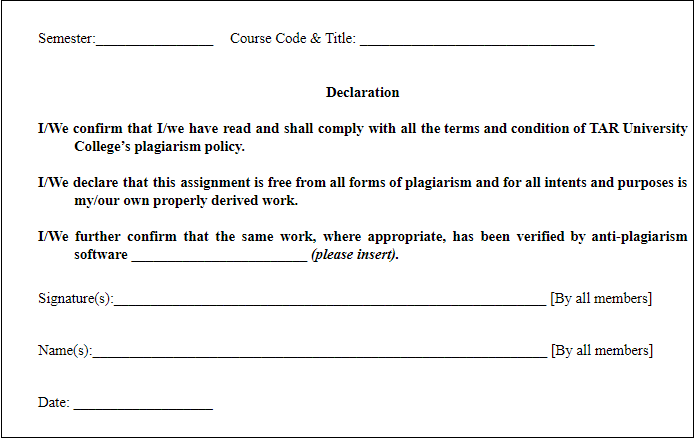
|  |
| --- |
|  |
| **Faculty of Computing and Information Technology** |
| **Assignment**  **OCT 2019 Semester** |
| |  |  |  | | --- | --- | --- | | **Course code** | **:** | AACS3064 | | **Course Title** | **:** | Computer Systems Architecture | |  |  |  | |
| |  |  |  |  | | --- | --- | --- | --- | | **Students’ Name &**  **ID No.** | **:** | Name: **\_Teo Wen Zhi**\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | ID. No.: **18WMD06564**\_\_\_\_ | | Name:\_ **Kam Wei Ming\_\_\_\_\_** \_\_\_\_\_\_ | ID. No.: **18WMD06387**\_\_\_\_ | | Name: **Kow Yann Tang**\_\_\_\_\_\_\_\_\_\_\_ | ID. No.: **18WMD06790**\_\_\_\_ | | **Programme\*** | **:** | **DCO2** | | | **Tutorial Group** | **:** | **6**\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | | **Tutor** | **:** | **Mr. Loh Kiean Nyak** | | | **Submission Date** | **:** | **Week 13, Monday, before 10 morning** | | |  | | | | |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Members’**  **Name** | **Introduction** | **Coding & logic** | **I/O design** | **Leadership and teamwork** | **Total** | | **( 15marks)** | **(40marks)** | | **(5 marks)** | **(60 marks)** | | **1. Teo Wen Zhi** |  |  |  |  |  | | **2. Kam Wei Ming** |  |  |  |  |  | | **3. Kow Yann Tang** |  |  |  |  |  | | **Comment:** | | | | |  | | **Date of submission** | | **:** | **6/1/2020(Monday)** | |  | | **Date received**  **(to-be filled by the tutor received)** | | **:** |  | |  | |

**Table of contents**

|  |  |  |
| --- | --- | --- |
| No | Contents | Page Numbers |
| 1 | Declaration Forms | 2 |
| 2 | Assessment Rubrics | 3 |
| 3 | **Introduction about the system**  (As per proposal)   * Industry selected * Company background * Functions of the Program * Formulas used * Assumptions *[If any]* * Flow chart | 4-6 |
| 4 | **Coding & logic**  (Select 3 advanced features from your program and explain) | 7-9 |
| 5 | **I/O design**   * Print-screens for the program operation with sample data. * Max 2 screens per page. * Include a label for each print-screen | 9-17 |
| 6 | **Leadership and teamwork Software used** | 18 |
| 7 | **Reference** | 19 |

