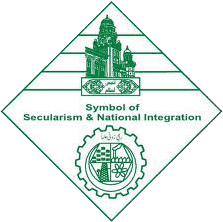
**Anjuman-I-Islam**

**M.H. Saboo Siddik Polytechnic**

****

**DATA structures USING C (DSU)**

**Microproject**

**Computer Engineering**

**Department**

**Co-3i**

**Title:**

**Year: 2022-23**

**Prepared by: Bank Management System**

* 210454: Shaikh Azlan Ahmed
* 210451: Abdurrahman Qureshi
* 210459: Owais Khan
* 210481: Faisal Ansari
* 220482: Chirag Gothankar

**Under the guidance of**: Mrs. Shafaque Ma’am

****

**MAHARASHTRA STATE**

**BOARD OF TECHNICAL EDUCATION**

**Certificate**

This is to certify that Mr. Azlan Ahmed Shaikh Roll no. 210454 of second semester of Diploma in Computer Engineering of institute M.H. Saboo Siddik Polytechnic(code:0002) has completed microproject satisfactorily in the subject: DSU (22317) for the academic year 2022-23 as prescribed in the curriculum.

Enrollment no: 2100020093

Place: Byculla, Mumbai

Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Head of the department:\_\_\_\_\_\_\_\_\_



SEAL OF



INSTITUDE



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Subject faculty Principal

****

**MAHARASHTRA STATE**

**BOARD OF TECHNICAL EDUCATION**

**Certificate**

This is to certify that Mr. Abdurrahman Qureshi Roll no. 210451 of second semester of Diploma in Computer Engineering of institute M.H. Saboo Siddik Polytechnic(code: 0002) has completed microproject satisfactorily in the subject: DSU (22317) for the academic year 2022-23 as prescribed in the curriculum.

Enrollment no: 2100020112

Place: Byculla, Mumbai

Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Head of the department:\_\_\_\_\_\_\_\_\_



SEAL OF



INSTITUDE



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Subject faculty Principal

****

**MAHARASHTRA STATE**

**BOARD OF TECHNICAL EDUCATION**

**Certificate**

This is to certify that Mr. Owais Khan Roll no. 210459 of second semester of Diploma in Computer Engineering of institute M.H. Saboo Siddik Polytechnic(code:0002) has completed microproject satisfactorily in the subject: DSU (22317) for the academic year 2022-23 as prescribed in the curriculum.

Enrollment no: 2100020140

Place: Byculla, Mumbai

Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Head of the department:\_\_\_\_\_\_\_\_\_



SEAL OF



INSTITUDE



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Subject faculty Principal

****

**MAHARASHTRA STATE**

**BOARD OF TECHNICAL EDUCATION**

**Certificate**

This is to certify that Mr. Faisal Ansari Roll no. 210481 of second semester of Diploma in Computer Engineering of institute M.H. Saboo Siddik Polytechnic(code:0002) has completed microproject satisfactorily in the subject: DSU (22317) for the academic year 2022-23 as prescribed in the curriculum.

Enrollment no: 2200020620

Place: Byculla, Mumbai

Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Head of the department:\_\_\_\_\_\_\_\_\_



SEAL OF



INSTITUDE



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Subject faculty Principal

****

**MAHARASHTRA STATE**

**BOARD OF TECHNICAL EDUCATION**

**Certificate**

This is to certify that Mr. Chirag Gothankar Roll no. 220482 of second semester of Diploma in Computer Engineering of institute M.H. Saboo Siddik Polytechnic(code:0002) has completed microproject satisfactorily in the subject: DSU (22317) for the academic year 2022-23 as prescribed in the curriculum.

Enrollment no:2200020623

Place: Byculla, Mumbai

Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Head of the department:\_\_\_\_\_\_\_\_\_



SEAL OF



INSTITUDE



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Subject faculty Principal

**Micro Project Evaluation Sheet**

**Name of Student: Abdurrahman Qureshi Enrollment No: 2100020112**

**Name of Programme: BANK MANAGEMENT SYSTEM Semester: 3rd**

**Course Title: Data structures using C Code: 22317**

**Title of the Micro-Project: BANK Management System**

### Course Outcomes Achieved: -

1. **Write a C program to perform insertion, deletion and searching operations on BST**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr. No.** | **Characteristics to be assessed** | **Poor (Marks 1** - **3)** | **Average (Marks 4** - **5)** | **Good**  **(Marks 6** - **8)** | **Excellent (Marks 9- 10)** | **Sub Total** |
| (A) Process and Product Assessment (Convert above total marks out of 6 Marks) | | | | | |  |
| 1 | Relevance to the course |  |  |  |  |  |
| 2 | Literature Review/information collection |  |  |  |  |
| 3 | Completion of the Target as per project proposal |  |  |  |  |
| 4 | Analysis of Data and representation |  |  |  |  |
| 5 | Quality of Prototype/Model |  |  |  |  |
| 6 | Report Preparation |  |  |  |  |
| (B) Individual Presentation/ Viva (Convert above total marks out of 4 Marks) | | | | | |  |
| 7 | Presentation |  |  |  |  |  |
| 8 | Viva |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **(A)**  **Process and Product Assessment (6 marks)** | **(B)**  **Individual Presentation & viva (4 marks)** | **Total Marks**  **10** |
|  |  |  |

