|  |  |  |
| --- | --- | --- |
|  | **logo**  **JSPM’s**  **JAYAWANTRAO SAWANT POLYTECHNIC, Handewadi Road, Hadapsar, Pune-28**  **Department of Computer Engineering**  **Academic Year 2019-20** |  |

**MICRO PROJECT**

**TITLE OF THE PROJECT**

* **Implement banking operations like: Deposit, Withdraw and Balance Inquiry.**

**Program :** CO **Program code:** CO3I

**Course :** Data structure using C  **Course code:**22317

**Class :** SYCO1  **Group No:** Q/17

**Project Guide: Mr.S.S.Apune.**

****

**MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION**

**Certificate**

This is to certify that Mr Ashish Dhananjay Kadam of III Semester of Diploma in Computer Engineering of Institute Jayawantrao Sawant Polytechnic (Code: 0711) has completed the Micro Project satisfactorily in Subject – Data structure using C (22317) for the academic year 2019- 2020 as prescribed in the curriculum.

Place: Hadapsar, Pune.

Enrollment No: 1807110129

Date: ………..

Exam Seat No:- 214508

Subject Teacher Head of the Department Principal

* **MICRO PROJECT GROUP DETAILS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr No.** | **Roll No.** | **Name** | **Enrollment No.** | **Seat No.** |
| **1** | 21 | Ashish Dhananjay Kadam | 1807110129 | 214508 |

* **INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr No.** | **Content** | **Page No.** |
| **1** | Certificate | 1,2 |
| **2** | Group Details | 3 |
| **3** | Index | 4 |
| **4** | Annexure IA (part A) | 5 |
| **5** | Introduction and Program | 6-12 |
| **6** | Code | 13-19 |
| **7** | Output | 20-24 |
| **8** | Algorithm | 25 |
| **9** | Flowchart | 26 |
| **10** | Annexure II A (Part B) | 27-28 |
| **11** | Annexure IV (Teachers Evaluation Sheet) | 29-37. |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  | **JSPM’s**  **JAYAWANTRAO SAWANT POLYTECHNIC, Handewadi Road, Hadapsar, Pune-28**  **Department of Computer Engineering**  **Academic Year 2019-20** | **logo** |

* **Title of Micro project:** Implement banking operations like: Deposit, Withdraw and Balance Inquiry.

1. **Brief Introduction:** In this project, we have explained the introduction to Data structure using C , C program to implement banking operations like: Deposit, Withdraw and Balance Inquiry. And also stated the screenshots of the outputs of program. We have also stated algorithm and flowchart and also listed its applications in Data structure using c

**2.0 Aim of Micro Project:** The aim of the project is to **implement banking operations like: Deposit, Withdraw and Balance Inquiry.**

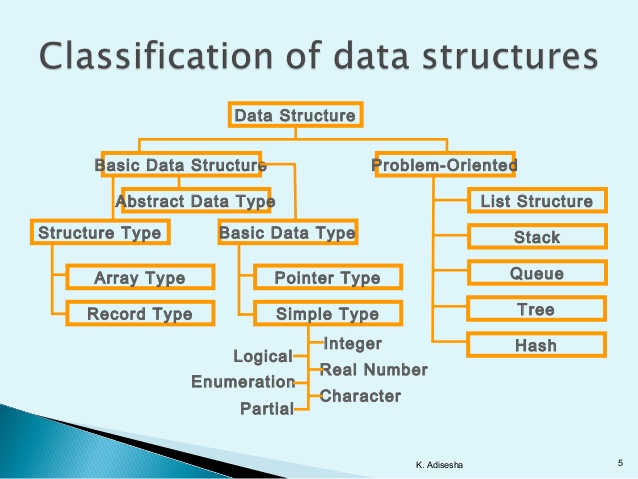
**3.0 Action Plan** (Sequence and time required for major activities for 8 week)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No** | **Details of activity** | **Planned start date** | **Planned Finish date** | **Name of Responsible Team members** |
| 1 | Collecting the information of view | 14/7/2019 | 14/7/2019 | Ashish kadam |
| 2 | Sorting the information of view | 28/7/2019 | 28/7/2019 |
| 3 | Compilation of the project | 11/8/2019 | 11/8/2019 |
| 4 | Submission of the project | 7/10/2019 | 7/10/2019 |

**4.0 Resources required** (major resources such as raw material, some machining facility, software etc.)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. NO** | **Name of resource / material** | **Specification** | **Quantity** | **Remarks** |
| 1 | Computer System | Windows 7 ultimate | 1 |  |
| 2 | pc | Windows 10 | 1 |  |
| 3 | Web | Chrome browser to acquire basic knowledge about computer graphics. | 1 |  |

**Data Structure in C**

****

**Data structures** are used to store data in a computer in an organized form. In C Programming Language Different types of data structures are; Array, Stack, Queue, Linked List, Tree. In term of computer programming language, a data structure may be selected or designed to store data for the purpose of working on it with various algorithms

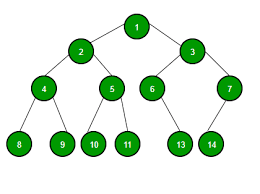
**Data Structure in C** Programming Language is a specialized format for organizing and storing data. In General data structure types include the file, array, record, table, tree... Etc.

* **Array:**Array is collection of similar data type, you can insert and deleted element form array without follow any order.
* **Stack:**Stack work on the basis of Last-In-First-Out (LIFO). Last entered element removed first.
* **Queue:**Queue work on the basis of First-In-First-Out (FIFO). First entered element removed first.
* **Linked List:**Linked list is the collection of node, Here you can insert and delete data in any order.
* **Tree:**Stores data in a nonlinear form with one root node and sub nodes.

**Algorithm**

An algorithm is a finite set instruction, which is written for solve any problem. Algorithm is not the complete code or program; it is just like an English language.

# Applications of tree data structure:

  
**Why Tree?**  
Unlike Array and Linked List, which are linear data structures, tree is hierarchical (or non-linear) data structure.