**Declaration Forms**



Date : 6/1/2020

**Assignment**

Assessment Rubrics

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Introduction | **0 - 5mark** | **6- 9mark** | | **10-15 marks** |
| Insufficient background context provided. | Comprehensive and sufficient background context provided. | | Excellent background context provided. |
| Coding & logic | **0 – 9 marks** | **10 – 19 marks** | **20– 29 marks** | **30 - 40marks** |
| Unable to present the symbolic codes which are dynamic and reusable.  Poor in coding and implementation of control statement.    Unable to demonstrate a well logically flow program and error occurs | Able to present the symbolic codes which are dynamic.  Able to use codes, registers & implementation of control statement.  Able to demonstrate logical flow with some unintended syntax errors and errors during assembly and program run time. | Able to present the symbolic codes with little dynamic and reusability.  Appropriate use of codes, registers & implementation of control statement.  Able to demonstrate a well logical flow with minimum unintended syntax errors and errors during assembly and program run time. | Able to present the symbolic codes which are dynamic and reusable.  Efficient and appropriate use of codes, registers & implementation of control statement.  Able to demonstrate a well logical flow without unintended syntax errors and errors during assembly and program run time. |
| I/O design | Poor / brief in input/output design. | Proper but simple input/output design. | Effective and efficient input/output design. | Consistent, effective and efficient input/output design. |
| Leadership and teamwork | **0-2 marks** |  |  | **3-5 marks** |
| Unable to communicate and work with team mate. |  |  | able to communicate well and work with team mate. |

**1.0 Introduction**

**Company Background - Popular Bank Berhad**

Popular Bank Berhad it is a Malaysia banking industry and it is headquartered in Kuala Lumpur, Malaysia offering financial services in Malaysia as well as the Asia-Pacific region. The bank is founded in 1970 by Teo Wen Zhi, the bank was listed on the Kuala Lumpur Stock Exchange on 27 June 1995 and since then has grown by leaps and bounds, organically as well as through mergers and acquisitions. In 2010 PBB merged with the EVN Bank Group placed PBB as Malaysia’s top 3 largest banking group by total assets and premier banking group with over with over RM 400.76 billion of assets and RM90.90 billion in market capitalization in 2018. Moreover, PBB also the largest bank in Malaysia by shareholders' funds, second largest by market capitalization.

Popular Bank currently has a network of 260 branches and over 2000 self-terminals in every state of Malaysia also provided the e-banking services to help user done the cash deposit, cash withdrawal and also the money transfer with simpler and efficiency way. The company also has its presence in the Asia-Pacific region with a network of 1 subsidiary in Singapore (Popular Bank Singapore limited with 80 branches), 1 subsidiary in China (Popular Bank China limited with 30 branches), 1 subsidiary in Vietnam (Popular Bank Vietnam limited with 15 branches) and 30 branches all over the world and different country such as Indonesia, Taiwan, US and more. In total, the company serves more than 10 million customers in countries where it operates.

Nowadays, Popular Bank is having the sufficient ability provide a comprehensive range of financial products and services which include different type of banking (personal, commercial, Islamic, investment) and also different type of services (trustee services, management of unit trust funds, nominee services, general insurance products) for cares each our customers, employees, shareholders and community PBB serves. We also look for opportunity in the challenges faced by individuals and businesses in their banking needs today so that we can innovate customer experiences tomorrow, continuously provide simple, more efficient and seamless banking experience for every person to sustain the position of being the most efficient, profitable and respected premier financial institution in Malaysia.

**Proposed Function**

1. **Login**   
   When the system started, the system will first prompt the user to log in their account to continue any transaction or services. The user will be asked to provide their user id and also their password. Any fail to login will be prompting an error message.
2. **Deposit**   
   Once the user successfully login their account, they can make a deposit transaction into their account. First, the user will be prompted to enter the amount. The amount will be first verified only then will be deposit into the account. After that, it will compute the interest will be gain in certain period with a deposit amount **more than** **or equal** to **RM 5000 in their account**.
3. **Withdraw**Other than deposit money into the account, the user also can withdraw their money out of the account. As usual, the user input the amount, after a series of validation, the transaction will finally process. The money will be deducted from the account and rewrite the balance. Transaction with **amount more than or equal RM 1000** will be charged an amount of **RM 5 as a withdraw fee**. For transaction **less than RM 1000**, that transaction will be **charged only 0.3%**.
4. **Transfer**We also provide fund transfer from one to another account. As an example, user A wanted to transfer RM 1000 to another user B, so user A first required to input the account number of user B and the amount of money. After validation is done, system will confirm with the user about the transaction only then the transaction will be done. For **transaction more than or equal RM 1,000**, the system will compute **1.0% according to the transaction amount** and deducted as fee for the transaction. Maximum transfer amount will be RM9999.00(including tax).
5. **Daily Summary Report**All the transactions made by that day will be recorded. In the end, if the user wished to check transaction have been made, he/she could enter the selection to check all the transaction have been done at that certain day.