### Comments/Suggestions about teamwork/leadership/inter-personal communication (if any)

**…………………………………………………………………………………………………………………………………………………………………………………………………………………………**

**Name and designation of the Teacher …………………........................**

### Dated Signature ……………………………………

**Micro Project Evaluation Sheet**

**Name of Student: Azlan Shaikh Enrollment No: 210020093**

**Name of Programme: BANK MANAGEMENT SYSTEM Semester: 3rd**

**Course Title: Data structures using C Code: 22317**

**Title of the Micro-Project: BANK Management System**

### Course Outcomes Achieved: -

1. **Write a C program to perform insertion, deletion and searching operations on BST**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr. No.** | **Characteristics to be assessed** | **Poor (Marks 1** - **3)** | **Average (Marks 4** - **5)** | **Good**  **(Marks 6** - **8)** | **Excellent (Marks 9- 10)** | **Sub Total** |
| (A) Process and Product Assessment (Convert above total marks out of 6 Marks) | | | | | |  |
| 1 | Relevance to the course |  |  |  |  |  |
| 2 | Literature Review/information collection |  |  |  |  |
| 3 | Completion of the Target as per project proposal |  |  |  |  |
| 4 | Analysis of Data and representation |  |  |  |  |
| 5 | Quality of Prototype/Model |  |  |  |  |
| 6 | Report Preparation |  |  |  |  |
| (B) Individual Presentation/ Viva (Convert above total marks out of 4 Marks) | | | | | |  |
| 7 | Presentation |  |  |  |  |  |
| 8 | Viva |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **(A)**  **Process and Product Assessment (6 marks)** | **(B)**  **Individual Presentation & viva (4 marks)** | **Total Marks**  **10** |
|  |  |  |

### Comments/Suggestions about teamwork/leadership/inter-personal communication (if any)

**…………………………………………………………………………………………………………………………………………………………………………………………………………………………**

**Name and designation of the Teacher …………………........................**

### Dated Signature ……………………………………

**Micro Project Evaluation Sheet**

**Name of Student: Chirag Gothankar Enrollment No: 2200020623**

**Name of Programme: BANK MANAGEMENT SYSTEM Semester: 3rd**

**Course Title: Data structures using C Code: 22317**

**Title of the Micro-Project: BANK Management System**

### Course Outcomes Achieved: -

1. **Write a C program to perform insertion, deletion and searching operations on BST**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr. No.** | **Characteristics to be assessed** | **Poor (Marks 1** - **3)** | **Average (Marks 4** - **5)** | **Good**  **(Marks 6** - **8)** | **Excellent (Marks 9- 10)** | **Sub Total** |
| (A) Process and Product Assessment (Convert above total marks out of 6 Marks) | | | | | |  |
| 1 | Relevance to the course |  |  |  |  |  |
| 2 | Literature Review/information collection |  |  |  |  |
| 3 | Completion of the Target as per project proposal |  |  |  |  |
| 4 | Analysis of Data and representation |  |  |  |  |
| 5 | Quality of Prototype/Model |  |  |  |  |
| 6 | Report Preparation |  |  |  |  |
| (B) Individual Presentation/ Viva (Convert above total marks out of 4 Marks) | | | | | |  |
| 7 | Presentation |  |  |  |  |  |
| 8 | Viva |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **(A)**  **Process and Product Assessment (6 marks)** | **(B)**  **Individual Presentation & viva (4 marks)** | **Total Marks**  **10** |
|  |  |  |

