**ALGORITHM**

**STRUCTURE DECLERATION**

**Struct directory**

Declare c\_name[30]

Declare hs\_name[30]

Declare locality[30]

Declare dist[30]

Declare s\_pr[30]

Declare phone\_no

**Declare variable for structure, details[200]**

**Struct space**

Declare dup\_name[30]

**MAIN MODULE**

Step 1: Start

Step 2: Declare choice, i

Step 3: Declare size, assign to 50

Step 4: Declare struct directory details[50]

Step 5: initialize to 50 customer details

Step 6: for i=0 to 50; i++

Step 6.1: print all the details

Step 7: do while (choice!=5)

Step 7.1: Enter the choice

Step 7.2: if (choice==1)

Step 7.2.1: Add\_customer(details,200,size) (function)

Step 7.2.2; increment size by 1

Step 7.2.3: continue

Step 7.2.4: End if

Step 7.3: else if(choice==2)

Step 7.3.1: searching(details,size) (function)

Step 7.3.2: continue

Step 7.3.3: End if

Step 7.4: else if(choice==3)

Step 7.4.1: update\_details(details,size) (function)

Step 7.4.2: continue

Step 7.4.3: End if

Step 7.5: else if(choice==4)

Step 7.4.1: reports(details,size) (function)

Step 7.4.2: continue

Step 7.4.3: End if

Step 8: print Thank you

Step 9: Return 0

Step 10: End

**MODULE 1** **ADDING CUSTOMER DETAILS:**

Step 1: start

Step 2: declare integer j, k

Step 3: output “Enter the details”

Step 4 : do while (!k)

step 4.1: enter customer name

step4.2: Validate customer name(function)

step4.3: Assign k=valid\_name(details[sz].c\_name);

Step 5 : do while (!k)

step 5.1: enter house name

step5.2: Validate house name(function)

step5.3: Assign k=valid\_name(details[sz].hs\_name);

Step 6 : do while (!k)

step 6.1: enter locality

step6.2: Validate locality(function)

step6.3: Assign k = valid\_name(details[sz].locality);

Step 7: do while (!k)

step 7.1: enter district

step 7.2: Assign j=0

step 7.3: while(details[sz].dist[j]!='\0')

step 7.3.1: Convert district to uppercase letters

step 7.3.2: increment j by 1

step 7.4: Validated district(function)

step 7.5: Assign k = valid\_name(details[sz].dist);

Step 8:do while (!k)

Step 8.1: Enter service provider details

step 8.3: while (details[sz].s\_pr[j]!='\0')

step 8.3.1: Convert service provider name to uppercase letters

step 8.3.2: increment j by 1

step 8.4: Validated service deatils(function)

step 8.5: Assign k = valid\_sevice(details[sz].s\_pr);

Step 9: do while (!k)

step 9.1: enter mobile number

step 9.2: Validate mobile number(function)

step 9.3: Assign k = valid\_ph\_no(details[sz].phone\_no);

Step 10: for i=0 to sz ; i++

Step 10.1: output print details of customer

**MODULE 2 VALIDATION:**

**VALIDATING NAME:**

Step 1: Declare j, flag

Step 2: Assign flag = 1

Step 3: Assign j= strlen(dup);

Step 4: for i=0 to j ; i++

Step 4.1: if(!(isalpha(\*(dup+i)) || isspace(\*(dup+i))))

Step 4.1.1: Assign flag =0

Step 4.1.2: break

Step 4.2: End if

Step 5: if (!flag)

Step 5.1: print invalid details

Step 5.2: End if

Step 6: Return flag

**VALIDATING SERVICE PROVIDER:**

Step 1: Declare flag and assign it to 0

Step 2: if (!strcmp(dup,"JIO"))

Step 2.1:Assign flag=1

Step 2.2: End if

Step 3: else if (!strcmp(dup,"BSNL"))

Step 3.1:Assign flag=1

Step 3.2: End if

Step 4: else if (!strcmp(dup,"VI"))

Step 4.1: Assign flag=1

Step 4.2: End if

Step 5: else if (!strcmp(dup,"AIRTEL"))

Step 5.1: Assign flag=1

Step 5.2: End if

Step 6: else

Step 6.1: Print invalid details

Step 7: return flag

**VALIDATING PHONE NUMBER:**

Step 1: Declare count, flag

Step 2: Assign flag to 0, count to 0

Step 3: while (dup!=0)

Step 3.1: Assign dup=dup/10

Step 3.2: Increment count by 1

Step 4: if (count!=10)

Step 4.1: Assign flag to 0

Step 4.2: print Invalid phone number

Step 4.3: Return flag

Step 4.4: End if

Step 5: else if (count==10)

Step 5.1: Return flag

Step 5.2: End if

**MODULE 3 SEARCHING:**

**NAME OF CUSTOMER VIA MOBILE NUMBER:**

Step 1: Declare ch, flag, size1, ph\_no,i

Step 2: Assign flag to 0

Step 3: do while(ch!=4)

Step 3.1: Enter the respective choice

Step 3.2: if (ch==1)