**Proposed Formulas Used**

**Deposit**

Balance + Amount Entered = New Balance

For our banking system, we provide an interest rate of **1.20% per annual** but user should with a minimum deposit of RM 5000 in the account. Balance (if >= 5000) \* 1.20% = Interest for every 12 months.

**Withdraw**

Balance – (Amount Entered (if < 1,000) + Amount Entered\* 0.3%) = New Balance

Balance – (Amount Entered (if >= 1,000) + 5) = New Balance

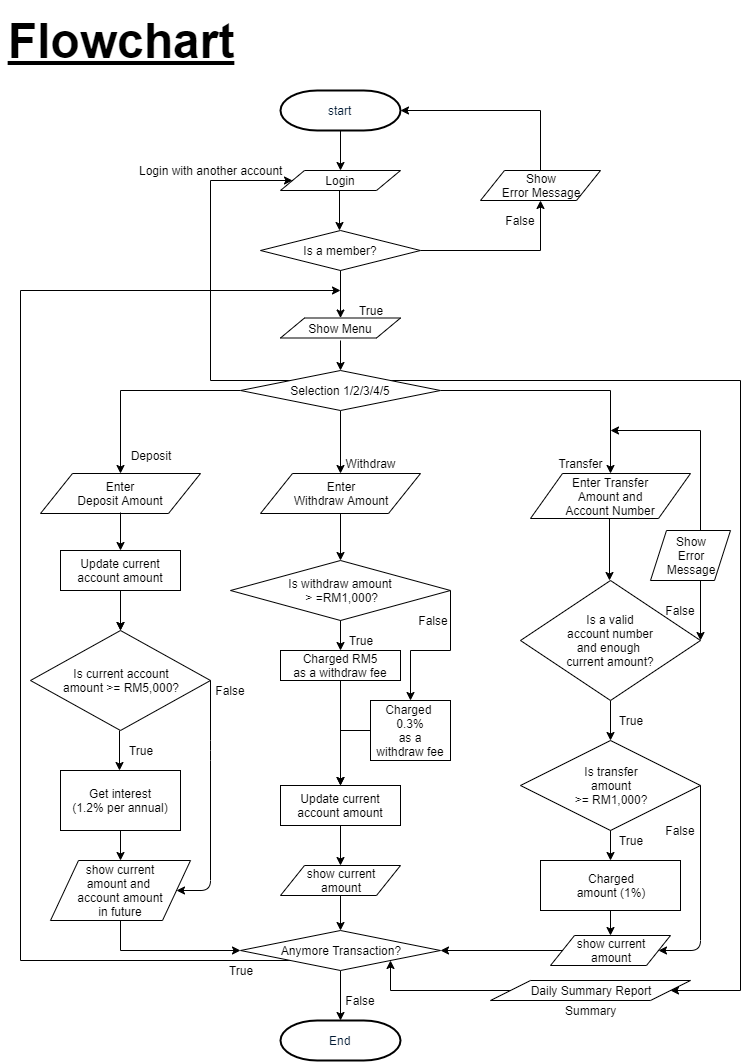
As shown above, this will be the formulas to apply in the withdraw service. Every transaction will be charged a certain amount of fees depends on the withdrawn amount

**Transfer**

Account User A: Balance – Amount Entered – (Amount Entered \* 1.0% (if amount >= 10,00))

Account User B: Balance + Amount Entered

This service allowed user transfer fund, but if the amount is more than equal RM 10,00, there will be a charge and the system will use the formulas above to calculate.

****

**2.0 Coding and logic**

**Login Function**

1. **Validation for log in account id and password input by user**

When first start the program, it will prompt the user to enter the account id and also the password. Then a sequence of validation will take place after the user done input. This is to prevent any unauthorized people to log in the account illegally and causing many security issues.

* **Logic of Login function**

The function is to actually confirming the identity of the user, also to prevent any unauthorized people to access to the account which will cause many security issues. So, first we will do validation for the account ID input, the account id will be compared with the account ids that store inside the variable. Then, validated account will only then process the next step, input password. Same like last step, the input password will be compared with the existing password. After these series of validation, the user only can enter the menu and choose the service they want.

**Deposit Function**

1. **5 digit and 2 decimal point addition process**

In this function able to add/deposit any amount within RM 99999.99 to user account based on the user input, but the result after deposit user account must not over 99999.99. Moreover, this function also perfectly done the addition calculation included carry of decimal pint/cents and carry of notes.