### Comments/Suggestions about teamwork/leadership/inter-personal communication (if any)

**…………………………………………………………………………………………………………………………………………………………………………………………………………………………**

**Name and designation of the Teacher …………………........................**

### Dated Signature ……………………………………

**Micro Project Evaluation Sheet**

**Name of Student: Faisal Ansari Enrollment No: 2200020620**

**Name of Programme: BANK MANAGEMENT SYSTEM Semester: 3rd**

**Course Title: Data structures using C Code: 22317**

**Title of the Micro-Project: BANK Management System**

### Course Outcomes Achieved: -

1. **Write a C program to perform insertion, deletion and searching operations on BST**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr. No.** | **Characteristics to be assessed** | **Poor (Marks 1** - **3)** | **Average (Marks 4** - **5)** | **Good**  **(Marks 6** - **8)** | **Excellent (Marks 9- 10)** | **Sub Total** |
| (A) Process and Product Assessment (Convert above total marks out of 6 Marks) | | | | | |  |
| 1 | Relevance to the course |  |  |  |  |  |
| 2 | Literature Review/information collection |  |  |  |  |
| 3 | Completion of the Target as per project proposal |  |  |  |  |
| 4 | Analysis of Data and representation |  |  |  |  |
| 5 | Quality of Prototype/Model |  |  |  |  |
| 6 | Report Preparation |  |  |  |  |
| (B) Individual Presentation/ Viva (Convert above total marks out of 4 Marks) | | | | | |  |
| 7 | Presentation |  |  |  |  |  |
| 8 | Viva |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **(A)**  **Process and Product Assessment (6 marks)** | **(B)**  **Individual Presentation & viva (4 marks)** | **Total Marks**  **10** |
|  |  |  |

### Comments/Suggestions about teamwork/leadership/inter-personal communication (if any)

**…………………………………………………………………………………………………………………………………………………………………………………………………………………………**

**Name and designation of the Teacher …………………........................**

### Dated Signature ……………………………………

***Microproject proposal***

**Title of microproje­ct: Bank Management System**

**Aims/Benefits of microproject**

Bank management system refers to the process of allowing users to create a new account, view the account’s records, make deposits and withdrawals. The discipline involves solving financial applications of a customer in banking Environment in order to nurture the needs of an end banking user by providing various ways to perform banking tasks.

The goal of any good Bank management system is to help bank managers keep track of the banking end users. It is used to keep the records of client, employee etc in bank. The system provides the access to the customer to create an account, deposit/withdraw cash from his account. The following presentation provides the specification for the system.

The benefits are both operational and financial. Not only will it serve to improve performance, but it’s also useful for tracking of hacking into account with the help of account tracking and security

1. **Course outcomes addressed**

b. Apply different searching and ordering techniques.

d. Implement basic operations on Linked List using Binary Search tree.

1. **Proposed methodology**

* Discussion of the given topic among group members.
* Literature survey
* Submission of project proposal
* Analysis of data
* Work divided among group members
* Compilation of content
* Representation
* Editing the content as per the instructions
* Report Preparation
* Viva and presentation

1. **Action Plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Weeks** | **Details of activity** | **Planned start date** | **Planned finish date** | **Name of responsible team members** |
| 1st | Discussions & finalization of topics |  |  |  |
| 2nd | Preparation of abstract |  |  |  |
| 3rd | Literature review |  |  |  |
| 4th & 5th | Collection of data |  |  |  |
| 6th & 7th | Discussion of outline of content |  |  |  |
| 8th & 9th | Formation of content |  |  |  |
| 10th | Editing & proof reading of content |  |  |  |
| 11th | Compilation of report & preparation |  |  |  |
| 12th | Final submission of microproject |  |  |  |
| 13th | Viva |  |  |  |

1. **Resources required**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. no.** | **Name of resources** | **Specifications** | **Qty** | **Remarks** |
| 1. | Online | Learning resources and various  websites | 3 sites | - |
| 2. | Desktop | Microsoft word, Turbo C++, VS Code with internet  facility. | 1 for each | - |
| 3. | Printer | Inkjet/Laser | 1 | - |
| 4. | Stationary | Papers, Spiral  binding, pictures,etc. | 1 chart  paper, pictures | - |

***Microproject Report***

**Title of Micro-project: Bank Management System**

1. **Rationale**

The primary purpose or rationale of Bank management is to ensure Customers safety for their property. In the bank, customers deposit and withdraw their money. Now to keep the belief and trust of customer, there is a positive need for management of the Bank. Smooth and efficient management is key to running a successful managing of banks. Regular feedback from the customer can increase the trust and efficient of working of a bank

1. **Aims/Benefits of microproject**

The benefits are both operational and financial. Not only will it serve to improve performance, but it’s also useful for tracking of hacking into account with the help of account tracking and security

1. **Course outcomes achieved**

b. Apply different searching and ordering techniques.

d. Implement basic operations on Linked List using binary search tree.

1. **Literature Review**

The purpose of this micro-project topic to create a Bank management system that will help us to manage the accounts of customer and reduces tedious work of managing accounts manually .It manages all the transactions like account creation, deposit and withdrawal and achieve customers satisfaction.

Our micro-project deals with customer login where customer can create account, delete and deposit/withdraw money from account and check or update his account. As the customer withdraw money from his/her account then the amount balanced in the account should be minimum $500. If the said condition is not met then the customer cannot proceed to withdraw amount later.