1. One reason to use trees might be because you want to store information that naturally forms a hierarchy. For example, the file system on a computer:

file system

———–

/ <-- root

/ \

... Home

/ \

upgrade course

/ / | \

... cs101 cs112 cs113

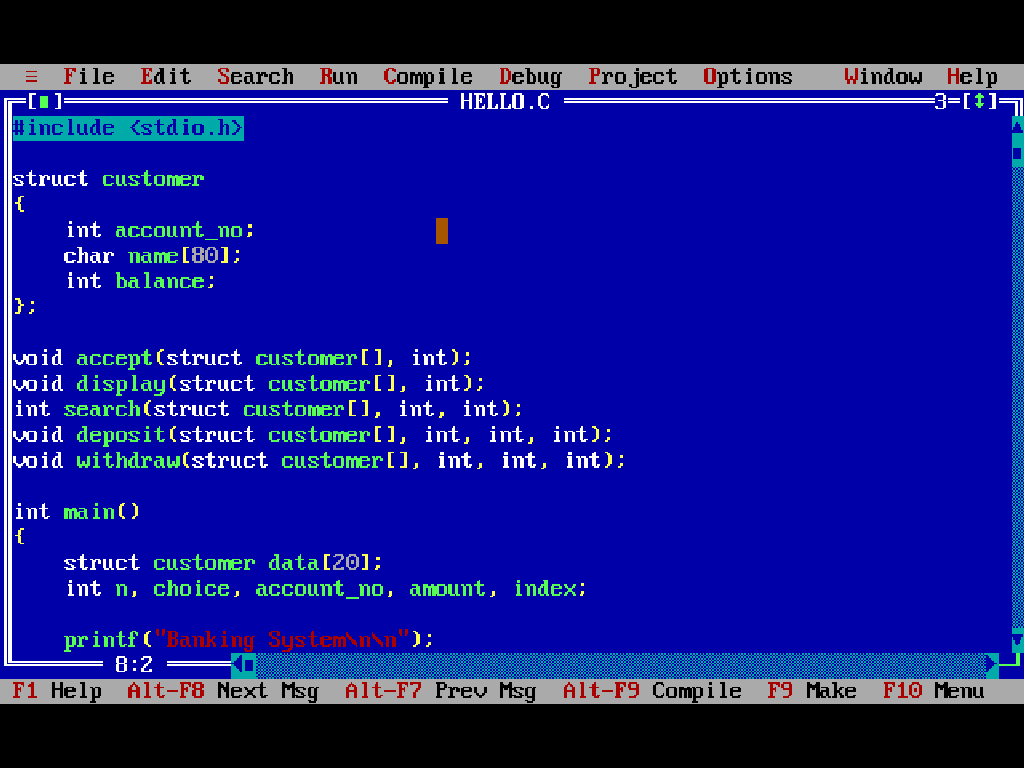
1. If we organize keys in form of a tree (with some ordering e.g., BST), we can search for a given key in moderate time (quicker than Linked List and slower than arrays). [Self-balancing search trees](http://en.wikipedia.org/wiki/Self-balancing_binary_search_tree)like [AVL](http://en.wikipedia.org/wiki/AVL_tree) and [Red-Black trees](http://en.wikipedia.org/wiki/Red-black_tree) guarantee an upper bound of O(Logn) for search.