1. **Interest calculation**

In this function will check for the user account amount, if they have the amount more than or equal RM 5,000 in their account that means they can get 1.2% interest per year. This system will calculate and show the compound interest and also the future amount for the future amount for the future 1 year, 5 years and 10 years. Furthermore, in this function will able to show until 6-digits 2 decimal point because that can also calculate the future interest for RM 99999.99 and also display the expect date for get the interest.

* **Logic for deposit function**

Instead of using 32 bits register, in Deposit function is using the variable (tthousand, Splace) for store the fifth or above digit. But for the compatibility between each function of the system, in deposit function has limit until 6 digits only. Then, to perform the calculation of deposit is using combination of division, multiplication and comparison to do carry, find out the decimal value and store in to variable (NOTES, CENT AND THOUSAND AMOUNTS STORED), for display will process that amount and put into array (stoAmt) to one by one display the decimal number and point. Furthermore, the interest calculation is simulating the compound interest formula and the date in the interest is based current date perform addition for show the expect date.

**Withdraw Function**

1. Up to 5 digits for the amount enter by user

In this function, user can only withdraw the amount in multiple of ten since it is actually make sense with real life bank system. Even user can only withdraw amount in multiple of ten, but it still may affect the decimal point if the withdraw amount is less than RM 1,000. It is because the charge fee is times 0.3% on the amount that user withdraw.

1. Double confirmation to user and user friendly

In this function, because whatever amount you enter or select, the system will charge the withdraw fee and notice user about confirm to withdraw even though will charge some fee. Comparing to real life Auto Teller Machine (ATM), it does not have let user knowing about the charge through the screen. So, it is user friendly actually.

1. Beautiful design on withdraw menu and meaningful error message

In this function still got own menu, because it is just like Auto Teller Machine (ATM) have some common selection. For this menu having 6 different colors on every line and it is actually is a beautiful design. Other than that, i also clear the screen after a certain input, else it will look like very messy. For the error message, i got fill the message with red color because red color easy let others pay attention on it.

* **Logic for withdraw function**

Instead of using 4-byte register, in Withdraw function is using the variable name called “TTNOTE” to store the fifth digit. Then, to perform the calculation of withdraw, i am using subtraction on withdraw amount, multiplication follow by division for the charge part can a lot of comparison for the validation and also bring the value from fifth digit or Ringgit to Cent.

**Transfer Function**

1. Link between 2 accounts

While doing transfer from one account to another account, there have to be a link between the 2 accounts only that transfer could be success. So, in the transfer function, I have successfully linked the 2 accounts, account A and B together. Example, if I logged in as account A and want to transfer RM3,000.00 to account B. The system will check from which account to which account then check whether is there sufficient balance. If all passed, then it will ask confirmation from the user and check the amount transfer over Rm 1,000.00. If so, it will compute 1% of the transfer amount as transfer fee. Then the transaction take place, Account A will be deducted RM3,000.00 and RM 30(transfer fee). In the meantime, Account B will receive RM3,000.00 as it is transferred from account A. In conclusion, the linkage is between the 2 accounts are success as it could perform multiple calculation in same time in different accounts.

1. Able to display 5-digit and perform calculation for 4 digit and 2 decimals point

In transfer function, as stated above, the maximum transfer amount is RM 9900.00 and after added tax is RM 9999.00. So, the function is able to perform the calculation perfectly for the 4-digit and 2 decimals point. Also, the function will perform carry or borrow for the calculation to prevent any errors occur. For display, the function managed to display 5 digit and 2 decimal points where the amount is within RM 99999.99.

1. A simple summary after transaction is done

After finishing all thing and the user confirmed the transaction, the function will display a simple summary to show the customer about the transaction details. It will display the amount transferred, amount of tax, the total amount deducted and also the final balance of the account.

* **Logic for transfer function**

So, for it to success linking 2 accounts, the function will perform a lot validation to ensure the data flow. For example, after entered the amount, it will check which account is currently logged in and which account will accept the transfer amount. Then, it will start to compute the calculation, deduct the amount with tax in the current logged in account and add the transfer amount without tax to the other account. Before all the calculation take place, it will ask confirmation from the user to prevent any unwanted error. Last, if all things gone through the validations and passed, it will display a transaction summary to show the customer the transfer transaction details.

