines > 3)
195.         {
196.             cout << "error";
197.             return 0;
198.         }
199.         else
200.             for (long int Thenumbersentered = 0; Thenumbersentered < numberoflines; Thenumb
ersentered++)
201.             {
202.                 cout << "Enter users mobilenumbers";
203.                 cin >> users[Thenumbersentered].mobilenumbers;
204.                 cout << "Enter users packagetype ";
205.                 cin >> users[Thenumbersentered].packagetype;
206.                 cout << "Enter users balance ";
207.                 cin >> users[Thenumbersentered].balance;
208.                 cout << "Enter Date of start " << endl;
209.                 cout << "Enter Day";
210.                 cin >> MyDate.Day;
211.                 cout << "Enter Month";
212.                 cin >> MyDate.Month;
213.                 cout << "Enter Year";
214.                 cin >> MyDate.Year;
215.                 cout << MyDate.Day << "/" << MyDate.Month << "/" << MyDate.Year << endl;
216.                 cout << "Enter Date of renewal date" << endl;
217.                 cout << "Enter Day";

```

```

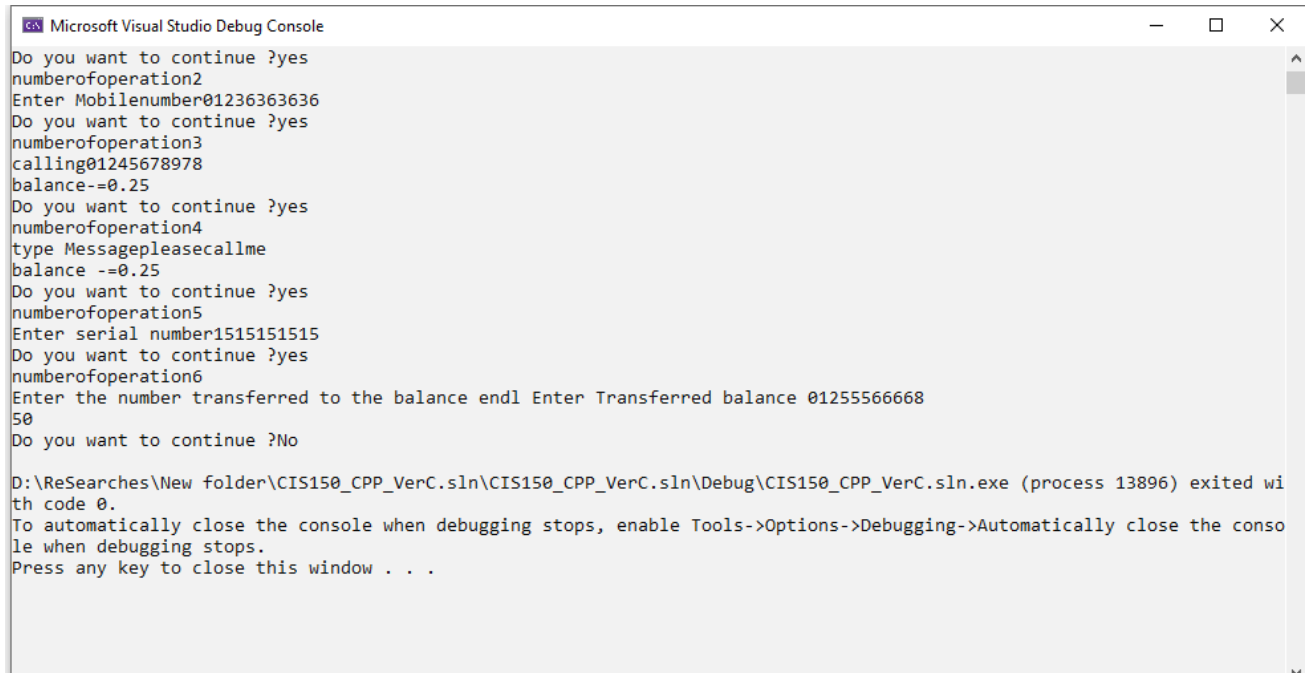
218.         cin >> MyDate.Day;
219.         cout << "Enter Month";
220.         cin >> MyDate.Month;
221.         cout << "Enter Year";
222.         cin >> MyDate.Year;
223.         cout << MyDate.Day << "/" << MyDate.Month << "/" << MyDate.Year << endl;
224.
225.     }
226.
227.     while (true)
228.     {
229.         int numerofooperation;
230.         cout << "numerofooperation";
231.         cin >> numerofooperation;
232.         switch (numerofooperation) //client can choose any services from App
233.         {
234.             case 1:
235.                 changepakagetype();
236.                 break;
237.
238.             case 2:
239.                 renewsubscription();
240.                 break;
241.
242.             case 3:
243.                 makecall();
244.                 break;
245.
246.             case 4:
247.                 sendmessage();
248.                 break;
249.
250.             case 5:
251.                 rechargebalance();
252.                 break;
253.
254.             case 6:
255.                 transferbalance();
256.                 break;
257.
258.             default:
259.                 break;
260.         }
261.
262.         string chek;
263.         cout << "Do you want to continue ?"; // check Does the user like to complete or
not?
264.         cin >> chek;
265.
266.         if (chek == "No")
267.         {
268.             break;
269.         }
270.     }
271. }
272. else
273. {
274.     cout << endl << "password is not valid" << endl;
275. }
276.
277. return 0;
278. }

```

## 6. Testing