1. **Actual methodology followed**
   * Discussed the micro-project topic among group members.
   * Surveyed Literature based on Bank Management System
   * Submitted our project proposal
   * Analysed data regarding the topic
   * Work was divided among group members
   * Content related to the topic was compiled
   * Edited the content as per the instructions
   * Prepared the source code
   * Prepared the report
   * Presented the project and Viva

1. **Skills developed/ Learning outcomes**

* Working in groups taking help of others and helping them too.
* Studying the details of Bank management system
* Enhanced skills in data structure using c subject.
* Designing microproject with minimum required resources and at low cost.
* Teamwork- Learning to work in team and boost individual confidence.
* Time management- Completion of microproject as scheduled.
* Technical writing- Preparing a report of proposed plan and report.

1. **Applications of this microproject**

* For solving financial applications of a customer in banking environment in order to nurture the needs of an end banking user by providing various ways to perform banking tasks.
* Online accounting facility, which includes checking current balance, making deposits and overdrafts, checking interest charges and maintain a record. ATM machines which are fully automated are making it even easier for customers to deal with banks.

**ALGORITHM FOR INSERTING A NODE IN BINARY SEARCH TREE**

1. Create a new BST node and assign values to it.

2. insert(node, key)

     i) If root == NULL,

         return the new node to the calling function.

     ii) if root=>data < key

         call the insert function with root=>right and assign the return value in root=>right.

        root->right = insert(root=>right,key)

     iii) if root=>data > key

         call the insert function with root->left and assign the return value in root=>left.

         root=>left = insert(root=>left,key)

3. Finally, return the original root pointer to the calling function.

**ALGORITHM FOR SEARCHING A NODE IN BINARY SEARCH TREE**

Search (root, item)

Step 1 - if (item = root → data) or (root = NULL)

return root.

else if (item < root → data)

return Search(root → left, item)

else.

return Search(root → right, item)

END if.

**ALGORITHM FOR TRAVERSING A NODE(IN-ORDER) IN BINARY SEARCH TREE**

1.Traverse the left subtree, i.e., call Inorder(left->subtree)

2.Visit the root.

3.Traverse the right subtree, i.e., call Inorder(right->subtree)

**ALGORITHM FOR SEARCHING A NODE(PRE-ORDER) IN BINARY SEARCH TREE**

1. Visit the root.
2. Traverse the left subtree, i.e., call Preorder(left->subtree)
3. Traverse the right subtree, i.e., call Preorder(right->subtree)

**ALGORITHM FOR SEARCHING A NODE(POST-ORDER) IN BINARY SEARCH TREE**

1.Traverse the left subtree, i.e., call Postorder(left->subtree)

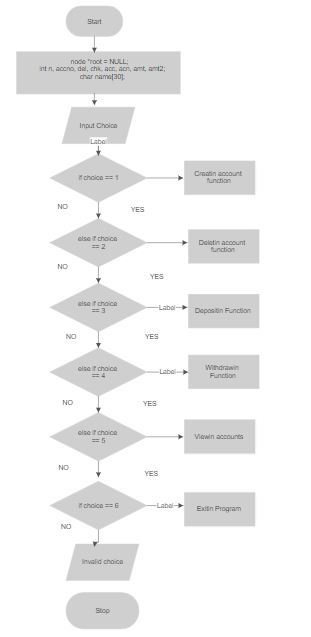
2.Traverse the right subtree, i.e., call Postorder(right->subtree)

3.Visit the root

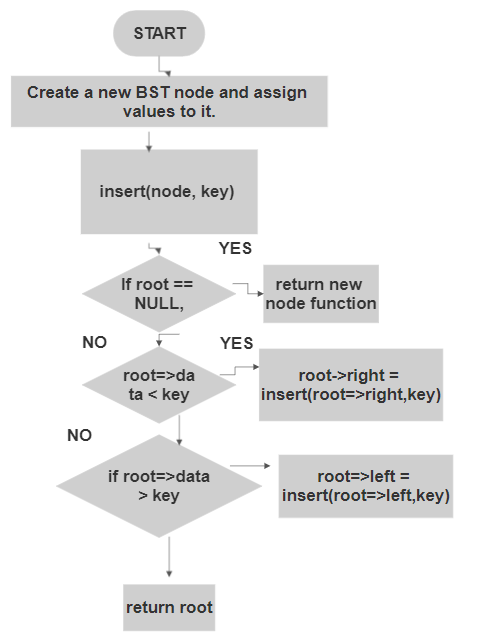
**ALGORITHM FOR IDELETING A NODE IN BINARY SEARCH TREE**

1. DELETE(T, rightchild)
2. if rightchild.left == NULL
3. TRANSPLANT(T, rightchild, rightchild.right)
4. elseif rightchild.right == NULL
5. TRANSPLANT(T, rightchild, rightchild.left)
6. else
7. leftchild = MINIMUM(rightchild.right) //minimum element in right subtree
8. if leftchild.parent != rightchild //rightchild is not direct child
9. TRANSPLANT(T, leftchild, leftchild.right)
10. leftchild.right = rightchild.right
11. leftchild.right.parent = leftchild
12. TRANSPLANT(T, rightchild, leftchild)
13. leftchild.left = rightchild.left
14. leftchild.left.parent = leftchild