**Summary Function**

1. Transaction record

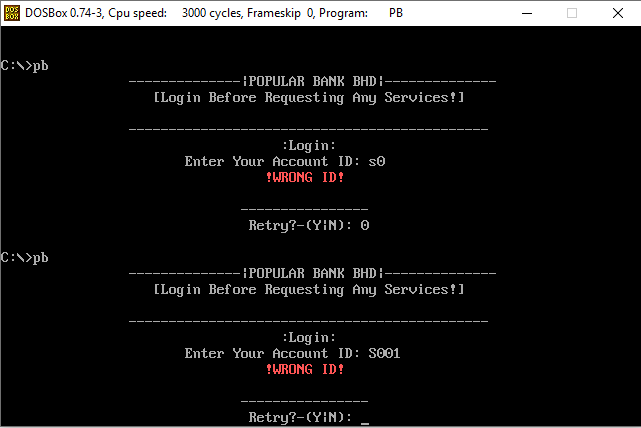
It will store every successful transaction from the beginning of the system startup to the present, but since are not using file to store the record so the record will not permanent stored and in this function is set 10 transaction as a cycle, that will replace the earliest record with latest record when start a new cycle. Furthermore, the summary also clear identifies the sequence of record that will also store the time and date of transaction.

* **Logic of Summary function**

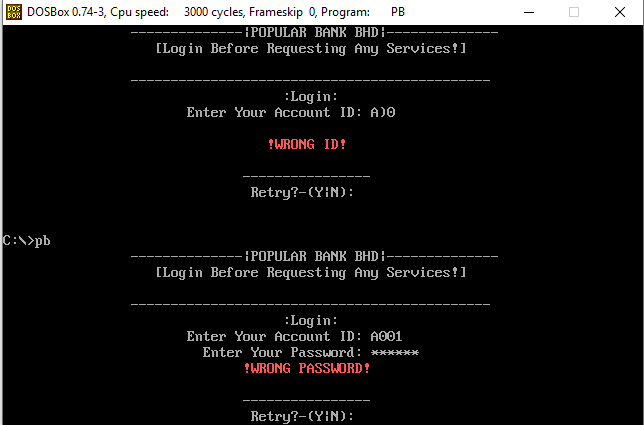
Because of not using the file to store the transaction record in this function is using combination of many different purpose and size of array to helping the whole system stored the record. The group of arrays can be roughly classified as account information and transaction time. Moreover, this function also separates by two-part store and show, every transaction done that will call this function to store it and for show need to choose the summary function in the system.

**3.0 Input/output Design**

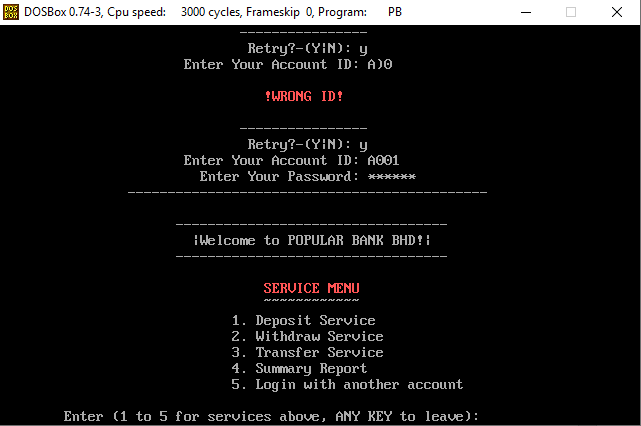
**Login**



The screenshot above showed validation for account ID.

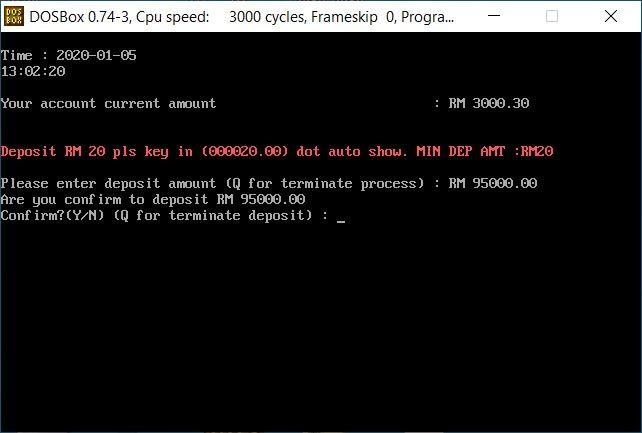


The screenshot above showed validation for the password.