Step 3.2.1: Read mobile number

Step 3.2.2: for i= 0 to size; i++

Step 3.2.2.1: if(ph\_no==details[i].phone\_no)

Step 3.2.2.1.1: Assign size1 to i

Step 3.2.2.1.2: Assign flag to 1

Step 3.2.2.1.3: break

Step 3.2.2.1.4: End if

Step 3.2.3: if(flag==1)

Step 3.2.3.1: output name of the customer

Step 3.2.3.2: End if

Step 3.2.4: else if (flag==0)

Step 3.2.4.1: output mobile number not registered

Step 3.2.4.2: End if

Step 3.2.5: End if

**LIST BY PARTICULAR SERVICE PROVIDER:**

Step 3.3: else if(ch==2)

Step 3.3.1: Declare j, assign to 0

Step 3.3.2: Declare service[10]

Step 3.3.3: Read service provider

Step 3.3.4: while (service[j]!='\0')

Step 3.3.4.1: Convert service to upper case letters

Step 3.3.4.2: increment j by 1

Step 3.3.5: for i=0 to size ;i++

Step 3.3.5.1: if (!strcmp(service,details[i].s\_pr))

Step 3.3.5.1.1: print details by service provider

Step 3.3.5.1.2: End if

Step 3.3.6: End if

**MOBILE NUMBER OF CUSTOMER VIA NAME:**

Step 3.4: else if(ch==3)

Step 3.4.1: Declare k, name[30]

Step 3.4.2: Read name

Step 3.4.3: Assign k = remove\_space(details,size,name)

Step 3.4.4: if(k)

Step 3.4.4. 1: print phone number of customer

Step 3.4.4.2: End if

Step 3.4.5: else if (k==0)

Step 3.4.5.1: print customer has not registered with us

Step 3.4.5.2: End if

Step 3.4.6: End if

Step 4: End function

**MODULE 4 REMOVE SPACE (RIGHT SPACE):**

Step 1: Declare struct space dup[size1]

Step 2: Declare j, fl,i, flag, k, assign fl to 0

Step 3: for j=0 to size1; j++

Step 3.1: Assign i=strlen(details[j].c\_name)

Step 3.2: Assign flag to 1

Step 3.3: while(flag)

Step 3.3.1: if(isalpha(details[j].c\_name[i-1]))

Step 3.3.1.1: Assign flag=0

Step 3.3.1.2: End if

Step 3.3.2: Decrement i by 1

Step 3.4: for k= 0 to i; k++

Step 3.4.1: Assign dup[j].dup\_name[k]=details[j].c\_name[k]

Step 3.5: Assign dup[j].dup\_name[i+1]='\0'

Step 3.6: if(!strcmp(dup[j].dup\_name,s))

Step 3.6.1: Assign fl to j

Step 3.6.2: break

Step 3.6.3: End if

Step 4: Return fl

**MODULE 5 UPDATE DETAILS:**

Step 1: Update\_details(struct directory details[],int size)

Step 2: declare integer Ch, I, j, k, size1, flag=0;

Step 3: declare char dup

Step 4: declare long int ph\_no

Step 5: do

Step 5.1: output “Enter the respective choice”

Step 5.2: output “1. Change service provider 2. Change in House Name, Locality and District 3. Adding new number for an existing customer 4. To exit the function”

Step 5.3: Input Ch

Step 5.4: If (Ch==1)

Step 5.4.1: do

Step 5.4.1.1: output “Enter Number of the Customer to change the details”

Step 5.4.1.2: input ph\_no

Step 5.4.1.3: k=valid\_ph\_no(ph\_no)

Step 5.4.1.4: while (! k)

Step 5.4.1.5: for (I =0;i<size ;i++)

Step 5.4.1.5.1: if(ph\_no==details[i].phone\_no)

Step 5.4.1.5.2: assign size=i

Step 5.4.1.5.3: assign flag=1

Step 5.4.1.5.4: break

Step 5.4.2: if (flag==1)

Step 5.4.2.1: output “Customer name is: Your current service is:”

Step 5.4.2.2: STRCPY (dup, details [size1].s\_pr)

Step 5.4.3: do

Step 5.4.3.1: output “Our service are airtel, Jio, VI, BSNL”

Step 5.4.3.2: input details [size1]. S\_pr

Step 5.4.4: assign j=0

Step 5.4.5: while (details[size1].s\_pr[j]!=\0)

Step 5.4.5.1: assign details[size1].s\_pr[j]=toupper(details[size1].s\_pr[j])

Step 5.4.5.2: assign j++

Step 5.4.6: assign k = valid\_sevice(details[size1].s\_pr)

Step 5.4.7: if(!strcmp(dup,details[size1].s\_pr))

Step 5.4.7.1: output “enter the server provider you want”

Step 5.4.7.2: assign k=0

Step 5.4.8: while (! K)

Step 5.4.9: output "Your new service provider is %s””

Step 5.4.10: output “Your service provider has been successfully changed”

Step 5.5: else if (Ch==2)

Step 5.5.1: flag=0

Step 5.5.2:do

Step 5.5.2.1: Output “Enter Mobile number of Customer”