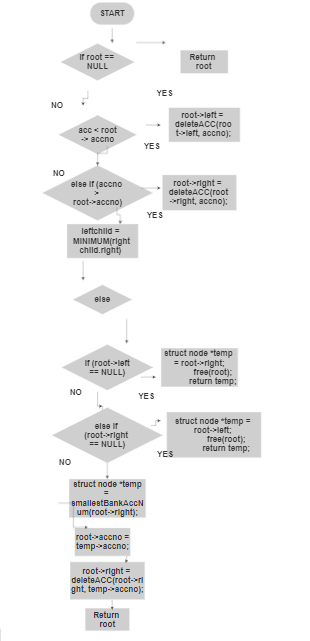
**DSU\_PROJECT FLOWCHART**

****

**INSERTION FLOWCHART**

****

**DELETIONFLOWCHART**

****

**SOURCE CODE:**

**//-------------------------------------------------------------------------------//**

**// Header Files**

**//-------------------------------------------------------------------------------//**

**#include <stdio.h>**

**#include <conio.h>**

**#include <stdlib.h>**

**#include <string.h>**

**//-------------------------------------------------------------------------------//**

**// Tree Structure**

**//-------------------------------------------------------------------------------//**

**typedef struct node**

**{**

**int accno;**

**int balance;**

**char name[30];**

**struct node \*left, \*right;**

**} node;**

**//-------------------------------------------------------------------------------//**

**// New Noden Function**

**//-------------------------------------------------------------------------------//**

**struct node \*newNode(char \*name, int accno)**

**{**

**node \*temp = (node \*)malloc(sizeof(struct node));**

**strcpy(temp->name, name);**

**temp->balance = 500;**

**temp->accno = accno;**

**temp->left = temp->right = NULL;**

**return temp;**

**}**

**//-------------------------------------------------------------------------------//**

**// Findin Smallestn Function**

**//-------------------------------------------------------------------------------//**

**struct node \*smallestBankAccNum(struct node \*root)**

**{**

**struct node \*current = root;**

**while (current && current->left != NULL)**

**current = current->left;**

**return current;**

**}**

**//-------------------------------------------------------------------------------//**

**// Deletin Function**

**//-------------------------------------------------------------------------------//**

**struct node \*deleteACC(struct node \*root, int accno)**

**{**

**if (root == NULL)**

**return root;**

**if (accno < root->accno)**

**root->left = deleteACC(root->left, accno);**

**else if (accno > root->accno)**

**root->right = deleteACC(root->right, accno);**

**else**

**{**

**if (root->left == NULL)**

**{**

**struct node \*temp = root->right;**

**free(root);**

**return temp;**

**}**

**else if (root->right == NULL)**

**{**

**struct node \*temp = root->left;**

**free(root);**

**return temp;**

**}**

**struct node \*temp = smallestBankAccNum(root->right);**

**root->accno = temp->accno;**

**root->right = deleteACC(root->right, temp->accno);**

**}**

**return root;**

**}**

**//-------------------------------------------------------------------------------//**

**// Insertin Function**

**//-------------------------------------------------------------------------------//**

**struct node \*insert(struct node \*root, char \*name, int accno)**

**{**

**if (root == NULL)**

**return newNode(name, accno);**

**if (accno < root->accno && accno != root->accno)**

**root->left = insert(root->left, name, accno);**

**else**

**root->right = insert(root->right, name, accno);**

**return root;**

**}**

**//-------------------------------------------------------------------------------//**

**// In-Orderin Function**

**//-------------------------------------------------------------------------------//**

**void inorder(struct node \*root)**

**{**

**if (root != NULL)**

**{**

**inorder(root->left);**

**printf("\t\t\t%s\t\t\t\t%d\t\t\t\t%d\t\t\t\n\n\n", root->name, root->accno, root->balance);**

**inorder(root->right);**

**}**

**}**

**//-------------------------------------------------------------------------------//**

**// Post-Orderin Function**

**//-------------------------------------------------------------------------------//**

**void preorder(struct node \*root)**

**{**

**if (root != NULL)**

**{**

**printf("\t\t\t%s\t\t\t\t%d\t\t\t\t%d\t\t\t\n\n\n", root->name, root->accno, root->balance);**

**preorder(root->left);**

**preorder(root->right);**

**}**

**}**

**//-------------------------------------------------------------------------------//**

**// Pre-Orderin Function**

**//-------------------------------------------------------------------------------//**

**void postorder(struct node \*root)**

**{**

**if (root != NULL)**

**{**

**postorder(root->left);**

**postorder(root->right);**

**printf("\t\t\t%s\t\t\t\t%d\t\t\t\t%d\t\t\t\n\n\n", root->name, root->accno, root->balance);**

**}**

**}**

**//-------------------------------------------------------------------------------//**

**// Depostin Function**

**//-------------------------------------------------------------------------------//**

**struct node \*deposit(node \*root, int acc, int amt)**

**{**

**node \*temp;**

**temp = root;**

**if (temp == NULL)**

**{**

**printf("\n\t\t\t\t\t\tYou Haven't Created an account yet...\ntry creating an account\n\n");**

**return NULL;**

**}**

**if (acc == temp->accno)**

**{**

**temp->balance += amt;**

**return root;**

**}**

**else if (temp->accno > acc)**

**{**

**return deposit(root->left, acc, amt);**

**}**

**else**

**{**

**return deposit(root->right, acc, amt);**

**}**

**}**

**//-------------------------------------------------------------------------------//**

**// Withdrawin Function**

**//-------------------------------------------------------------------------------//**

**struct node \*withdraw(node \*root, int acc, int amt)**

**{**

**node \*temp;**

**temp = root;**

**if (temp == NULL)**

**{**

**printf("\n\t\t\t\t\t\tYou Haven't Created an account yet...\ntry creating an account\n\n");**

**return NULL;**

**}**

**if (acc == temp->accno)**

**{**

**if (acc <= 500)**

**{**

**printf("\n\t\t\t\t\t\tOperation Denied:Minimum Deposit amount should be $500");**

**return 0;**

**}**

**else**

**{**

**temp->balance -= amt;**

**return root;**

**}**

**}**

**else if (temp->accno > acc)**

**{**

**return deposit(root->left, acc, amt);**

**}**

**else**

**{**

**return deposit(root->right, acc, amt);**

**}**

**}**

**//-------------------------------------------------------------------------------//**

**// Menu Function**

**//-------------------------------------------------------------------------------//**

**int menu()**

**{**

**int chc;**

**printf("\n\n------------------------------------------> BANKING MANAGEMENT SYSTEM <-----------------------------------------------");**

**printf("\n\n\n-----------------------------------------[ WELCOME TO Omega Trust CORPS ]---------------------------------------------\n\n\n");**

**printf("\t\t\t\t\t\t1.Create an Account\n\t\t\t\t\t\t2.Delete an Account\n\t\t\t\t\t\t3.Deposit\n\t\t\t\t\t\t4.Withdrawal\n\t\t\t\t\t\t5.View Account\n\t\t\t\t\t\t6.Exit\n");**