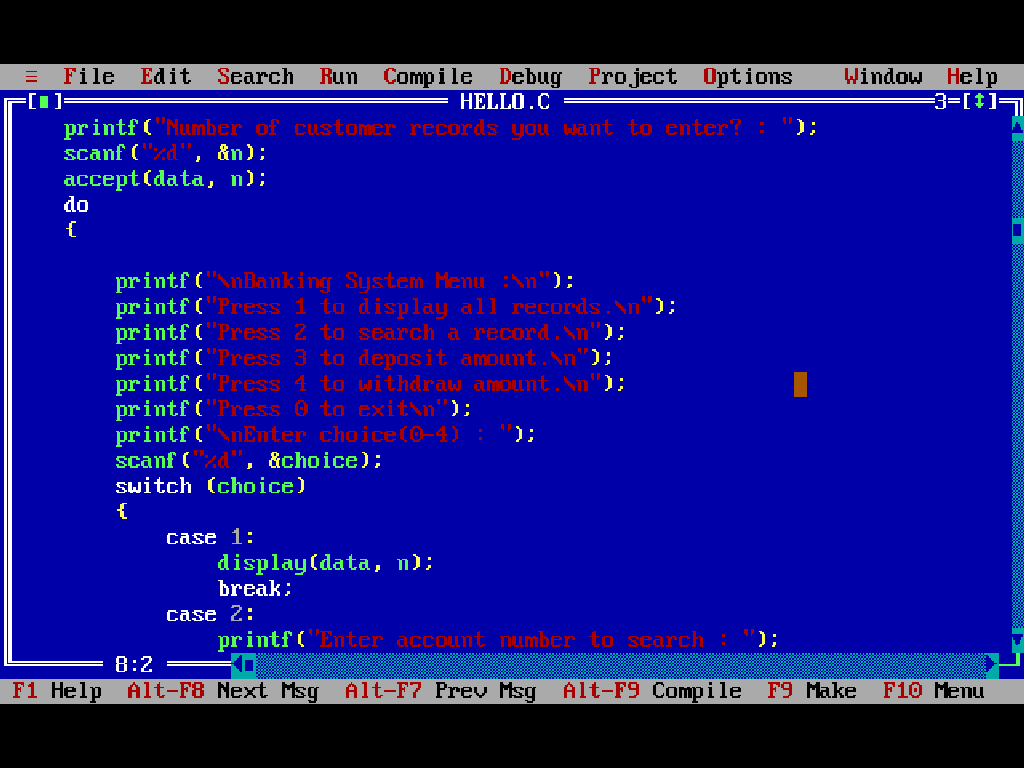


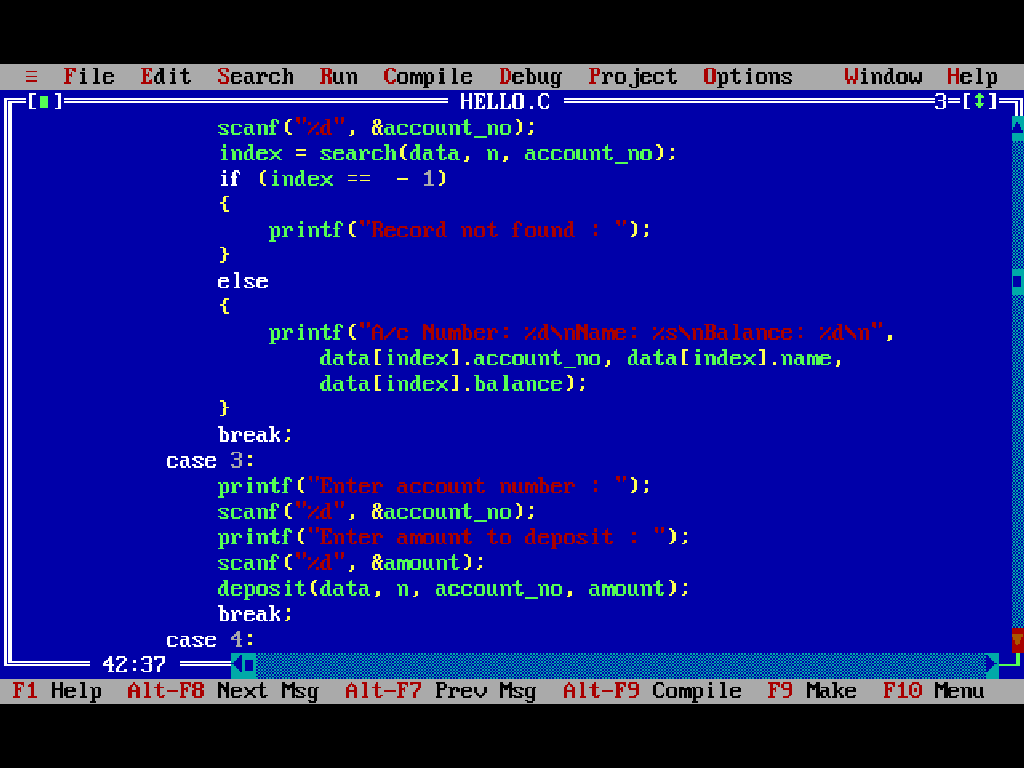
1. We can insert/delete keys in moderate time (quicker than Arrays and slower than Unordered Linked Lists). [Self-balancing search trees](http://en.wikipedia.org/wiki/Self-balancing_binary_search_tree)like [AVL](http://en.wikipedia.org/wiki/AVL_tree) and [Red-Black trees](http://en.wikipedia.org/wiki/Red-black_tree) guarantee an upper bound of O (Logn) for insertion/deletion.
2. Like Linked Lists and unlike Arrays, Pointer implementation of trees doesn’t have an upper limit on number of nodes as nodes are linked using pointers.

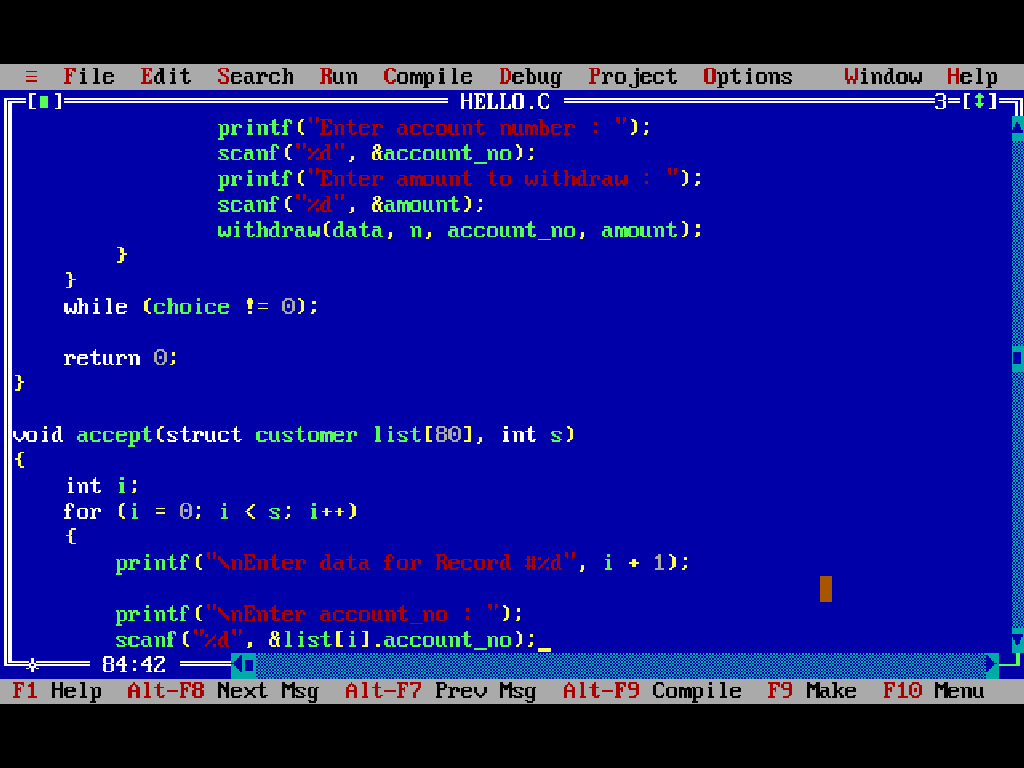
* **Program to implement banking operations like: Deposit, Withdraw and Balance Inquiry.**

