

NPI000150-ICP assignment.

by Jiten Chapagain

Submission date: 30-Jun-2022 06:44AM (UTC+0800)

Submission ID: 1864736618

File name: NPI000150-ICP_assignment.docx (1.92M)

Word count: 2933

Character count: 15110

Abstraction

This project is based on covid-19 donation management system furthermore, program is based on c-programming. The main aim of this covid-19 donation management system is to manage and keep record of the medical equipment which are received and distribute by the Malaysia Red Crescent society. In this report you will find the program with the flowchart and pseudocode.

It was pleasure to design the program with the help of different c-programming idea. With the help of, function of c-programming the task has been break in the different stages while doing coding. In this project I had used the main c-programming concept of structure, array, pointer, variable, loop etc.

Acknowledgement

This document report had been created with a lot of assist from many source and people like friends and our subject teacher Mr. Sushi Adhikari. Along with module teacher and friends, I would like to give sincere respect to APU, Infamax collage for giving me opportunity to prepare this report and display my knowledge and idea that I have gain in this collage through this medium.

1. Introduction and Assumption

The program is based on the Red Crescent Society, a non-profit organization based in Malaysia, which stores and distributes donations related to COVID-19. This program is console and console based and has no graphic display or user interface. The main goal of this program is to provide Red Crescent with a clean, simple and fully functional donation management service. The program can add new donation records, add donations to existing records, view donation records, search for donation records, update or save distribution records, and more. This program is only accessible to company administrators. Despite its features, this program has many drawbacks and limitations. These are described in System Limitations at the bottom of the document.

For this application admin is the authorized person to manage all donation record.

2. Project design

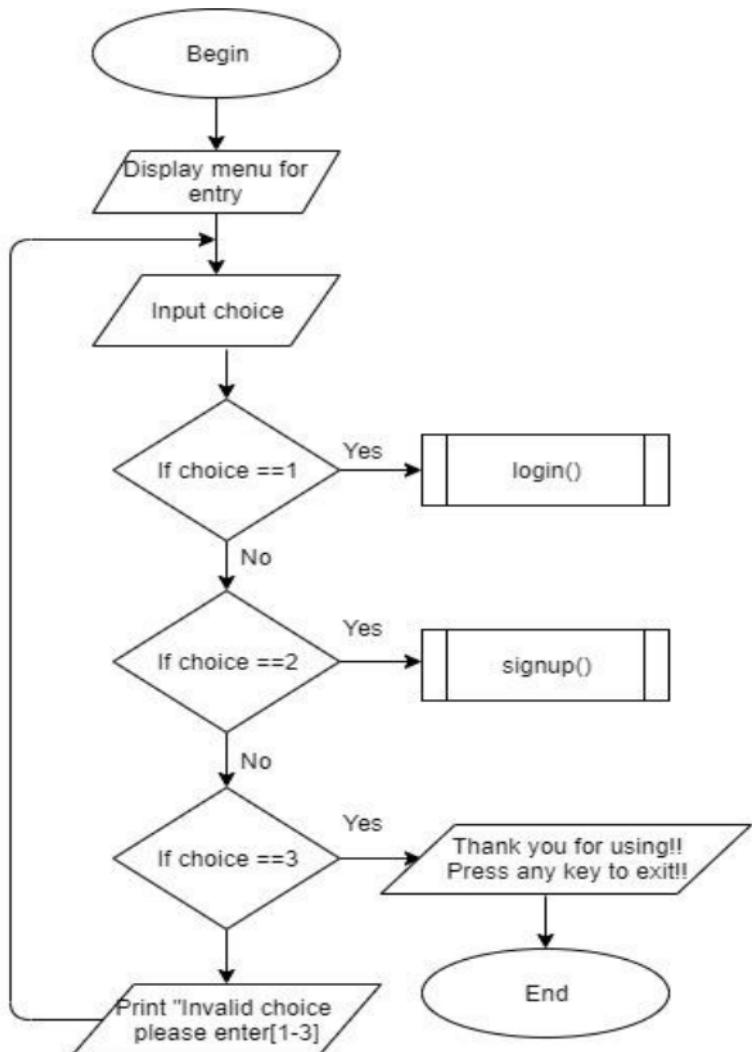
Pseudocode and flowchart are the important thing to design the project for doing coding. The explanation of line by line code from the program is called pseudocode. Furthermore, the frame representation of pseudocode and algorithm is known as flowchart. The given below pseudocode and flowchart are the main layer of the program.

2.1 Main

Pseudocode:

1. Begin
2. Display main entry choice for the donation management system.
3. Input choice.
4. Switch the choice entered.
 - If choice ==1 then,
 go to login
 - If choice ==2 then,
 go to signup,
 - If choice ==3 then ,
 exit the program,
 - Else
 print “Invalid choice, Please enter [1-3]”,
 Return to step 3,
 End if,
5. Call mainmenu.
6. End.

Flowchart:

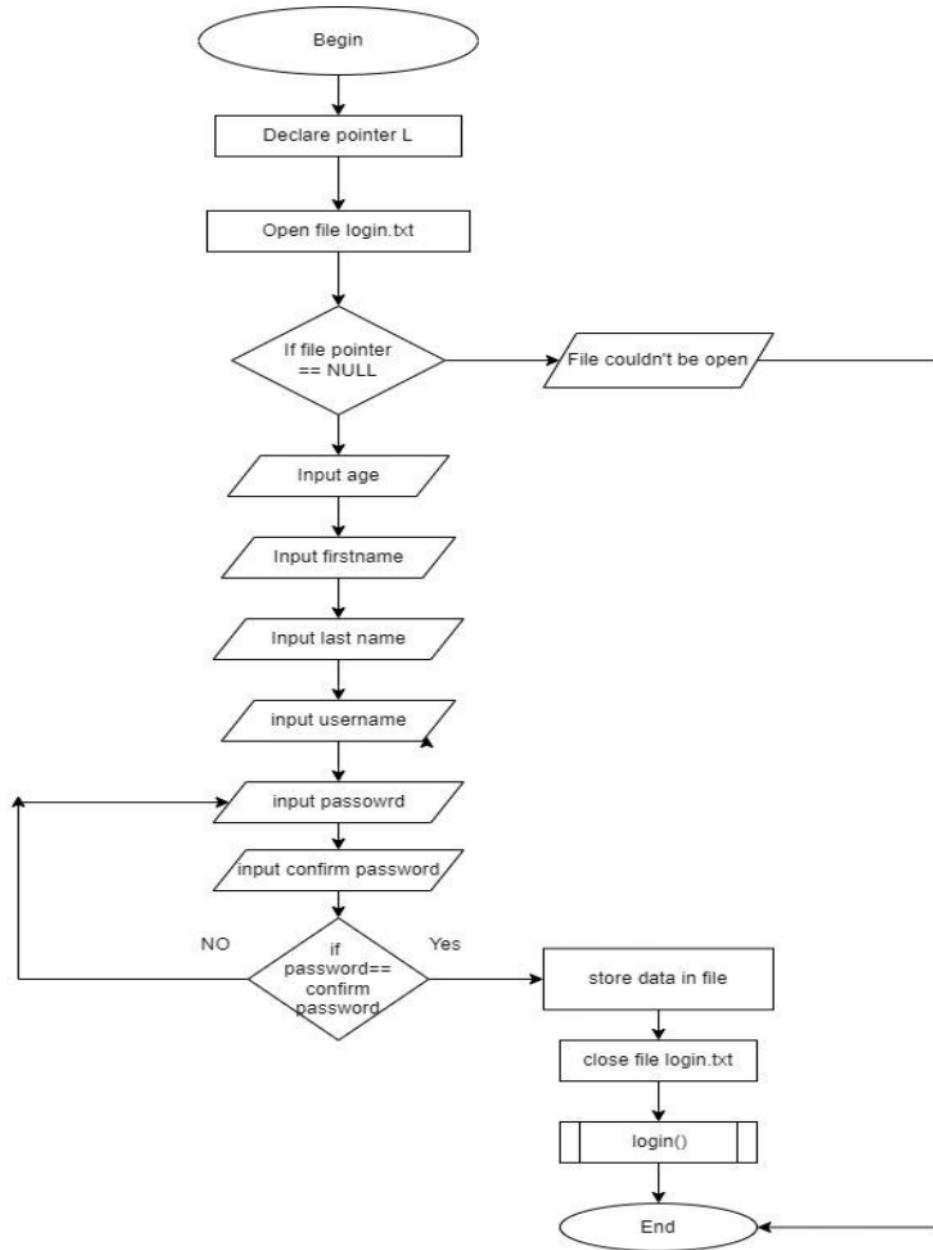


2.2 Signup

Pseudocode:

1. Begin
2. Declare file pointer L.
3. Open file login.txt in append mode.
4. If file pointer == NULL then,
 Print "File couldn't be opened"
End if
5. Enter Age, First name, Last name, Username.
6. Enter password, confirm password.
7. If password==confirm password then,
 Print "account register"
Else Return to step 6.
End if
8. Store data in file.
9. Close file.
10. Call login ().
11. End.