```
Microsoft Visual Studio Debug Console
Enter userID: 987654321
Enter e-mail: Ali_Ahmed@gmail.com
Enter name:Ali
Enter password :
password is valid
Enter address:AinShams
Enter number of lines :2
Enter users mobilenumero01245454545
Enter users packagetype high
Enter users balance 200
Enter Date of start
Enter Day2
Enter Month6
Enter Year2020
2/6/2020
Enter Date of renewal date
Enter Day2
Enter Month7
Enter Year2020
2/7/2020
Enter users mobilenumero01236363636
Enter users packagetype low
Enter users balance 100
Enter Date of start
Enter Day5
Enter Month6
Enter Year2020
5/6/2020
Enter Date of renewal date
Enter Day5
```

```
Microsoft Visual Studio Debug Console
Enter users mobilenumero01236363636
Enter users packagetype low
Enter users balance 100
Enter Date of start
Enter Day5
Enter Month6
Enter Year2020
5/6/2020
Enter Date of renewal date
Enter Day5
Enter Month7
Enter Year2020
5/7/2020
numerofooperation1
Enter Mobilenumero/n01245454545
Enter Name of new paketypelow
1245454545/nlow
Do you want to continue ?yes
numerofooperation2
Enter Mobilenumero01236363636
Do you want to continue ?yes
numerofooperation3
calling01245678978
balance--0.25
Do you want to continue ?yes
numerofooperation4
type Messagepleasecallme
balance --0.25
Do you want to continue ?yes
numerofooperation5
```

A screenshot of the Microsoft Visual Studio Debug Console window. The window title is "Microsoft Visual Studio Debug Console". The output text is as follows:

```
Do you want to continue ?yes
numerofooperation2
Enter Mobilenumber01236363636
Do you want to continue ?yes
numerofooperation3
calling01245678978
balance-=0.25
Do you want to continue ?yes
numerofooperation4
type Messagepleaserecallme
balance -=0.25
Do you want to continue ?yes
numerofooperation5
Enter serial number1515151515
Do you want to continue ?yes
numerofooperation6
Enter the number transferred to the balance endl Enter Transferred balance 01255566668
50
Do you want to continue ?No

D:\ReSearches\New folder\CIS150_CPP_VerC.sln\CIS150_CPP_VerC.sln\Debug\CIS150_CPP_VerC.sln.exe (process 13896) exited wi
th code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```

## 7. Research references

- [1] Dāsa, R. (2016). Clean Code. In *Learn CakePHP* (pp. 14-17). Apress, Berkeley, CA.
- [2] Gurnett, A., Caufield, G., Frisch, D., Milburn, C., & Parmar, M. An Exploration of Clean Code.
- [3] Howard, M. J., Gupta, S., Pollock, L., & Vijay-Shanker, K. (2013, May). Automatically mining software-based, semantically-similar words from comment-code mappings. In *2013 10th Working Conference on Mining Software Repositories (MSR)* (pp. 377-386). IEEE.
- [4] Visser, J., Rigal, S., Wijnholds, G., van Eck, P., & van der Leek, R. (2016). *Building Maintainable Software, C# Edition: Ten Guidelines for Future-Proof Code*. " O'Reilly Media, Inc."
- [5] Chen, P., Harrison, M. A., & Minakata, I. (2000, January). Incremental document formatting. In *Proceedings of the ACM conference on Document processing systems* (pp. 93-100).