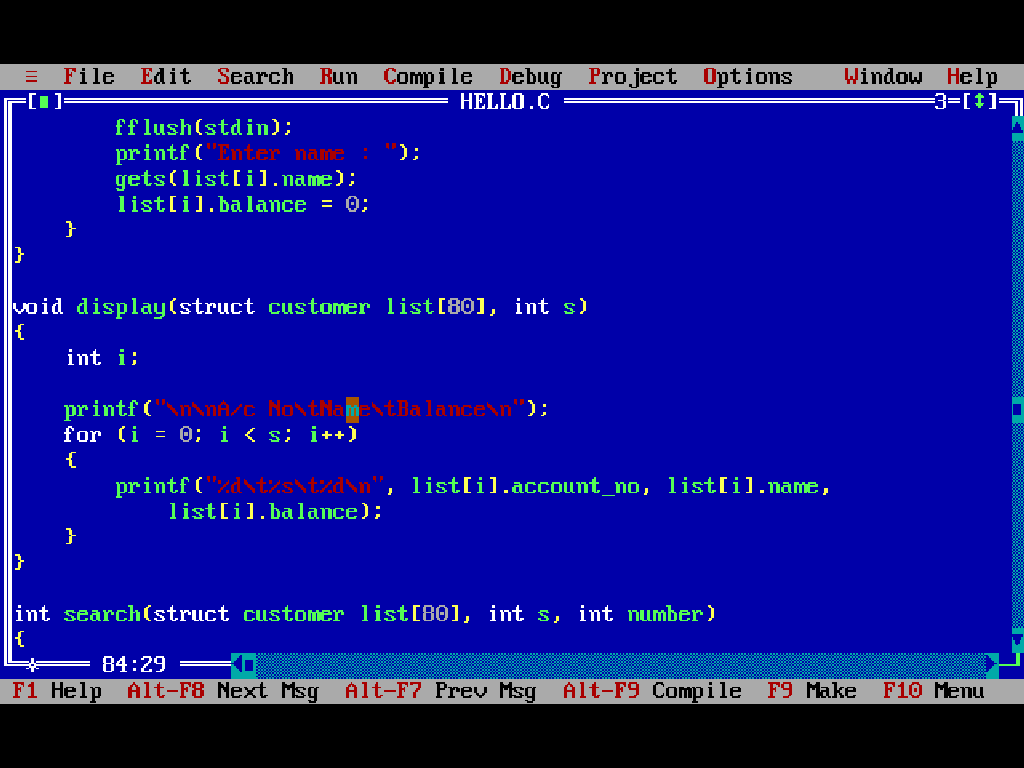
**Program:**

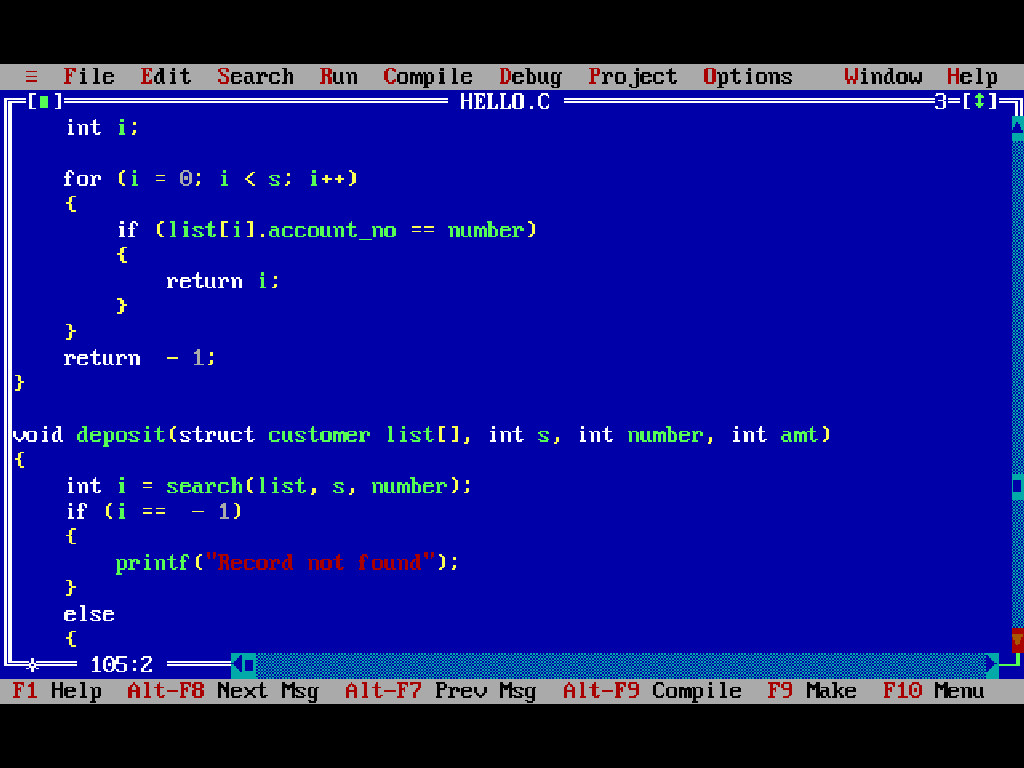


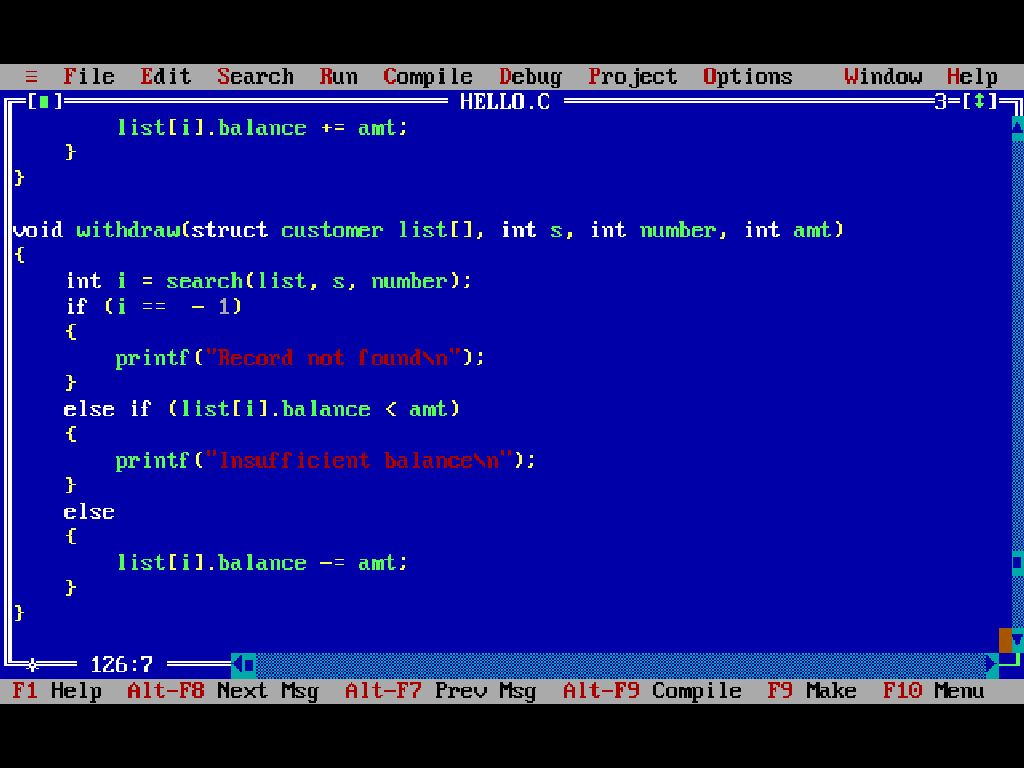












**Code:**

****

#include <stdio.h>

struct customer

{

intaccount\_no;

char name[80];

int balance;