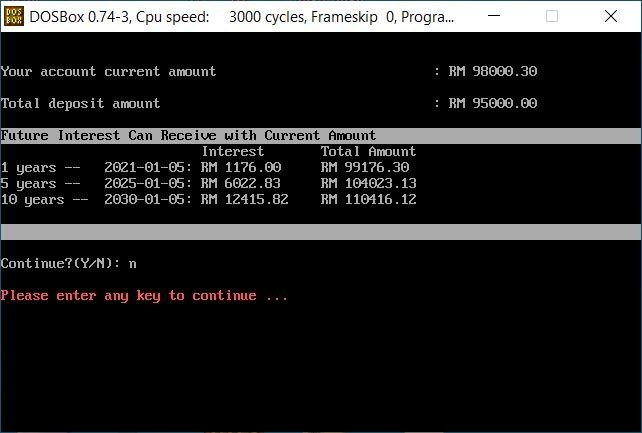


If success logged in, it will start displaying menu of and u can select it for the service stated.

**Deposit**

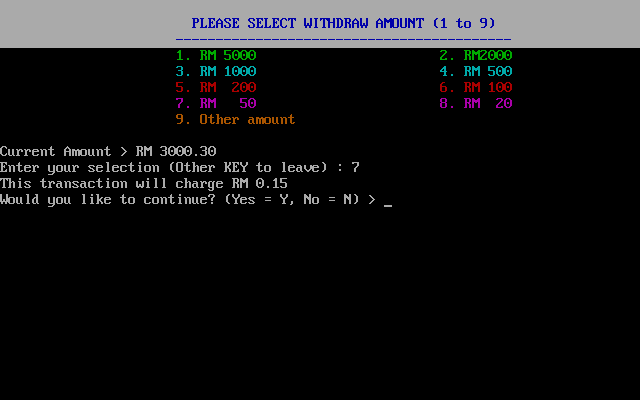


Screenshot above shown function 1 stated in the menu, deposit function, it will prompt the user for amount deposit.

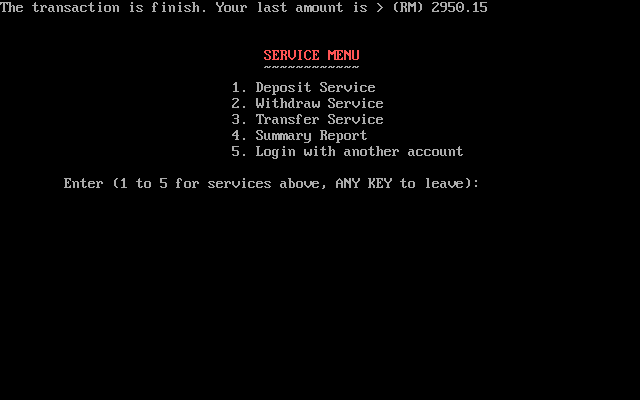


After confirmed by the user, it will auto display all the relevant details stated in the function description (total deposit amount and future interest). Then it will ask the user whether continue transaction or not.