**printf("\n\n\t\t\t\t\t\tChoose An action = ");**

**scanf("%d", &chc);**

**return (chc);**

**}**

**//-------------------------------------------------------------------------------//**

**// Project Function**

**//-------------------------------------------------------------------------------//**

**void DSU\_PROJECT()**

**{**

**system("color a");**

**// system("font 32");**

**node \*root = NULL;**

**int n, accno, del, chk, acc, acn, amt, amt2;**

**char name[30];**

**A:**

**n = menu();**

**switch (n)**

**{**

**case 1:**

**printf("\n\t\t\t\t\t\tEnter Your Name = ");**

**scanf("%s", name);**

**printf("\t\t\t\t\t\tEnter Account Number = ");**

**scanf("%d", &accno);**

**root = insert(root, name, accno);**

**printf("\n\t\t\t\t\t<<<<Account Created Successfully>>>>\n");**

**goto A;**

**case 2:**

**printf("\t\t\t\t\t\tEnter Account No = ");**

**scanf("%d", &del);**

**root = deleteACC(root, del);**

**printf("\n\t\t\t\t\t<<<<Account Deleted Successfully>>>>\n");**

**goto A;**

**case 3:**

**printf("\t\t\t\t\t\tEnter the Account Number = ");**

**scanf("%d", &acc);**

**printf("\t\t\t\t\t\tEnter the Amount to Deposit = ");**

**scanf("%d", &amt);**

**root = deposit(root, acc, amt);**

**if (root != NULL)**

**{**

**printf("\t\t\t\t\t\tAccount holder = %s", root->name);**

**printf("\n\t\t\t\t\t\tAccount Number = %d", root->accno);**

**printf("\n\t\t\t\t\t\tAccount Balance = $%d", root->balance);**

**}**

**goto A;**

**case 4:**

**printf("\t\t\t\t\t\tEnter Account Number = ");**

**scanf("%d", &acn);**

**printf("\t\t\t\t\t\tEnter the Amount to Withdraw =$");**

**scanf("%d", &amt2);**

**root = withdraw(root, acn, amt2);**

**if (root != NULL)**

**{**

**printf("\t\t\t\t\t\tAccount holder = %s", root->name);**

**printf("\n\t\t\t\t\t\tAccount Number = %d", root->accno);**

**printf("\n\t\t\t\t\t\tAccount Balance = $%d", root->balance);**

**}**

**goto A;**

**case 5:**

**printf("\n\t\t\t\t\t\t1.In-order\n\t\t\t\t\t\t2.Pre-order\n\t\t\t\t\t\t3.Post-order\n");**

**printf("\t\t\t\t\t\tChoose an Action = ");**

**scanf("%d", &chk);**

**if (chk == 1)**

**{**

**printf("\n\t\t\t\t\tAccounts are Arranged in (in-order)\n ");**

**printf("\t\t\t\t\t----------------------------------");**

**printf("\n\t\t\t\t\t<<<<Account Details>>>>\n\n");**

**printf("\n\n\n\t\t\tAccount Holder\t\t\tAccount Number\t\t\tAccount Balance\n\n\n");**

**inorder(root);**

**}**

**else if (chk == 2)**

**{**

**printf("\n\t\t\t\t\tAccounts are Arranged in (Pre-order)\n ");**

**printf("\t\t\t\t\t----------------------------------");**

**printf("\n\t\t\t\t\t<<<<Account Details>>>>\n\n");**

**printf("\n\n\n\t\t\tAccount Holder\t\t\tAccount Number\t\t\tAccount Balance\n\n\n");**

**preorder(root);**

**}**

**else**

**{**

**printf("\n\t\t\t\t\tAccounts are Arranged in (Post-order)\n ");**

**printf("\t\t\t\t\t----------------------------------");**

**printf("\n\t\t\t\t\t<<<<Account Details>>>>\n\n");**

**printf("\n\n\n\t\t\tAccount Holder\t\t\tAccount Number\t\t\tAccount Balance\n\n\n");**

**postorder(root);**

**}**

**goto A;**

**case 6:**

**return 0;**

**default:**

**printf("\t\t\t\t\t\tInvalid Choice!!");**

**goto A;**

**}**

**getch();**

**}**

**//-------------------------------------------------------------------------------//**

**// Main Function**

**//-------------------------------------------------------------------------------//**

**int main()**

**{**

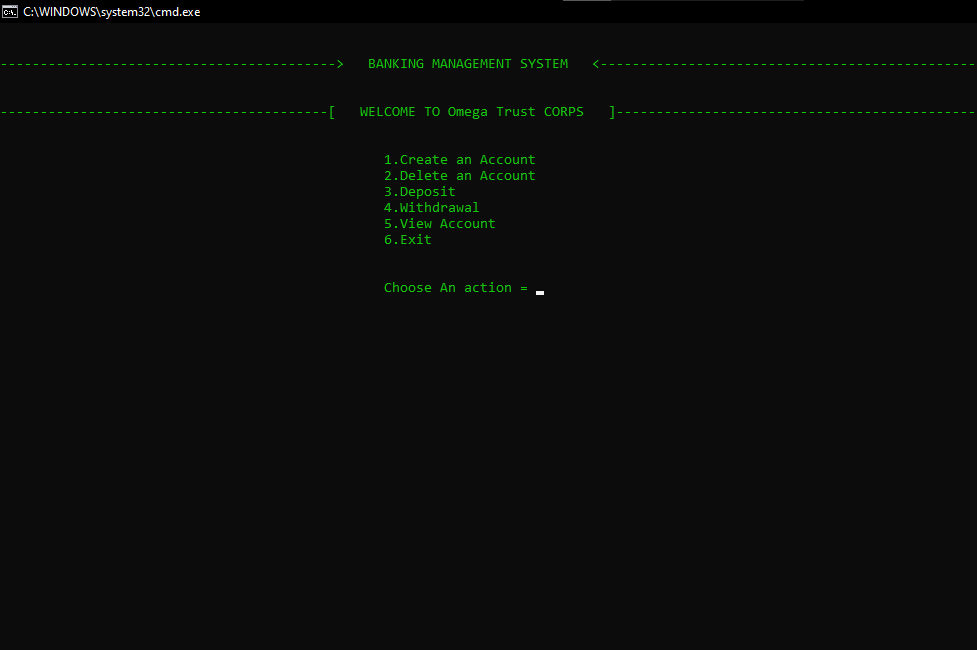
**DSU\_PROJECT();**

**return 0;**

**}**

**OUTPUTS:**

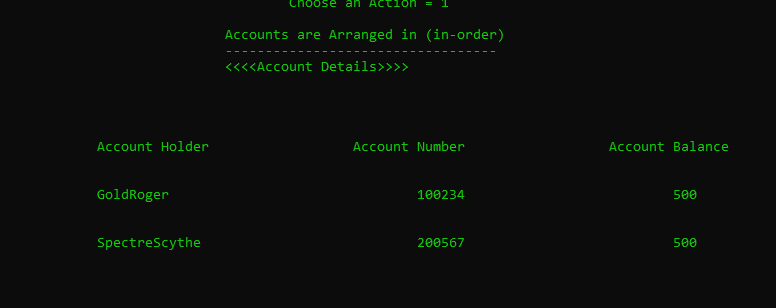
Bank Menu

****

Creating an account

****

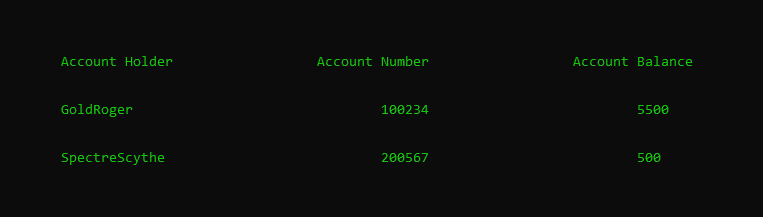
Creating another account and viewing them

****

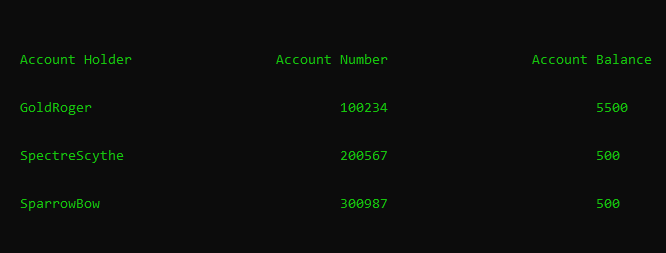
Depositing into GoldRoger Account

****

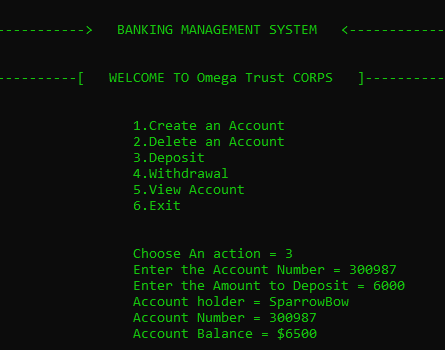
Viewing after depositing

****

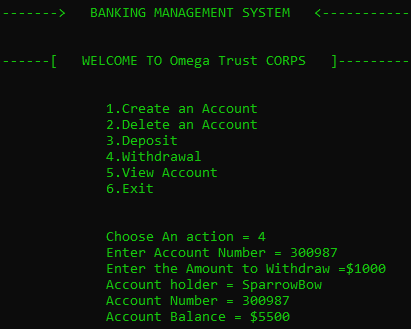
Creating a third account for withdrawing

****

Depositing into the third account

****

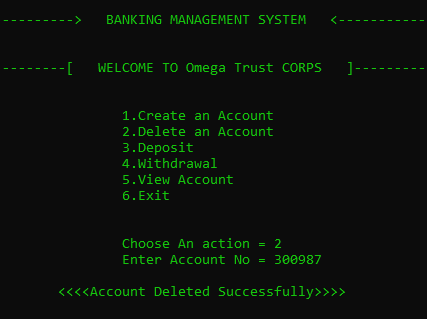
Withdrawing from the account

****

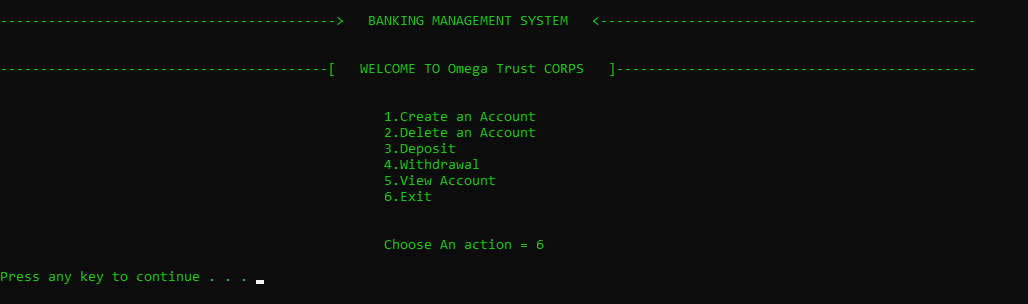
Viewing after withdrawing

****

Deleting the account

****

Exiting the program

****