};

void accept(struct customer[], int);

void display(struct customer[], int);

int search(struct customer[], int, int);

void deposit(struct customer[], int, int, int);

void withdraw(struct customer[], int, int, int);

int main()

{

struct customer data[20];

int n, choice, account\_no, amount, index;

printf("Banking System\n\n");

printf("Number of customer records you want to enter? : ");

scanf("%d", &n);

accept(data, n);

do

{

printf("\nBanking System Menu :\n");

printf("Press 1 to display all records.\n");

printf("Press 2 to search a record.\n");

printf("Press 3 to deposit amount.\n");

printf("Press 4 to withdraw amount.\n");

printf("Press 0 to exit\n");

printf("\nEnter choice(0-4) : ");

scanf("%d", &choice);

switch (choice)

{

case 1:

display(data, n);

break;

case 2:

printf("Enter account number to search : ");

scanf("%d", &account\_no);

index = search(data, n, account\_no);

if (index == - 1)

{

printf("Record not found : ");

}

else

{

printf("A/c Number: %d\nName: %s\nBalance: %d\n",

data[index].account\_no, data[index].name,

data[index].balance);

}

break;

case 3:

printf("Enter account number : ");

scanf("%d", &account\_no);

printf("Enter amount to deposit : ");

scanf("%d", &amount);

deposit(data, n, account\_no, amount);

break;

case 4:

printf("Enter account number : ");

scanf("%d", &account\_no);

printf("Enter amount to withdraw : ");

scanf("%d", &amount);

withdraw (data, n, account\_no, amount);

}

}

while (choice != 0);

return 0;

}

void accept(struct customer list[80], int s)

{

int i;

for (i = 0; i < s; i++)

{

printf("\nEnter data for Record #%d", i + 1);

printf("\nEnteraccount\_no : ");

scanf("%d", &list[i].account\_no);

fflush(stdin);

printf("Enter name : ");

gets(list[i].name);

list[i].balance = 0;

}

}

void display(struct customer list[80], int s)

{

int i;

printf("\n\nA/c No\tName\tBalance\n");

for (i = 0; i < s; i++)

{

printf("%d\t%s\t%d\n", list[i].account\_no, list[i].name,

list[i].balance);

}

}

int search(struct customer list[80], int s, int number)

{

int i;

for (i = 0; i < s; i++)

{

if (list[i].account\_no == number)

{

return i;

}

}

return - 1;

}

void deposit(struct customer list[], int s, int number, intamt)

{

int i = search(list, s, number);

if (i == - 1)

{

printf("Record not found");

}

else

{

list[i].balance += amt;

}

}

void withdraw(struct customer list[], int s, int number, intamt)

{

int i = search(list, s, number);

if (i == - 1)

{

printf("Record not found\n");

}

else if (list[i].balance <amt)

{

printf("Insufficient balance\n");