Flowchart:

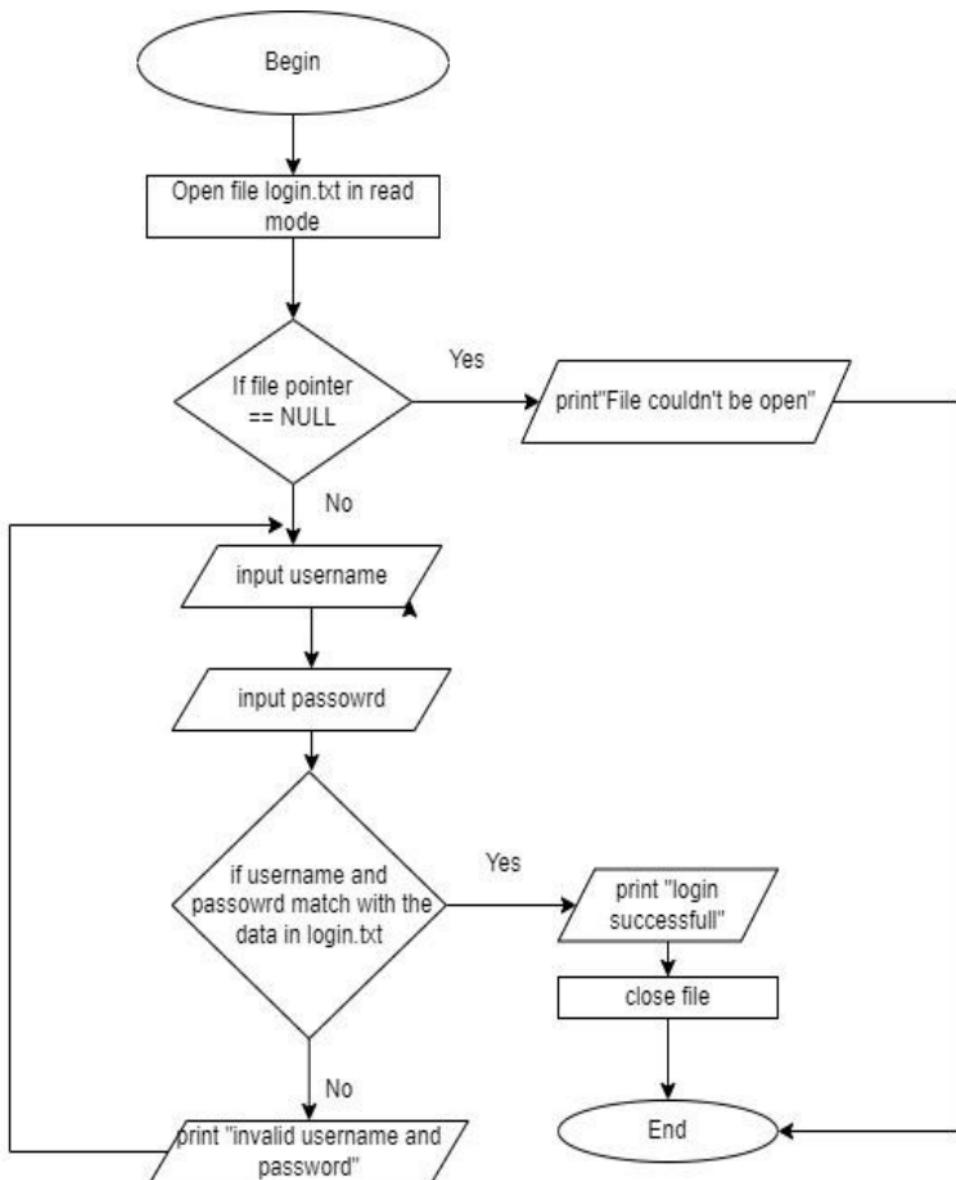


2.3 Login

Pseudocode:

1. Begin
2. Open file login.txt in read mode.
3. If file == NULL then,
 Print "File could not be opened"
 End if
4. Input username and password.
5. If username and password match with data in login.txt
 Then, print "login successful"
 Else
 Print "invalid username or password"
 Return to step 4.
 End if
6. End

Flowchart:

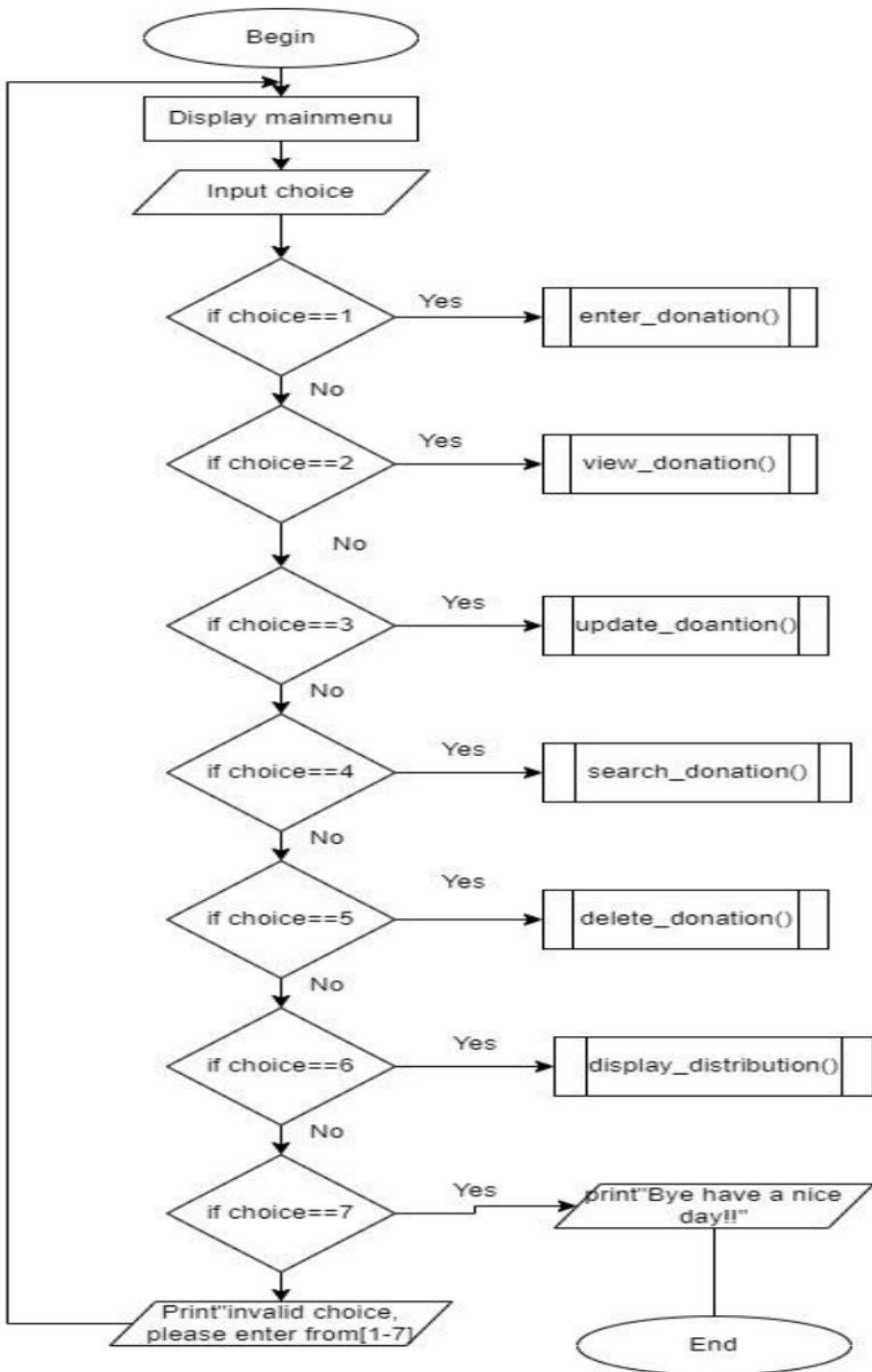


2.4 Mainmenu

Pseudocode:

```
1. Start
2. Read choice
3. Display mainmenu.
4. Input choice.
5. Switch choice
    If choice=1
        Goto enter_donation() function.
    If choice=2
        Goto view_donation() function.
    If choice=3
        Goto update_donation() function.
    If choice=4
        Goto search_donation() function.
    If choice=5
        Goto delete_donation() function.
    If choice=6
        Goto display_distribution() function.
    If choice=7
        Exit the function.
        Print "Bye Have a nice day".
    Else
        Print "Invalid choice, Please enter from [1-7]".
        Goto step 1.
    End if
6. End
```

Flowchart:

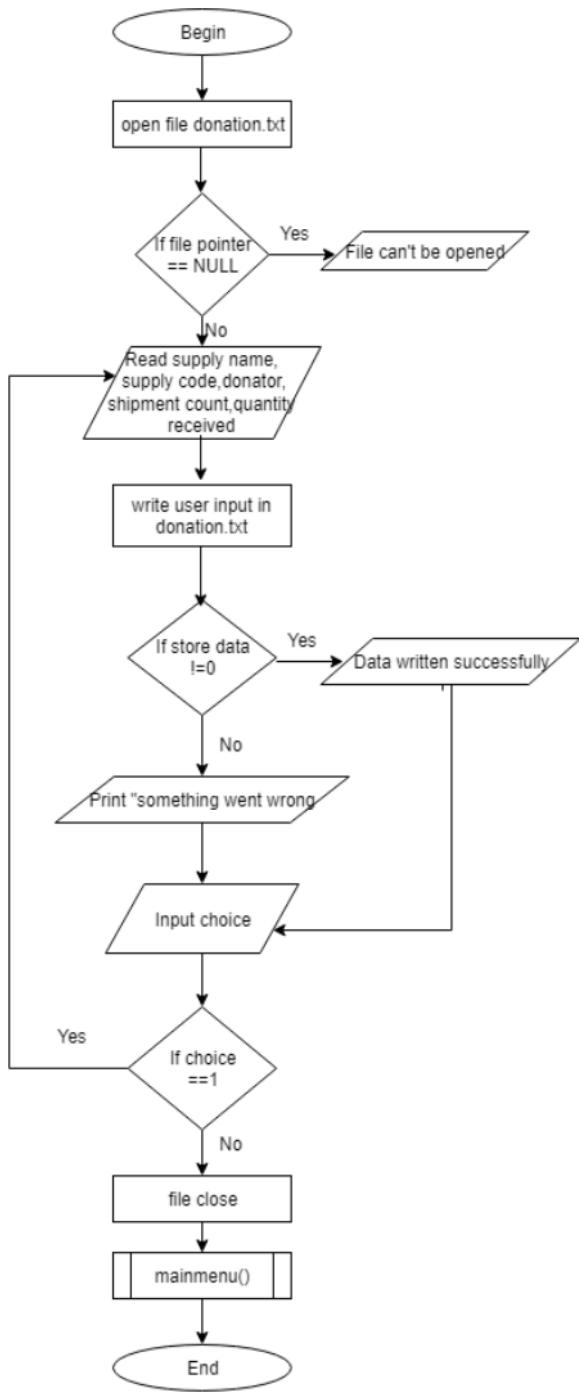


2.5 Enter donation

Pseudocode:

1. Begin
2. Declare file pointer L2.
3. Open file donation.txt in append mode.
4. If file pointer==NULL
 Print "file can't be opened"
Else
5. Start for loop.
 Enter supply name, supply code, donator, shipment count, quantity received.
 Store data in donation.txt file
 If (store data! =0) Then,
 Print "Data written successfully".
 Else
 Print "something went wrong".
End if
- Input choice
- If choice==1
 Goto step 5.
- Else
 Return.
6. End for loop.
7. Close file.
8. Call mainmenu () function.
9. End

Flowchart:

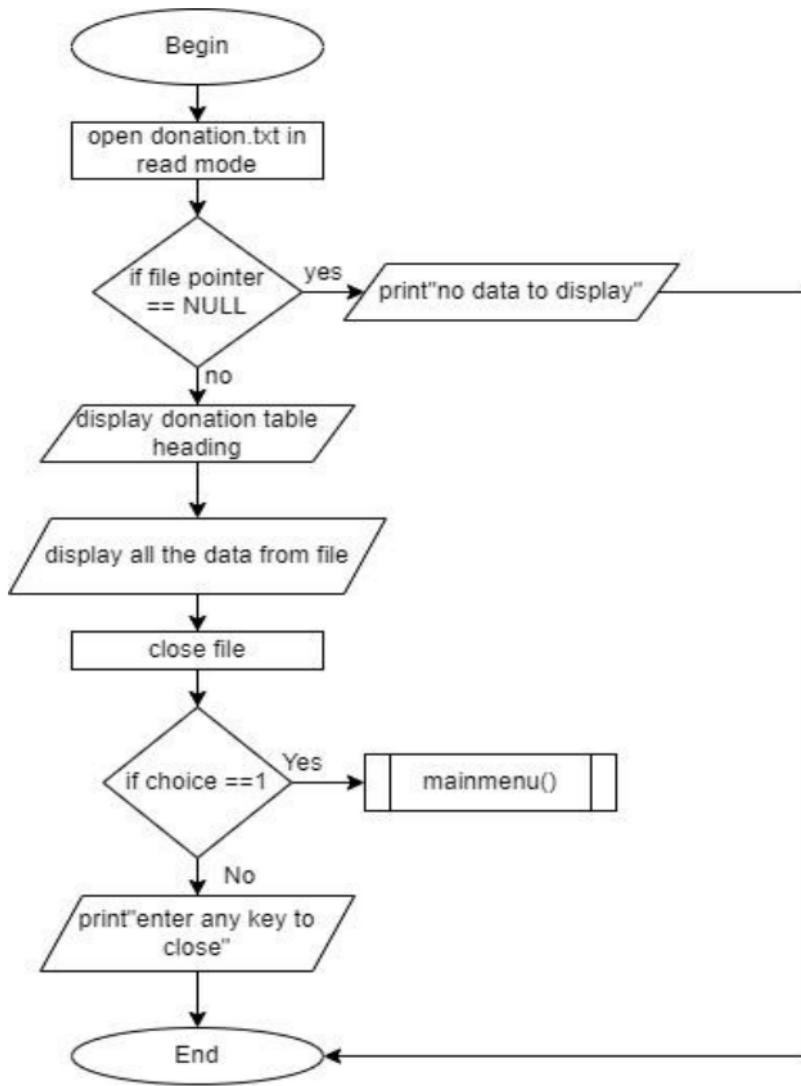


2.6 View donation

Pseudocode:

1. Begin
2. Declare file pointer L2.
3. Open file donation.txt in read mode.
4. If file pointer = NULL
 Print "No data to display"
- End if
5. Show donation table heading.
6. Read all donation from file.
7. Display all donation.
8. Close file.
9. Input choice.
10. If choice==1
 Call mainmenu () function.
- Else
 Print "Enter any key to close".
- Return.
- End if
11. End

Flowchart:

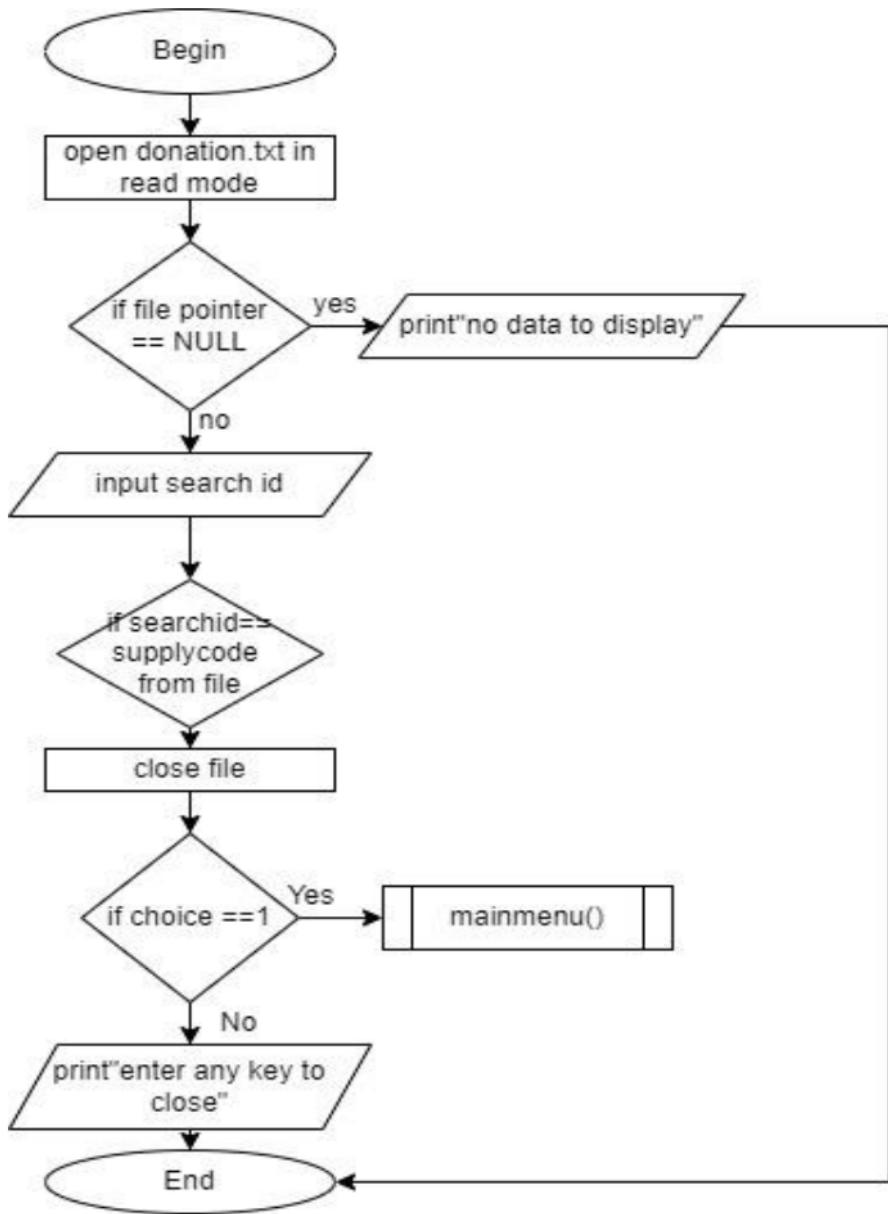


2.7 Search donation

Pseudocode:

1. Begin.
2. Declare file pointer L2.
3. Open file donation.txt in read mode.
4. If file pointer== NULL
 Print "No data to display".
 End if
5. Read search id to be searched.
6. Read supply code from file.
7. If search id == supply id
 Display all data of supply id.
 Close file.
 End if
8. Input choice
9. If choice==1.
 Got mainmenu () function.
 Else
 Print "Enter any key to close".
 Return.
10. End

Flowchart:

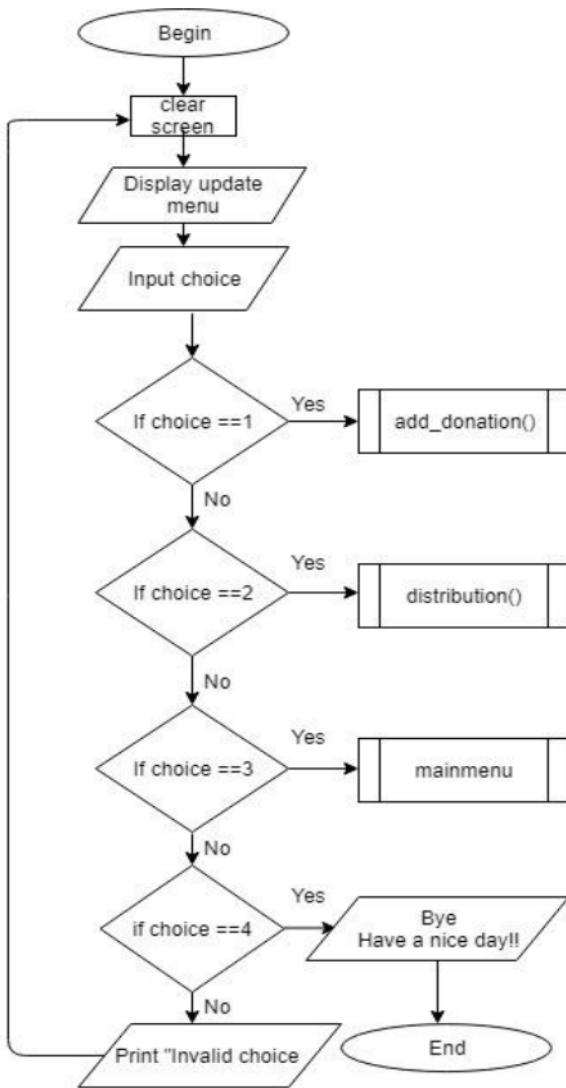


2.8 Update donation

Pseudocode:

```
1. Begin
2. Clear screen
3. Display update menu.
4. Input choice.
5. Switch the choice.
   If choice ==1 then,
      Goto add_donation ()�
   If choice ==2 then,
      Goto distribution ()�
   If choice ==3 then,
      Goto mainmenu ()�
   If choice ==4 then,
      Print "Bye Have a nice day"
   Else
      Print "Invalid input"
      Return to step 2.
End if.
6. End
```

Flowchart:

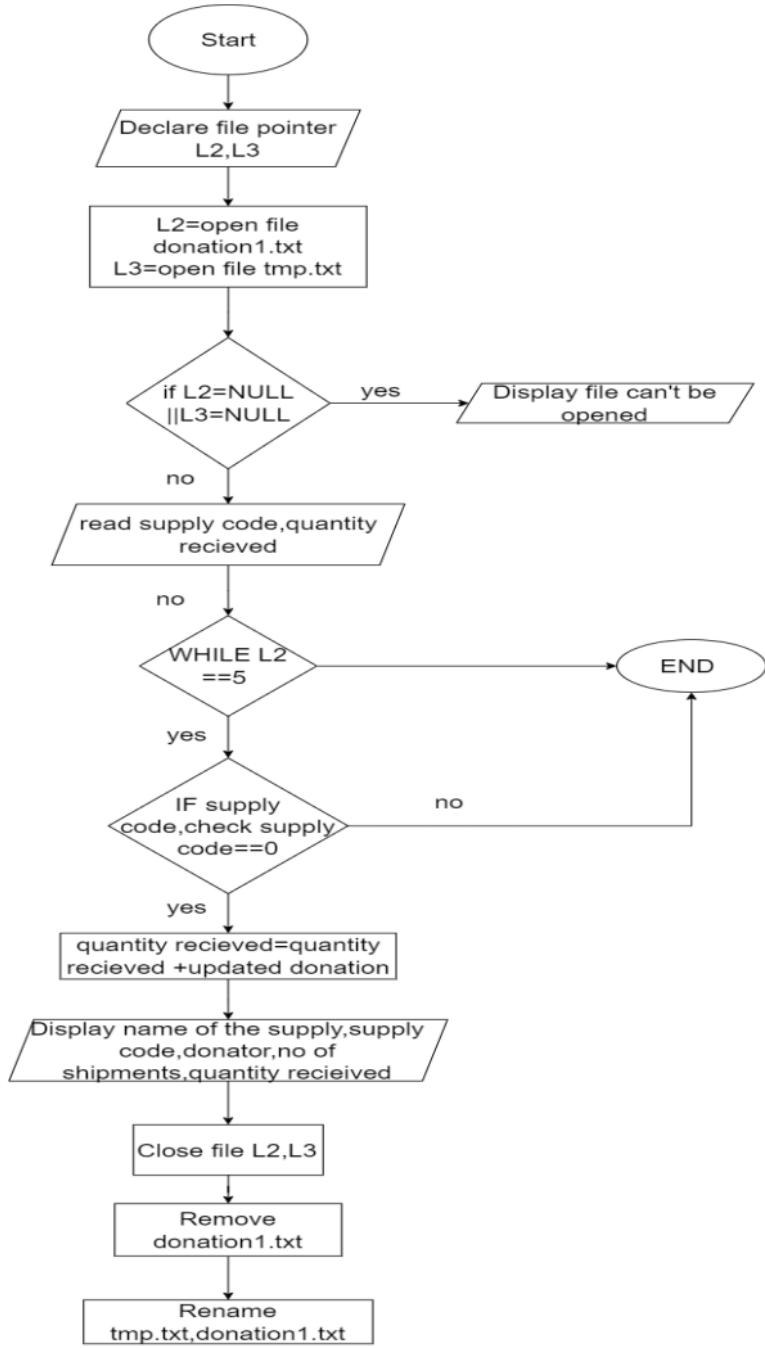


2.9 Add donation

Pseudocode:

1. Begin
2. Declare pointer L2, L3.
3. Open file donation.txt in read mode.
4. Open file tmp.txt in append mode.
5. If L2== NULL or L3 == NULL
 Print “File can’t be opened”
 End if
6. While L2==5.
7. Read check supply code and updated quantity.
8. Read supply code from file.
 If supply code == check supply code
 Quantity received= quantity received +updated donation.
 Store data in fp.
 Close L2.
 Close L3.
9. Remove donation.txt
10. Rename tmp.txt, donation.txt
11. Call update donation () .
12. End

Flowchart:

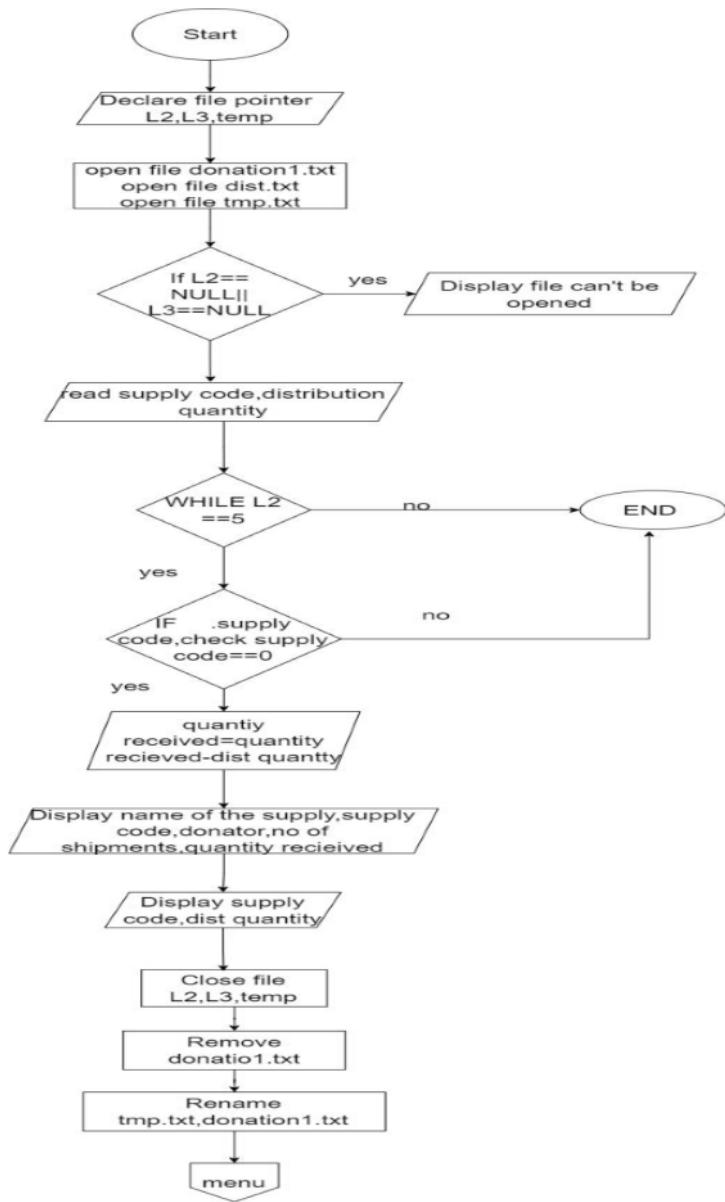


2.10 Distribution

Pseudocode:

1. Begin
2. Declare pointer L2, L3 and tmp.
3. Open file donation.txt in read mode.
4. Open file dist.txt in append mode.
5. Open file tmp.txt in append mode.
6. If L2== NULL or tmp == NULL
 Print "File can't be opened"
 End if
7. While L2==5
 Read check supply code and dist quantity.
 Read supply code from file.
 If supply code == check supply code
 Quantity received= quantity received -dist quantity
 Store data in tmp.
 Store supply name, supply code, dist. quantity in L3.
 Close file L3.
 Close L2.
 End loop.
8. Close tmp.
9. Remove donation.txt
10. Rename tmp.txt, donation.txt
11. Call mainmenu ().
12. End

Flowchart:

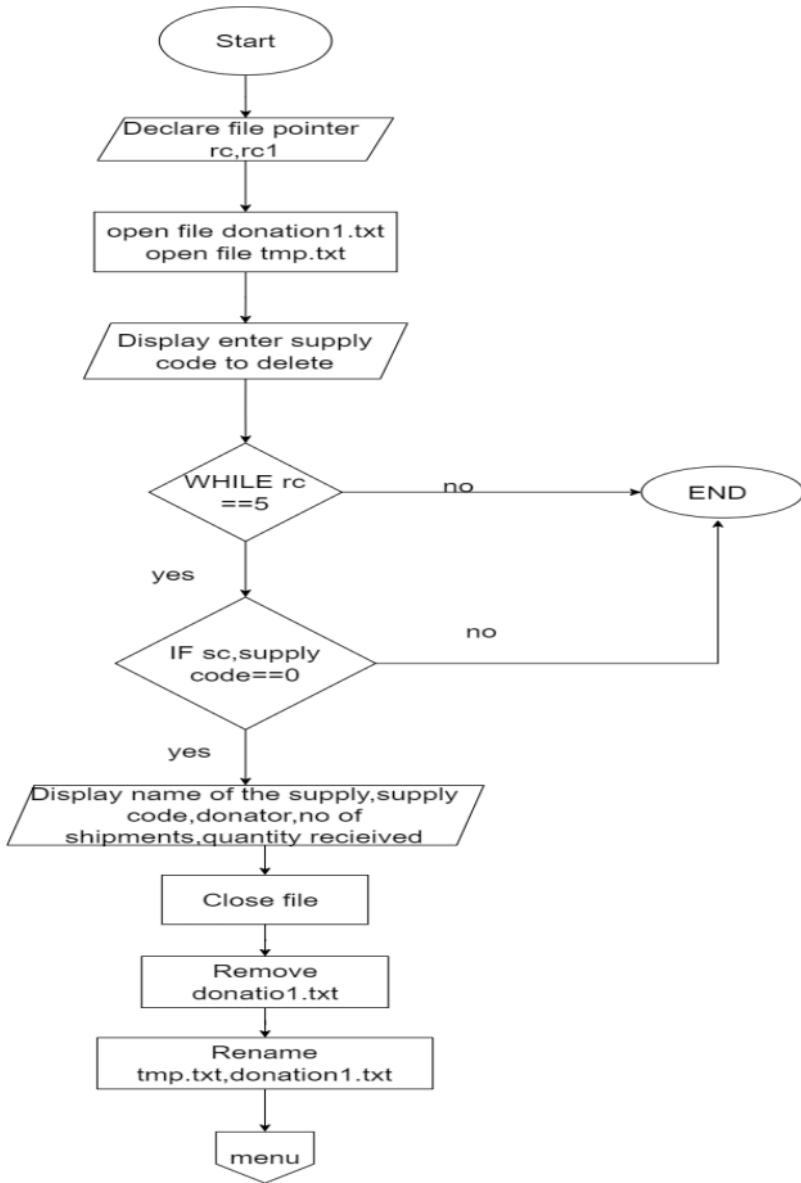


2.11 Delete

Pseudocode:

1. Begin
2. Declare file pointer L2 and L3.
3. Open file donation.txt in read mode.
4. Open file tmp.txt in append mode.
5. Read searchcode to be deleted.
6. If searchcode match supply code with data in donation.txt
 Delete all data of that supply code from file.
 Store data from.
 Close file.
 Remove donation.txt
 Rename tmp.txt,donation.txt
7. Call mainmenu ().
8. End

Flowchart:

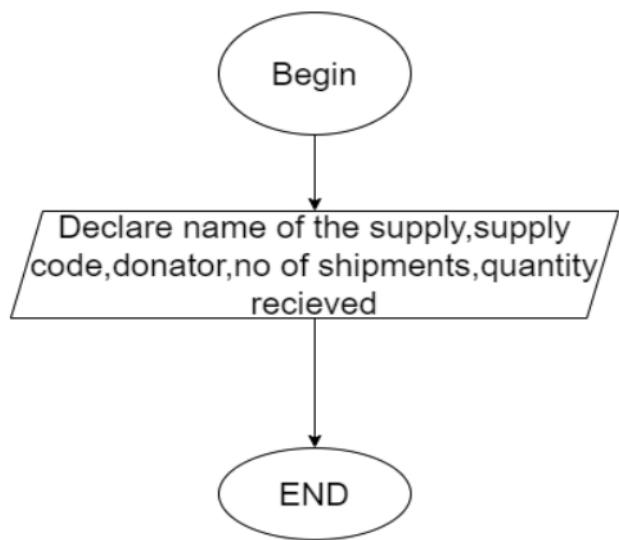


2.12 Struct

Pseudocode

1. Begin
2. Declare name of the supply, supply code, donator, no of shipments, quantity received
3. END

Flowchart



2.13 Bubble sort (struct distem P [], int n)

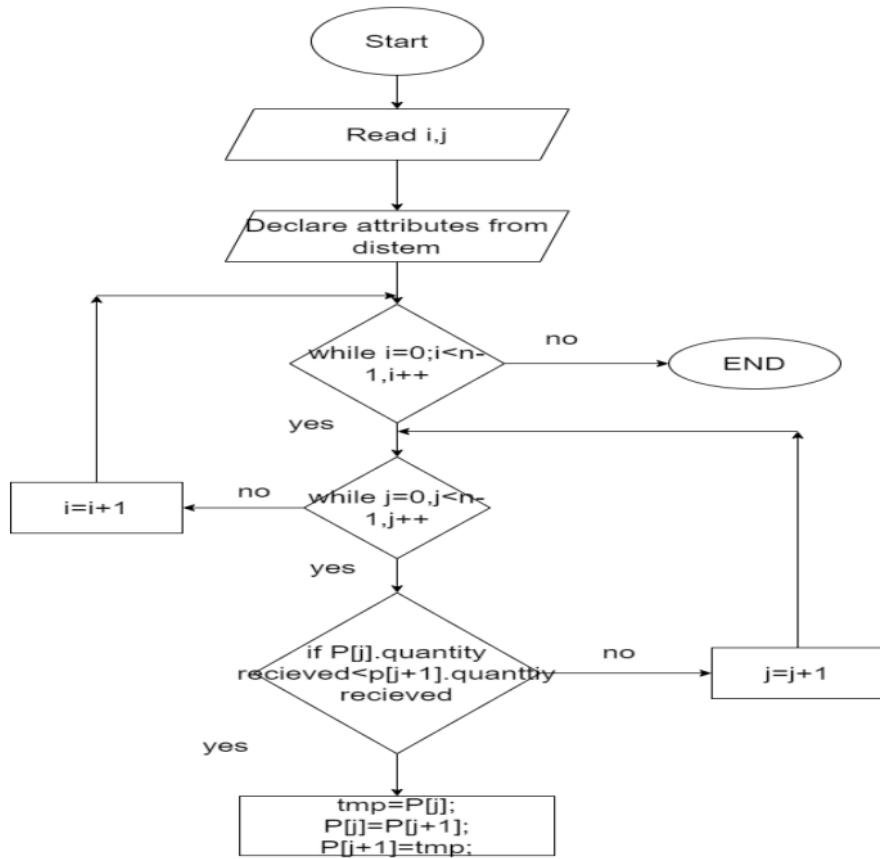
Pseudocode

```

1. Begin
2. Read i,j
3. Struct distem tmp
   For (i=0; i<n-1; i++)
      For (j=0; j<n-i; j++)
         If P[j]. quantity received < P[j+1]. quantity received
            Tmp=P[j]
            P[j]=P[j+1]
            P[j+1]=tmp
         Return
4. END

```

Flowchart:

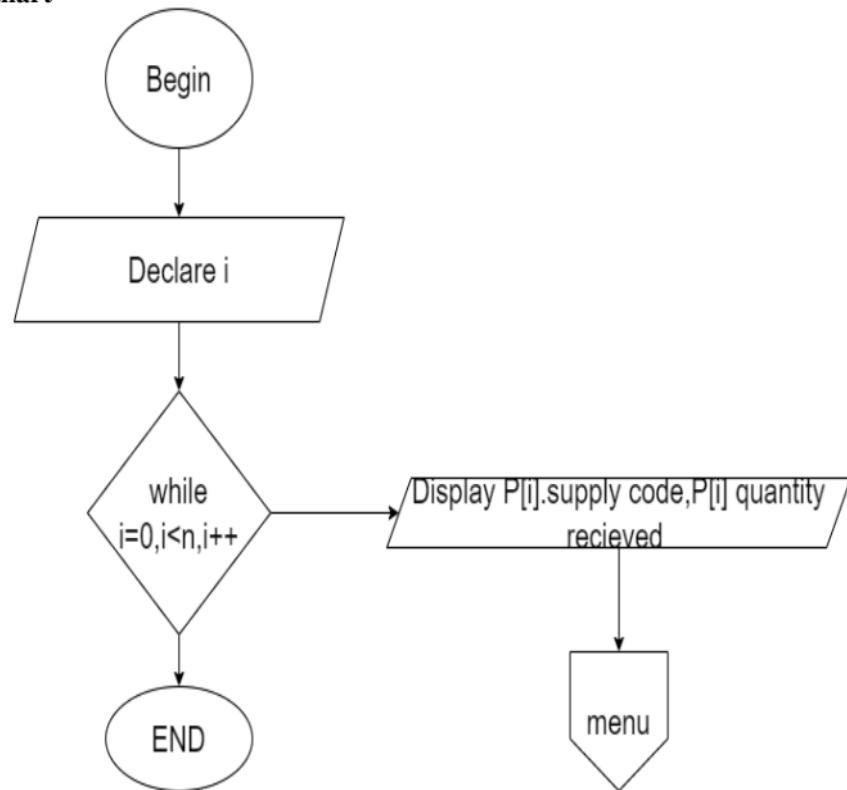


2.14 View distribution

Pseudocode

1. Begin 2
2. Declare i
 For (i=0, i<n, i++)
 Display “ P[i]. supply code, P[i]. quantity received”
3. Call menu
4. End

Flowchart

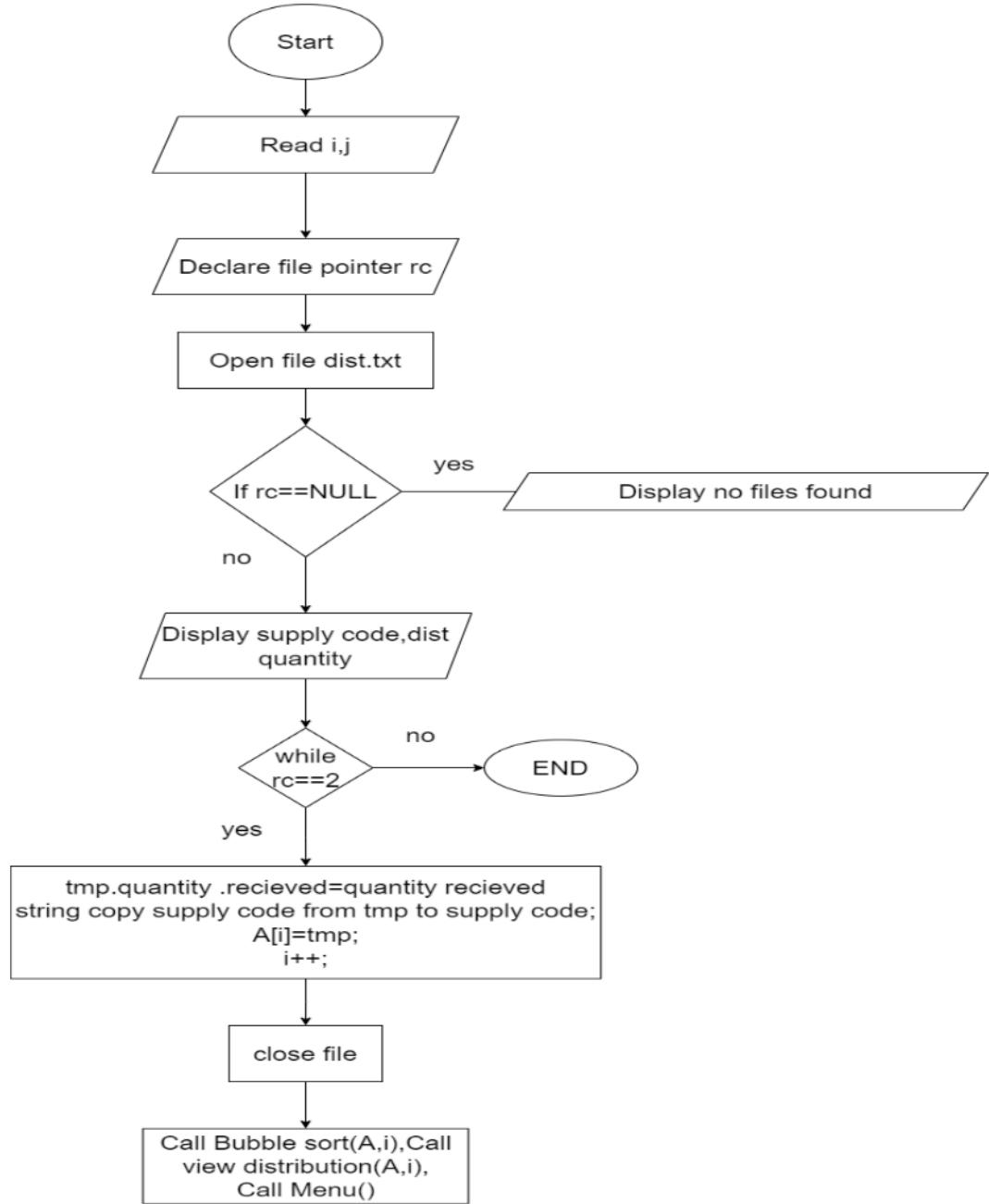


2.15 Display distribution

Pseudocode

1. Begin
2. Declare i=0
3. Struct distem A[40]
4. Struct distem tmp
5. Declare file pointer rc
6. Open file dist.txt
7. If rc==NULL
 Display “no files found”
Else
 Display supply code, dist quantity
- While rc==2
 Tmp.quantity received= quantity received
 Strcpy(tmp. Supply code, supply code)
 A[i]=tmp;
 i++;
8. Close file
9. Call Bubble sort (A,i)
10. Call View distribution(A,i)
11. Call Menu ()
12. END

Flowchart:



3.0 Function used in the program:

3.1 Library functions

These functions are the function allocated in a usual area which has integrated capabilities that are predefined in the library of the C. printf, scanf, etc. are the pre-defined functions in C.

The following are the functions used and they are described as seen below:

S.N.	Header files	Purpose
1.	#include	C preprocessor includes the contents to read in a text file Commanded by this command
2.	<stdio.h>	Information of input and output operations contained here
3.	<conio.h>	For creating console platform screen
4.	<string.h>	For character array manipulation

3.2 User defined Functions

- Main()

The outer menu which consist of function like login, signup and exit.