Step 5.5.2.2: Input phone number

Step 5.5.2.3: Assign k=Valid phone number

Step 5.5.3: While (! K)

Step 5.5.4: for(i=0;i<size;i++)

Step 5.5.4.1: if phone number=details

Step 5.5.4.1.1: Assign size=I, flag =1

Step 5.5.4.1.2: Break

Step 5.5.5: if flag==1

Step 5.5.6: “Enter customer details”

Step 5.5.7: do

Step 5.5.8: Input House Name, Locality, District

Step 5.5.9: Assign input values to const variable

Step 5.5.10: While (! K)

Step 5.5.11: Output updated details

Step 5.6: else if (Ch==3)

Step 5.6.1: Assume flag=0

Step 5.6.2: do

Step 5.6.2.1: Output “Enter current number”

Step 5.6.2.2: Input Phone number

Step 5.6.2.3: Assign with constant value

Step 5.6.3: While (! K)

Step 5.6.4: for(i=0;i<size;i++)

Step 5.6.4.1: if phone number=details

Step 5.6.4.1.1: Assign size=I, flag =1

Step 5.6.4.1.2: Break

Step 5.6.5: if flag==1

Step 5.6.5.1: Output “Existing Mobile number of the customer”

Step 5.6.5.2: Output “Enter new number”

Step 5.6.5.3: Input New number

Step 5.6.5.4: Output “Updated details is”

Step 5.6.6: else if (flag ==0)

Step 5.6.6.1: Output “Enter mobile number has not registered”

**MODULE 6 REPORTS:**

Step 1: Declaration of variables ch, flag=0, size1, i, k, ph\_no, s[30]

Step 2: Enter the Respective choice(ch)

Step 3: If ch==1

Step 3.1: Read the mobile number from the user

Step 3.2: Validate the phone number (Read number again till the entered mobile number is valid)

Step 3.3: For (i=0;i<size;i++)

Step 3.3.1: if ph\_no==details[i].phone\_no

Step 3.3.1.1: Assign size1=i

Step 3.3.1.2: Assign flag=1

Step 3.3.1.3: break

Step 3.4: if flag==1

Step 3.4.1: Display all the details of the customer

Step 3.5: else

Step 3.5.1: Display “Entered mobile Number is not with us”

Step 4: Else If ch==2

Step 4.1: Declare variables j=0, b=0, a=0, v=0, flag=0, p, tot

Step 4.2: Read the district name from the user

Step 4.3: Validate the name (Read name again till the entered district name is valid)

Step 4.4: For (i=0;i<size;i++)

Step 4.4.1: if(!strcmp(details[i].dist,s))

Step 4.4.1.1: Assign flag=1

Step 4.4.1.2: if(!strcmp(details[i].s\_pr,"JIO"))

Step 4.4.1.2.1: Assign j=j+1

Step 4.4.1.3: if(!strcmp(details[i].s\_pr," BSNL"))

Step 4.4.1.3.1: Assign b=b+1

Step 4.4.1.4: if(!strcmp(details[i].s\_pr," AIRTEL"))

Step 4.4.1.4.1: Assign a=a+1

Step 4.4.1.5: if(!strcmp(details[i].s\_pr," VI"))

Step 4.4.1.5.1: Assign v=v+1

Step 4.5: if(flag==0)

Step 4.5.1: Display “There are no customers in the district”

Step 4.6: else if(flag==1)

Step 4.6.1: Assign tot=j+b+a+v

Step 4.6.2: Calculate and Display “Percentage of people using JIO and (j\*100)/tot”

Step 4.6.2: Calculate and Display “Percentage of people using AIRTEL and (a\*100)/tot”

Step 4.6.2: Calculate and Display “Percentage of people using VI and (v\*100)/tot”

Step 4.6.2: Calculate and Display “Percentage of people using BSNL and (b\*100)/tot)”

Step 5: Else If ch==3

Step 5.1: Declare variables j=0, b=0, a=0, v=0, flag=0, p, tot

Step 5.2: For (i=0;i<size;i++)

Step 5.2.1: if(!strcmp(details[i].s\_pr,"JIO"))

Step 5.2.1.1: Assign j=j+1

Step 5.2.2: if(!strcmp(details[i].s\_pr," BSNL"))

Step 5.2.2.1: Assign b=b+1

Step 5.2.3: if(!strcmp(details[i].s\_pr," AIRTEL"))

Step 5.2.3.1: Assign a=a+1

Step 5.2.4: if(!strcmp(details[i].s\_pr," VI"))

Step 5.2.4.1: Assign v=v+1

Step 5.3: tot=j+b+a+v

Step 5.4: if((j>b&&j>a)&&j>v)

Step 5.4.1: Display “JIO has highest number of Customers”

Step 5.5: else if((b>j&&b>a)&&b>v)

Step 5.5.1: Display “BSNL has highest number of Customers”

Step 5.6: else if((a>j&&a>b)&&a>v)

Step 5.6.1: Display “AIRTEL has highest number of Customers”

Step 5.7: else

Step 5.7.1: Display “VI has highest number of Customers”

-----------------------------------------------------END----------------------------------------------------