}

else

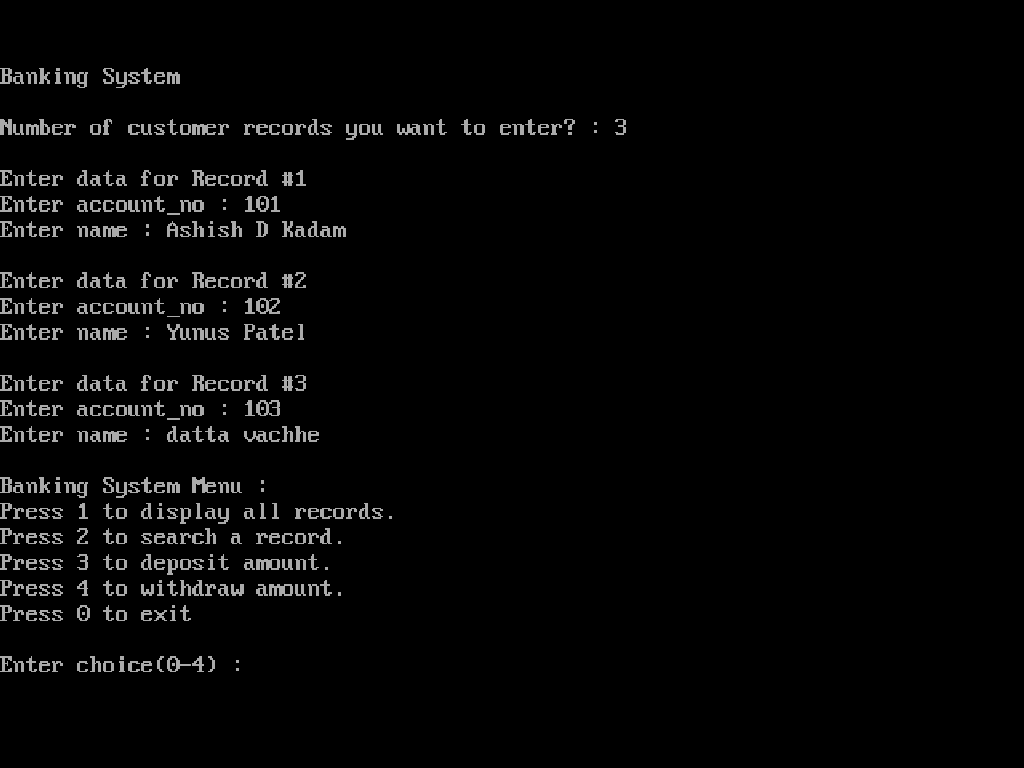
{

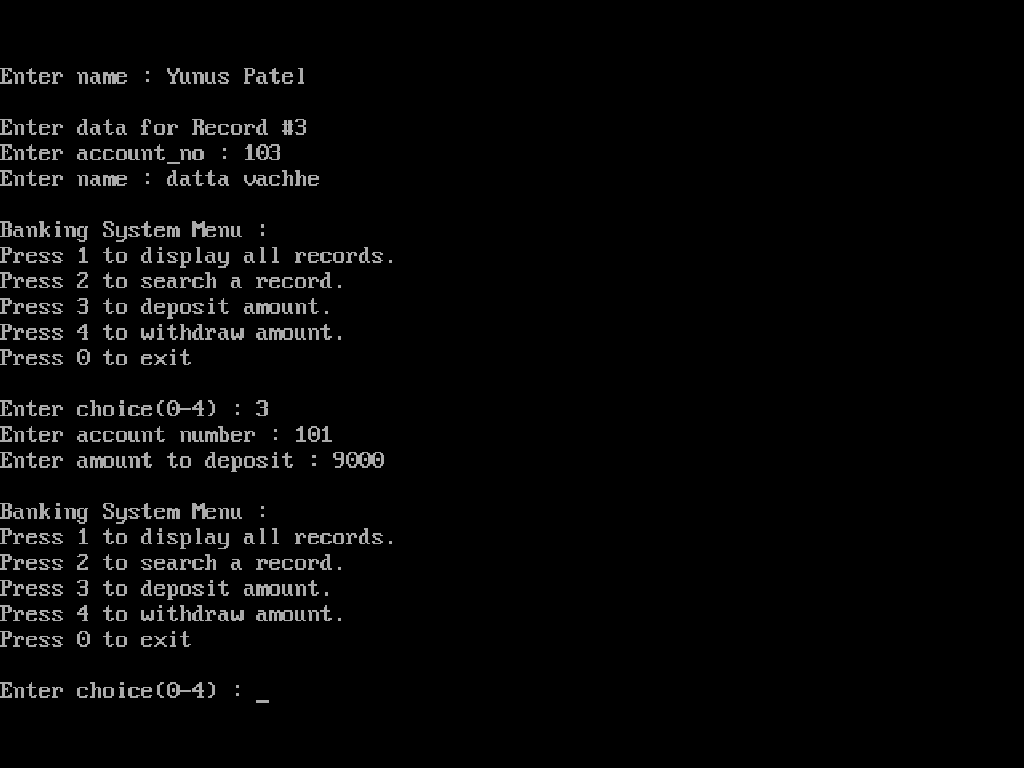
list[i].balance -= amt;