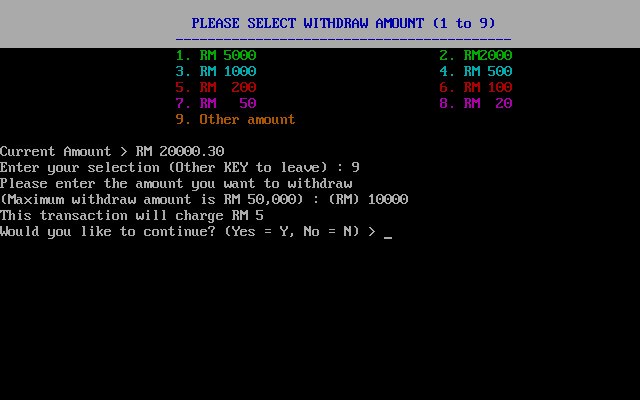
**Withdraw**



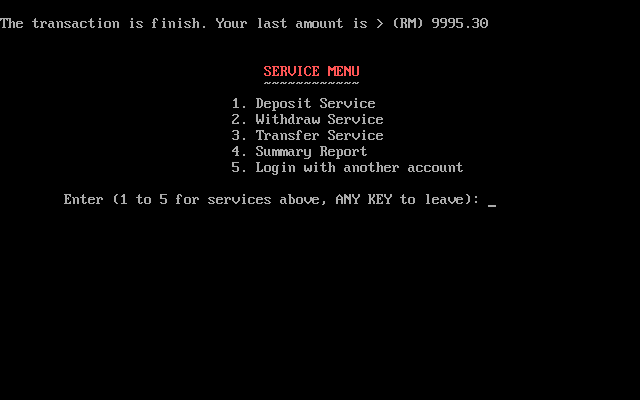
To withdraw, there are 2 ways. First, you can choose the amount at the menu provided.



After confirm the transaction.

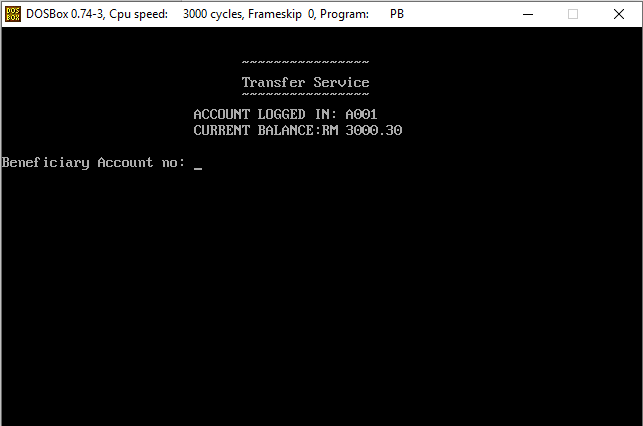


The other way is choose 9(other amount), then u can input amount you wanted to withdraw.

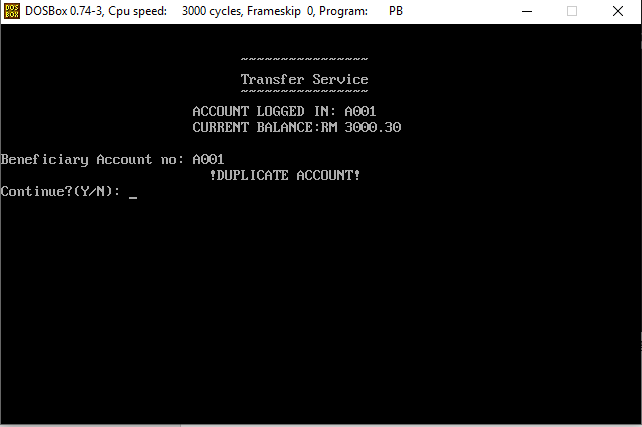


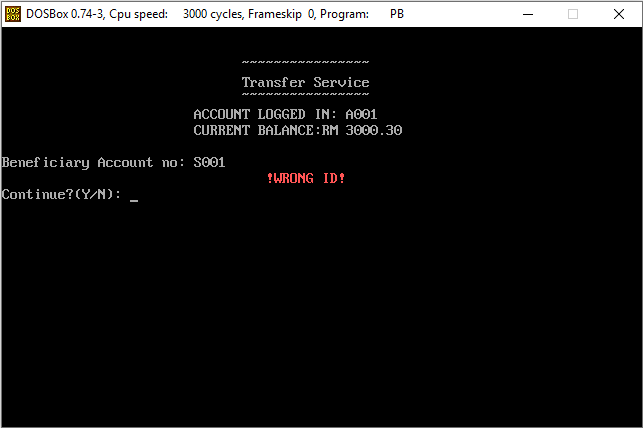
After finish the transaction**.**

**Transfer**

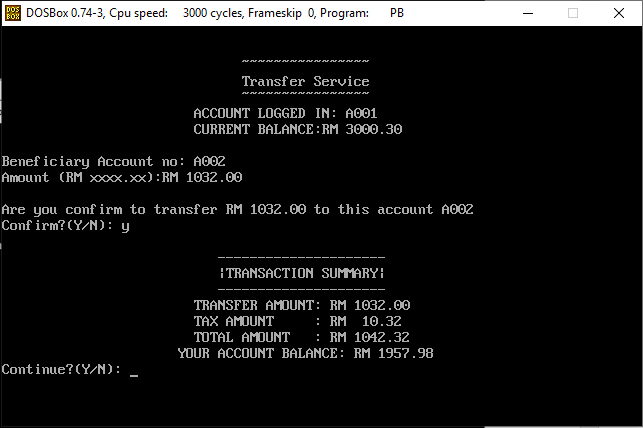


This is transfer service, first it will display the current logged in account info, then it will prompt the user to enter the others beneficiary account number.