- Login()

This function allows user to login with their username and password which they get from signing up the application. After this the user will go in main menu.

- Signup()

This function help the user to register the username and password by registering their own information.

- Mainmenu()

This function consist of the all the menu of other function required in this application. From this function user can go to other stage as per their need.

- **Enter_donation()**
This function allow the user to enter the donation which they get from other organization.
- **View_donation()**
This function help to view all the donation which were store in system file i.e. store by the user.
- **Search_donation()**
It allows the user to search their required donation from the system.
- **Update donation()**
This function consist of the menu form which we can go to update the donation which is store in system.
- **add_donation()**
This function change the data of the system by adding the donation assign by user.
- **distribution()**
The distribute amount of data will be change from this function and help the user to know about the information of distribution.
- **delete_donation()**
This function allow the user to delete the donation from the system.
- **display_distribution()**
This function allow the user to know the list of the information about the distribution.
- **bubblesort()**
This function will allow the user to arrange the data in a proper order i.e. either in ascending order or descending order.

4.0 Concept of the c programming

It is a universal, procedural computer programming language introduced in the 1970s. Originally developed for writing operating systems. Easy to write with low memory access and a set of simple keywords. It is mainly popular in areas such as protocol stacks and laptop architectures.

4.1 Functions

C functions are used to split a large program into smaller programs. This makes it easier to debug and write your program, which makes your program more efficient. Library and custom functions are functions used in donation management systems.

Function sample

```
void view_donation()
{
    char s_name[40],s_code[40],donator[40];
    int no_of_shipment;
    float quantity_received;
    int choice;
    system("cls");

    FILE *L2;
    L2=fopen("donation.txt","r");

    if(L2==NULL)
    {
        printf("\nNo data to display...\n");
    }
    printf("\n\n\t\tView Donation record:");
    printf("\n\t\t-----\n\t");
    printf("\n\t\t Following donation record read from file!!\n");
    printf("\t\t-----\n");

    printf("\tSupply Name\t\tSupply code\t\tDonator\t\tShipment Count\t\tQuantity Received\n");
    printf("\t\t-----\t\t-----\t\t-----\t\t-----\t\t-----\n");
    while(fscanf(L2,"%s %s %s %d %f",s_name,s_code,donator,&no_of_shipment,&quantity_received)==5)
    {
        printf("\n\t\t%s\t\t%s\t\t%s\t\t%d\t\t%f\n",s_name,s_code,donator,no_of_shipment,quantity_received);
    }

    fclose(L2);

    printf("\n\n\t\tDo you want to goto mainmenu?[1=yes,0=no]?:");
    scanf("%d",&choice); fflush(stdin);

    if(choice==1)
    {
        mainmenu();
    }
    else
    {
        system ("cls");
        printf("\t\t\n\t\tEnter any key to close!!!!!\n\t\t\n\t\t\n\t\t");
        return;
    }
}
```

4.2 Structure:

Structures are user-defined data types that consist of various or many types of data types and keywords. The keyword "struct" is used in C programming to create a structure. The following is an example of the structure used in the program.

```
}

struct item{
    char s_name[40],s_code[40],donator[40];
    int no_of_shipments;
    float quantity_recieved;
};
```

4.3 Arrays:

Arrays are defined as data structures made up of groups of elements. This is a collection of one or more values of the same type. In our program, arrays are used to store offer codes, offer names, and donor values.

```
char s_name[40],s_code[40],donator[40],check_supply[40];
```

4.4 Loops: Loops are used to repeat the same line of code in a condition. The loop consists of two parts: the main loop and the control statement. Control statements are made up of conditions, and the loop body is a programming process. Below is a sample loop used in the program.

```
for(;;)
{
    printf("\n\t\t\tEnter Donation record:");
    printf("\n\t\t\t-----\n\t");
    printf("\n\t\t\t-----\n\t");
    printf("\n\t\t\tSupply Name: ");
    scanf("%s",&s_name); fflush(stdin);
    printf("\n\t\t\tSupply Code: ");
    gets(s_code); fflush(stdin);
    printf("\n\t\t\tDonator: ");
    gets(donator); fflush(stdin);
    printf("\n\t\t\tShipment Count: ");
    scanf("%d",&no_of_shipment); fflush(stdin);
    printf("\n\t\t\tReceived Quantity(M): ");
    scanf("%f",&quantity_received); fflush(stdin);

    fprintf(L2,"%s %s %s %d %f",s_name,s_code,donator,no_of_shipment,quantity_received);

    if(fprintf!=0)
    {
        printf("\n\n Data written successfully to file!\n");
    }
    else
    {
        printf("\n\nSomething went wrong..Try again..");
    }
    printf("Do you want to Re Enter donation?[1=yes,0=no ]:");
    scanf("%d",&choice); fflush(stdin);
    if(choice==1) continue;
    else
        break;
}
fclose(L2);
mainmenu();
}
```

In the above figure for loop is used to repeate the the system until the value is true.

4.5 Decision making:

Decision-making statements in C are also known as if-else statements. Performs one function in some cases and another in other cases. The following is an example of the decision making used in the program.

```
if(fprintf!=0)
{
    printf("\n\n Data written successfully to file!\n");
}
else
{
    printf("\n\nSomething went wrong.. \nTry again..");
}
printf("Do you want to Re Enter donation?[1=yes,0=no ]:");
scanf("%d",&choice); fflush(stdin);
if(choice==1) continue;
else
break;
```

4.6 File Handling

File management is used to create data files, write data to the created files, or read data from files. The C language uses C file management to store all the data available to the program in a file. This data can be retrieved / extracted from these files and made to work again in any program.

File processing keywords are fopen (), fclose (), fwrite (), fread (), fprintf (), fwritef (), fscanf () , etc. The main keyword used in the function is fopen & # 40. ; & # 41 ; , fclose (), fscanf () and fprintf () .

fopen (); is used as opener in a file, close file can be done by fclose(), to scan a file fscanf can be used, file can be saved by using fprintf(), It is used to display the data that has been created. The files handled by this program are dist.txt, donations. Temporary files named txt and temp.txt or tmp.txt.

```
void distribution()
{
    char s_name[40],s_code[40],donator[40],check_supply[40];
    int no_of_shipment;
    float quantity_received,dist_quantity;

    FILE *L2=fopen("donation.txt","r");
    FILE *L3=fopen("dist.txt","a");
    FILE *tmp=fopen("tmp.txt","a");

    if(L2==NULL || tmp==NULL)
    {
        printf("\nFile Could Not be opened.\n"); getch(); return;
    }

    printf("\nSupply code: "); scanf("%s",&check_supply); fflush(stdin);
    printf("\nDistribution Quantity: "); scanf("%f",&dist_quantity); fflush(stdin);
    while(fscanf(L2,"%s %s %s %d %f",s_name,s_code,donator,&no_of_shipment,&quantity_received)==5)
    {
        if(strcmp(check_supply,s_code)==0)
        {
            quantity_received=quantity_received-dist_quantity;
            fprintf(tmp,"\n%s %s %s %d %.2f",s_name,s_code,donator,no_of_shipment,quantity_received);
            fprintf(L3,"%s %s %.2f\n",s_name, s_code,dist_quantity );

            fclose(L3);
            printf("\n Data written successfully on dist.txt file.\n");
            continue;
        }
        fprintf(tmp,"\n%s %s %s %d %.2f",s_name,s_code,donator,no_of_shipment,quantity_received);
    }
    fclose(L2);fclose(tmp);
    remove("donation.txt");
    rename("tmp.txt","donation.txt");
    getch();
}
```

4.7 Pointer:

The variable that specifies the address of another variable is called pointer. It is used programmatically as a file pointer or pointer addressing. An example is shown below:

```
FILE *L2=fopen("donation.txt","r");
FILE *L3=fopen("dist.txt","a");
FILE *tmp=fopen("tmp.txt","a");
```

5.0 Additional Features of program

Although the minimum task requirements are very low, some of the additional features of this program are used to make the program more user-friendly, cleaner and more efficient. Some of the features added to the program are listed below.

i) **Signup**

Firstly user have to enter in the signup menu and after completing all the required thing the user can go to login page and enter into the mainmenu of the system.

ii) **Login**

Login feature prompts the user for a username and password before accessing the system and can only be accessed with the defined username and password.

iii) **Clean user interface**

However, due to the lack of graphic content, the visual experience is clear, easy to see and easy to read.

6.0 System Limitations

This program is a console-based C program and lacks security and functionality in many respects. Some of the restrictions of the program are:

i) **Very low data validation**

The program does not include data validation because the user can enter and save any input in the file and the program may crash.

ii) **Limited Functions The**

System has many restrictions on the performance of the following functions: B. There may be missing features in the management system for deleting data, as well as many other features that do not exist in the system.

iii) Low security support

The system can be called no security program because there is no security other than this login screen and you cannot change the login credentials and their default settings so that you can access and match the system.