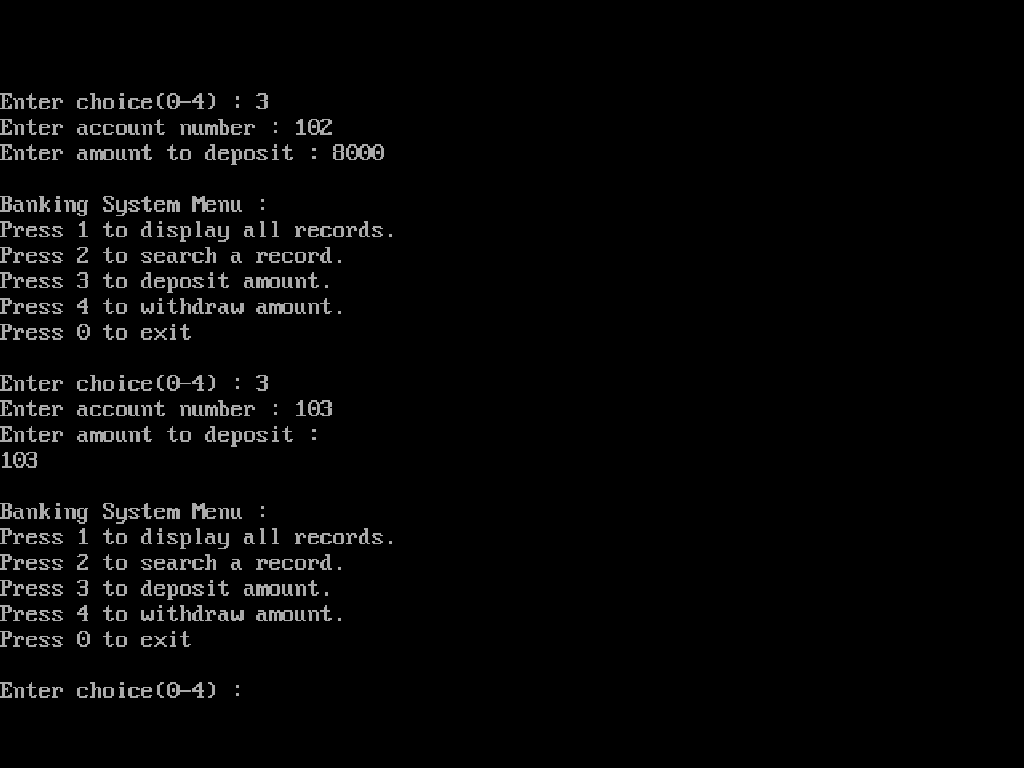
}

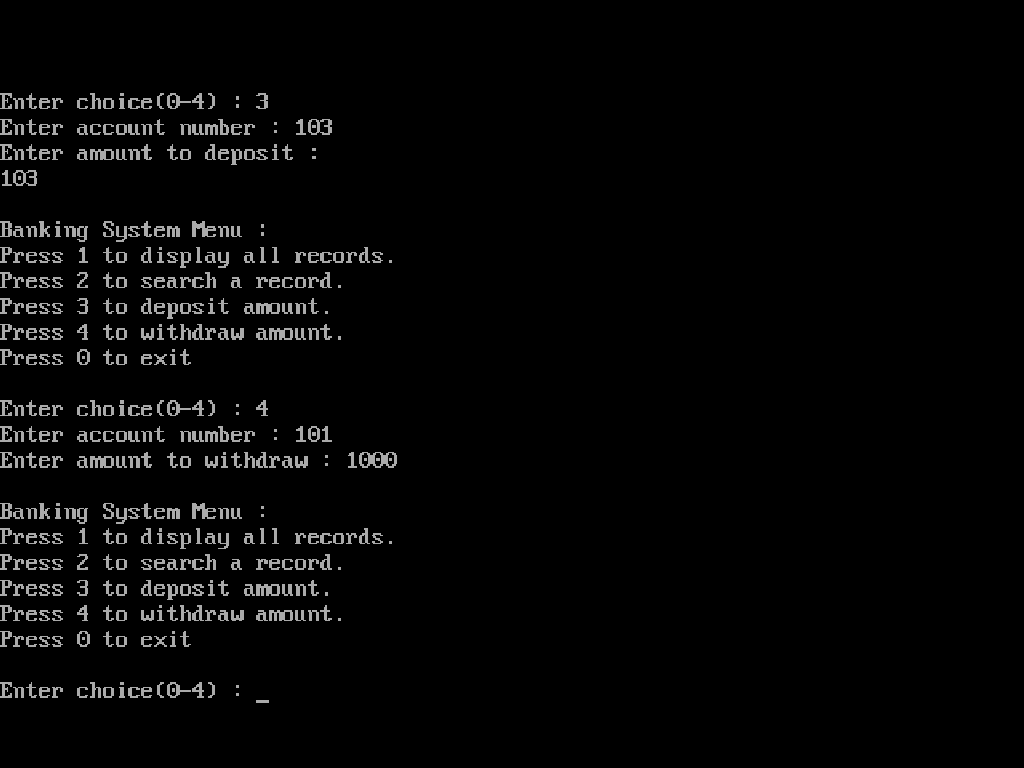
}

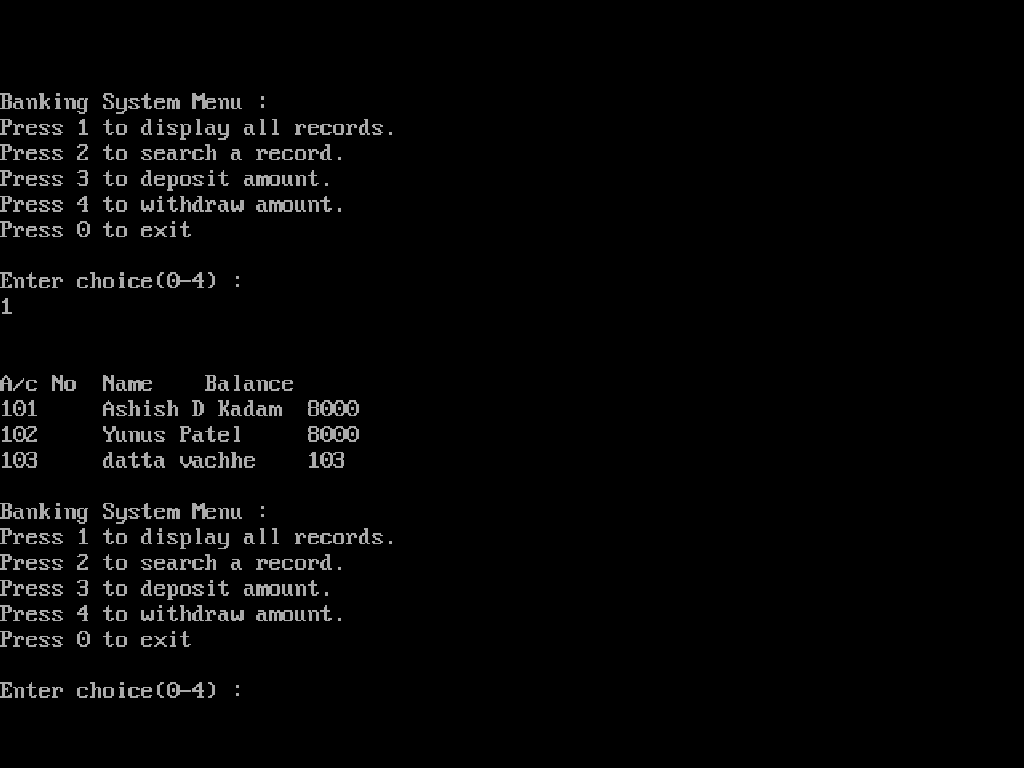
**Output:**











* **Algorithm:**

Step 1: Create structure for customer.

Step 2: Enter customer records number.

Step 3: Call accept function.

Step 4: Enter account number from customer.

Step 5: Select choice.

If: 1--> Display record.

2--> to search a record.

3--> to deposit amount.

4--> Withdraw amount.

5-->to exit.

Step 6: Switch choice

Case 1: Display (data)