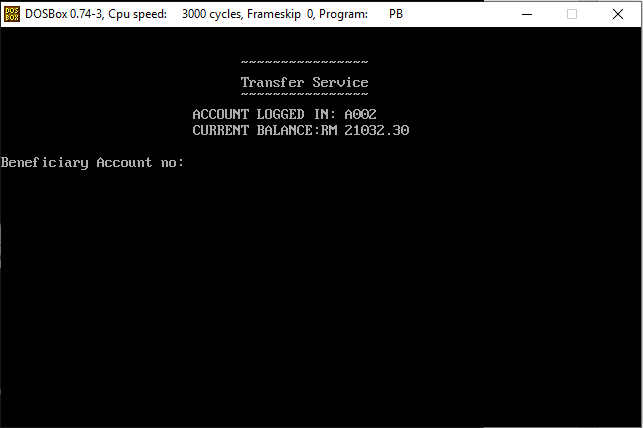




The 2 screenshot above show validation for the beneficiary account number.

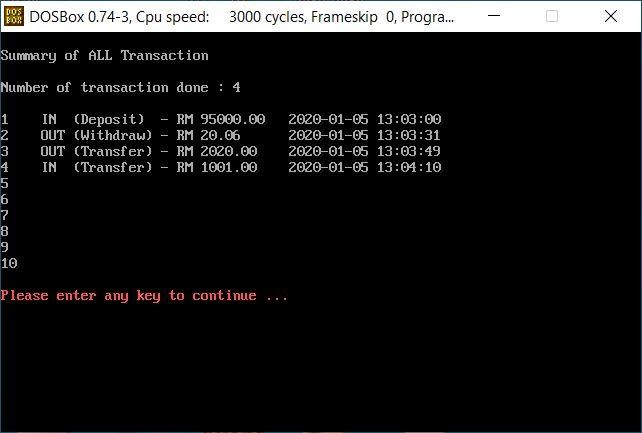
****

The screenshot above is showing after the user confirming the transaction.

****

We can see that the money is successfully transferred to account B. The original balance for account B was RM 20000.30

**Summary**

****

This is the summary part. It will display all the transaction done by the customer.

**4.0 Leadership and teamwork software used**

In this assignment, my teammates and me worked very well. During the we working out the assignment, we were able to communicate well. We frequently conduct meeting at the campus area to discuss the problems we met. Our team leader, Teo Wen Zhi will always push us and giving us some help whenever we needed. After several weeks of hard work, finally we came out with a complete program which have to be credit to all of our hard work and effort commit. Along the way we do our assignment, we used many software to also conduct our discuss our assignment issue as sometimes not all of us can meet together. Example software we used like skype, whatsapp and so on. In conclusion, all of us worked very well and hard to finish the assignment, we also worked as a team carry out any of the task stated in the assignment question. Lastly, we have to thank our tutor Mr. Loh for teaching all the knowledge we need, our leader Teo Wen Zhi for coordinate us for our assignment work, and also all the team members which willing help each other to finish the assignment.

**5.0 Reference Section**

1. Tutorial points – Arithmetic Instructions

<https://www.tutorialspoint.com/assembly_programming/assembly_arithmetic_instructions.htm?fbclid=IwAR3i5UVLDZbP3CB5gJkman7U_CRhcT-1UqI7wYUkC9-x6Dl8NbNYHcZKyi4>

2. Tutorial points – Strings

<https://www.tutorialspoint.com/assembly_programming/assembly_strings.htm>

3. Tutorial points – Arrays

<https://www.tutorialspoint.com/assembly_programming/assembly_arrays.htm>

4. Introduction to 8086 Assembly Language

<https://www.shsu.edu/~csc_tjm/fall2003/cs272/intro_to_asm.html>