7.0 Sample output:

The screenshots figures below are taken as sample output of the COBID-19 DBMS application.

```
*****
**WELCOME TO C-19 DMS**
*****  
  
Which operator do you want to perform?  
  
1. login  
2. signup  
3. Exit  
  
please enter a valid number to perform task!!  
:_
```

Fig 1: Entry menu sample output.

In the above figure the entry menu of COVID-19 DBMS. There are 3 option for user for login, signup and exit the program.

```
SIGNUP PAGE
-----
Enter your information:

Age:20
First Name:Jiten
last Name:Chapagaain
set your username and password which will be use for login!!
Username:Jatin
Password:Jatin45
Confirm Password:Jatin45
```

Fig 2: signup menu sample output

In the signup menu user has to enter the information as asked in the figure above. After completing the process, now the user can login with newly created COVID-19 DBMS account.

```
LOGIN PAGE
-----
Enter username:Jatin
Enter password:Jatin45
login successfull
Enter any key to go in home page of donation management system!!!■
```

Fig 3: login menu sample output

In the above figure the user ask to enter their login username and password which they get from signing up. After entering the correct username and password, the user can proceed ahead.

```
*****
**Donation Management System Home Page**
*****
1. Enter doantion
2. View donation
3. Update donation
4. Search donation
5. Delete donation
6. Display_distributed_donation
7. Exit

Which option do you want to choose?
Number :
```

Fig 4: mainmenu sample output

In this home page the user has 7 option, they can choose enter donation, view donation, update donation, search donation, delete donation. Display distribute donation and exit is the option available.

```
Enter Donation record:
-----
Supply Name: Hand_sanitizer
Supply Code: HS
Donator: USA
Shipment Count: 1
Received Quantity(M): 3.5

Data written successfully to file!
Do you want to Re Enter donation?[1=yes,0=no ]:
```

Fig 5: Enter donation sample output

The above screen shots the sample of output record function. In this function user can store the new donation record. After providing all detail the user will be asked to choose to add another record or not and if user press 1 he will continue to add record but if he press any other key they will go to mainmenu.

View Donation record:				
Following donation record read from file!!				
Supply Name	Supply code	Donator	Shipment Count	Quantity Received
C_Thermometer	CT	Japan	1	1.200000
Hand_Sanitizers	HS	USA	1	3.500000
Face_Mask	FM	China	2	120.000000
Surgical_Mask	SM	China	2	38.000000
Oxygen_Mask	OM	Saudi	2	9.000000
Hand_sanitizer	HS	USA	1	3.500000

Do you want to goto mainmenu?[1=yes,0=no]?:

Fig 6: view donation sample output

In the above figure we can see all the data and which are available in system.

Search Donation record:				
Enter a ID which you want to search!!: CT				
Supply Name	Supply code	Donator	Shipment Count	Quantity Received
C_Thermometer	CT	Japan	1	1.200000

Do you want to goto mainmenu?[1=yes,0=no]?:

Fig 7: search donation sample output

This is the sample output of search record. In this interface the user will asked to enter the supply code they want to search. After that the user will be asked to enter choice if he want to

go to mainmenu or not, if user input 1 the user will go to mainmenu and if he press any other character or integer the application will exit.

```
Update Donation record:  
-----  
which number you are looking for?  
1.Add Donation:  
2.Distribute Donation:  
3.Goto mainmenu:  
  
Enter Your Choice From[1-2]:
```

Fig 8: update donation sample.

In this sample the user can add and donation and add the value of quantity in system on the other hand user can subtract the value of quantity from the system.

```
Update Donation record:  
-----  
which number you are looking for?  
1.Add Donation:  
2.Distribute Donation:  
3.Goto mainmenu:  
  
Enter Your Choice From[1-2]: 1  
Supply Code: CT  
Received Quantity: 1.0
```

Fig 9: Add donation sample

In the above figure user enter the supply quantity and enter the receive quantity of the medical equipment. After that the quantity of CT code will be change.

```
Search Donation record:  
-----  
Enter a ID which you want to search!!: CT  
  
Supply Name           Supply code      Donator      Shipment Count  Quantity Received  
-----           -----          -----          -----          -----  
C_Thermometer        CT            Japan          1             2.200000  
  
Do you want to goto mainmenu?[1=yes,0=no]?:
```

Fig 10: Change value by add donation

From the above figure we can see that the amount of quantity has been change after we add the donation quantity.

```
Update Donation record:  
-----  
which number you are looking for?  
1.Add Donation:  
2.Distribute Donation:  
3.Goto mainmenu:  
  
Enter Your Choice From[1-2]: 2  
  
Supply code: HS  
  
Distribution Quantity: 2.0  
  
Data written successfully on dist.txt file.
```

Fig 11: Distribution donation sample

In the above figure user will choose the distribution option and enter the supply code and the quantity to be change in the system.

```
Distribution List...

-----
Supply Code      Dist.Quantity(M)
-----
CT              20.000000
CT              10.000000
CT              2.000000
HS              2.000000
SM              1.000000
                  0.000000

PRESS any key to continue..
```

Fig 13: display distribution sample output

The above diagram is the sample of the distribution list where user find the data of the distribution item and their quantity.

```
Delete Record.....  
-----  
Enter Supply Code to be delete!!: OM  
-----  
Process exited after 7.084 seconds with return value 0  
Press any key to continue . . .
```

Fig 14: Delete donation sample output

The above interface will ask the user to enter the supply code which have to delete and after that the all the data bout the respective supply code will be deleted from the system.

```
View Donation record:  
-----  
Following donation record read from file!!  


| Supply Name     | Supply code | Donator | Shipment Count | Quantity Received |
|-----------------|-------------|---------|----------------|-------------------|
| C_Thermometer   | CT          | Japan   | 1              | 2.200000          |
| Hand_Sanitizers | HS          | USA     | 1              | 2.000000          |
| Face_Mask       | FM          | China   | 2              | 120.000000        |
| Surgical_Mask   | SM          | China   | 2              | 38.000000         |

  
Do you want to goto mainmenu?[1=yes,0=no]?:
```

Fig 15: delete item is out of the list.

From the above figure user can see that the delete item is out of the list.

Conclusion

This program (COVID-19 DMS) is a console-based program, does not contain heavy graphic content, etc., and is programmed in the

C programming language. This application is programmed to track the records of Malaysia's Red Crescent Society and provide a clean and easy way to manage donations. Working on this mini-project isn't enough, but you can take it to the next level of programming in C and developing through thought. It wasn't really an easy task for me because I wasn't very familiar with the world of programming and didn't know most of the basics of writing this program. However, it helped me understand C programming concepts such as functions, struct, loops, conditional statements, and arrays. It also took logical understanding and thinking to another level.

Reference:

- Array (javatpoint,n.d): <https://www.javatpoint.com/c-array#:~:text=An%20array%20is%20defined%20as,%2C%20double%2C%20float%2C%20etc.>
- Pointer(cs.fsu.edu,n.d):<https://www.cs.fsu.edu/~myers/c++/notes/pointers1.html#:~:text=A%20pointer%20is%20a%20variable,essential%20for%20dynamic%20memory%20allocation>
- File handling(Harry,2019): <https://www.youtube.com/watch?v=qg69AXmHhx8>
- Loops (programiz.com,n.d): <https://www.programiz.com/c-programming/c-for-loop>
- C tutorial(w3schools,n.d): <https://www.w3schools.com/c/index.php>
- Pseudocodes (economictimes.indiatimes,n.d):
<https://economictimes.indiatimes.com/definition/pseudocode>
- Flowchart(asq.org,n.d): <https://asq.org/quality-resources/flowchart#:~:text=A%20flowchart%20is%20a%20picture,process%2C%20or%20a%20project%20plan.>
- Quora,2022. Create a Login Menu .[Online] Available at [quora.com](#)
- Programiz,(n.d) C while and do while loops [Online] Available at [programiz.com](#)
- Includehelp(n.d), Conflicting types of error in Dev C++,[Online],Available [includehelp.com](#)
- Technonet,2022, Fwrite Function in C [Online] Available at [technonet.com_articleworld.com.\(n.d\)](#)
- Cprogrammingnotes,(n.d) Bubble Sort in C ,[Online],[cprogrammingnotes.com](#)
- Guru99.com(n.d),C programming Functions [Online] Available at [guru99.com](#)
- Byjus,2022,File Handling in C,[Online] available at [byjus.com](#)

NPI000150-ICP assignment.

ORIGINALITY REPORT



PRIMARY SOURCES

- 1 Submitted to Asia Pacific University College of Technology and Innovation (UCTI) 3%
Student Paper
- 2 www.nexoncn.com 1 %
Internet Source

Exclude quotes Off

Exclude bibliography On

Exclude matches < 10 words