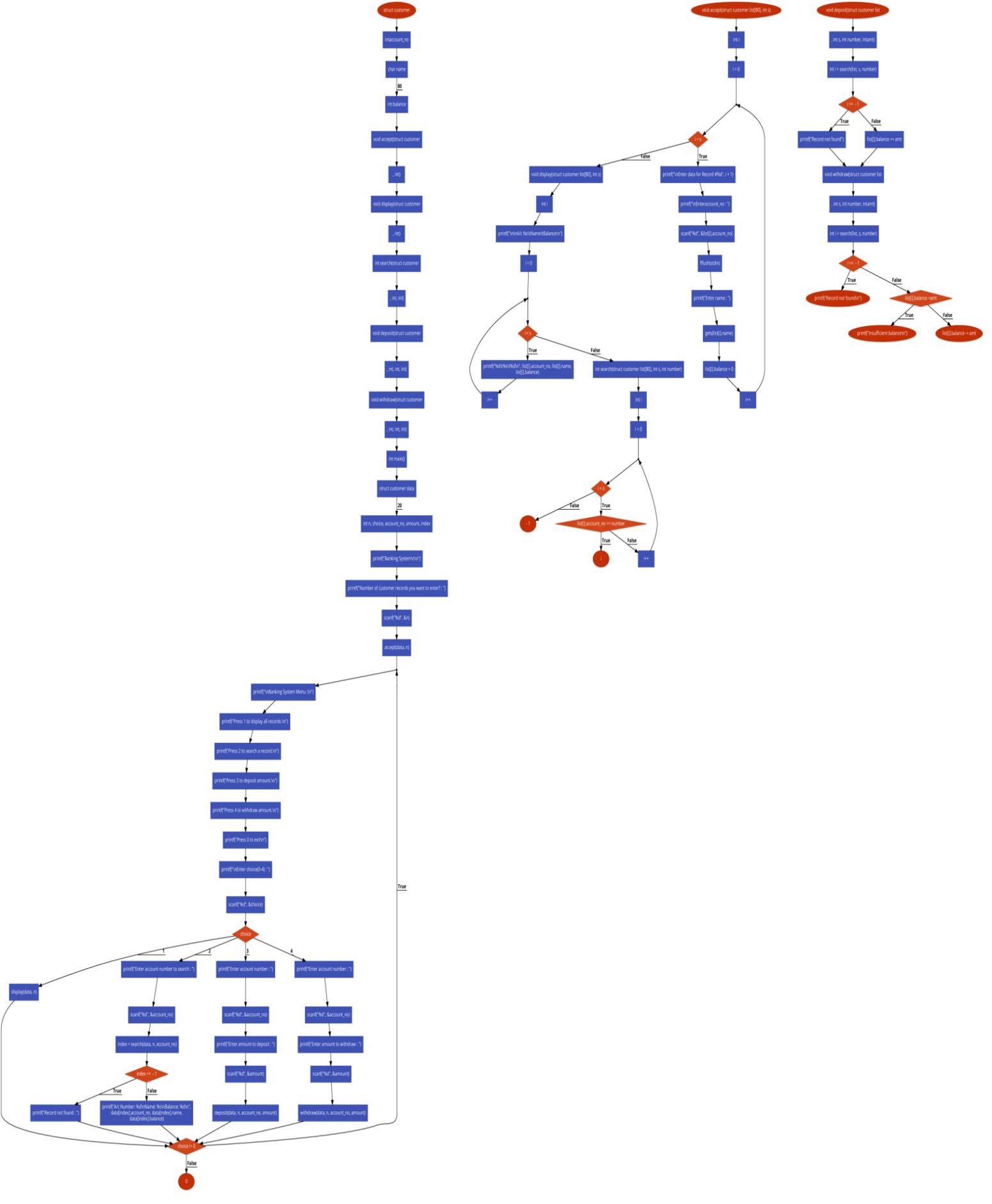
Case 2: Account number to search data

Case 3: Enter account number to deposit amount.

Case 4: Enter account number to withdraw amount.

Step 7: Stop

* **Flowchart:**



|  |  |  |
| --- | --- | --- |
|  | **JSPM’s**  **JAYAWANTRAO SAWANT POLYTECHNIC, Handewadi Road, Hadapsar, Pune-28**  **Department of Computer Engineering**  **Academic Year 2019-20** | **logo** |

* **Title of Micro project: Implement banking operations like: Deposit, Withdraw and Balance Inquiry.**

**1.0 Brief Description:** In this project, we have explained the introduction to Data structure using c in C program to implement banking operations like: Deposit, Withdraw and Balance Inquiry. And also stated the screenshots of the outputs of program. We have also stated algorithm and flowchart and also listed its applications in Data structure using c

**2.0 Aim of micro Project**: Implement banking operations like: Deposit, Withdraw and Balance Inquiry.

**3.0Course Outcome Integrated: To learn and understand the basic knowledge of Data structure using c**

**4.0Actual Procedure Followed: 1. Created and gathered information and sorted it in order.**

**2. Understood the concepts and planned the dates to do it.**

**3. Executed the plan properly whole gaining knowledge and discipline.**

**5.0 Actual Resources Used** (mention the actual resources used.)

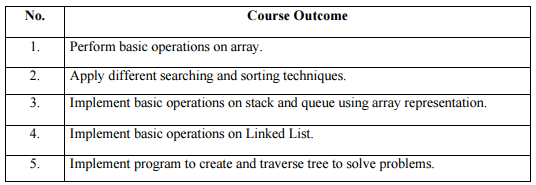
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. NO** | **Name of resource / material** | **Specification** | **Quantity** | **Remarks** |
| 1 | Computer System | Windows 7 ultimate. | 1 |  |
| 2 | Mobile | Samsung,lg,oppo. | 1 |  |
| 3 | Web | Chrome browser to acquire basic knowledge about Data structure using c | 1 |  |

**6.0 Outputs of the Micro Projects: Thus we have studied and under stood all the basic concepts and functions in Data Structure using C. Outputs of projects have attached at respective places.**

**7.0 Skill Developed/ Learning out of this Micro project**

Throughout this project we developed some important skills like leadership quality, scheduling of the project and risk management. One of the most important skills we developed is the communication and coordination between our team members. We also learned planning skills, time management and adaptability.

* **Teacher Evaluation Sheet**
* **Name of students:** Ashish Dhananjay Kadam.
* **Enrollment No**: 1807110129.
* **Name of program:** Computer Engineering.
* **Semester**: Third.
* **Course Title**: Data structure using C.
* **Code: 22317**
* **Title of Micro Project**: Implement banking operations like: Deposit, Withdraw and Balance Inquiry.
* **Course Outcomes Achieved:**



* **Evaluation as per suggested Rubric for Assessment of Micro Project**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr. No** | **Characteristic to be assessed** | **Poor**  (Marks1- 3) | **Average**  (Marks 4 -5) | **Good**  (Marks 6 -8) | **Excellent**  (Marks 9 - 10) |
| 1 | Relevance to the course |  |  |  |  |
| 2 | Literature Survey / Information collection |  |  |  |  |
| 3 | Project Proposal |  |  |  |  |
| 4 | Completion of the Target as per Project Proposal |  |  |  |  |
| 5 | Analysis of data and representation |  |  |  |  |
| 6 | Quality of Prototype/ Model |  |  |  |  |
| 7 | Report preparation |  |  |  |  |
| 8 | Presentation |  |  |  |  |
| 9 | Defence |  |  |  |  |

* **Micro Project Evaluation Sheet**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Process Assessment** | | **Product Assessment** | | **Total Marks**  **10** |
| **Part A - Project Proposal**  **(2 Marks)** | **Project Methodology**  **(2 Marks)** | **Par B - Project Report/ working Model**  **(2 Marks)** | **Individual Presentation/ Viva**  **(4 Marks)** |
|  |  |  |  |  |

**Note: Every course teacher is expected to assign marks for group evaluation in first 3 columns and individual evaluation 4th column**

**Comment/ suggestion about team work/leadership/ interpersonal communication (If any)**

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………

**Any other comment:**

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

**Name and Designation of the Faculty Member: Mr.S.S.Apune.**

**Signature: …………………………**