

# **PROJECT REPORT ON CONTACT DIARY**

**A project report submitted in partial fulfillment of the requirements  
for the class – XII Computer Science.**



## **SUBMITTED BY**

**Mast. Shubham Mehadia  
Mast. Naveen Kaswan  
Mast. Abhishek Tilwar  
Mast. Dheeraj Rathi**

## **PROJECT GUIDE**

**Mr. Nitin D. Aglawe  
PGT. Computer Science**

**Department of Computer Science  
Narayana Vidyalayam  
Nagpur  
2018 – 2019**

## **DECLARATION**

**We the undersigned hereby declare that the Project Report titled “Contact Diary” submitted by us is based on actual work carried out by us under the guidance of**

**Mr. Nitin D. Aglawe.**

**Any reference to work done by any other person or institution or any material obtained from other sources have been duly cited and referenced. It is further to state that this work is not submitted anywhere else for any examination.**

**Thanking you.**

**Place: Nagpur**

**Yours Sincerely**

**Date: \_\_\_\_\_**

**Mast. Shubham Mehadia**

**Mast. Naveen Kaswan**

**Mast. Abhishek Tilwar**

**Mast. Dheeraj Rathi**

# **CERTIFICATE**

**This is to certify that the project entitled “Contact Diary” is carried out and developed by Mast. Shubham Mehadia, Mast. Naveen Kaswan, Mast. Abhishek Tilwar and Mast. Dheeraj Rathi students of class XII in partial fulfillment of the requirement of the CBSE Board under my guidance and supervision.**

**The said work has been assessed by me and I am satisfied that the same is up to the standard envisaged for the level of the course.**

**Project Guide  
(Mr. Nitin D. Aglawe)  
PGT. Computer Science  
Narayana Vidyalayam  
Nagpur.**

**Principal’s Signature**

**External Examiner**

## **ACKNOWLEDGEMENT**

**We express our sincere thanks to all those who have provided us valuable guidance towards the completion of this mini project as a part of the syllabus of class XII Computer Science.**

**At first, we extend our thanks to honorable Principal Mrs. Mala Chembath, Narayana Vidyalayam, Nagpur for providing us the varieties of opportunities, infrastructure, facilities and inspiration to gather professional knowledge and material without which it would have been impossible to complete this hard work.**

**We hereby take the opportunity to express deep sense of gratitude and whole hearted thanks to our project guide Mr. Nitin D. Aglawe (PGT Computer Science) for his inspiration and guidance throughout the course in this project work.**

**Finally, We extend special thanks and respect to parents and friends for encouraging us for class XII Computer Science course.**

**Thank you.**

**Place : Nagpur**

**Date: \_\_\_\_\_**

**Mast. Shubham Mehadia**

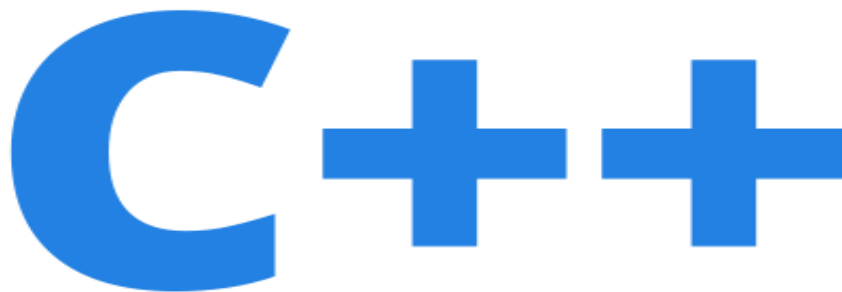
**Mast. Naveen Kaswan**

**Mast. Abhishek Tilwar**

**Mast. Dheeraj Rathi**

## **INTRODUCTION**

C++ is a general-purpose programming language. It has imperative, object-oriented and generic programming features while also providing facilities for low level manipulation.



It was designed with bias towards system programming and embedded, resource- constrained and large systems, with performance, efficiency and flexibility of use as its design highlights. C++ has also been found useful in many other contexts which key strengths being software infrastructure and resource constrained applications, including desktop applications, servers (e.g. e-commerce, web search or SQL servers), and performance- critical applications (e.g. telephone switches or space probes). C++ is a compiled language, with implementations of it available on many platforms many other programming languages have been influenced by C ++ including C#D Java and newer versions of C.

C++ was developed by Bjarne Stroustrup at Bell Labs since 1979, as an extension of the C language as he wanted an efficient and flexible language similar to C, which also provided high-level features for program organization. He

extended the C language by adding features from the language Simula 67. As well as C and Simula's influences, other languages also influenced C++ including ALGOL 68, Ada, CLU and ML.

C++ is standardized by the international organization for standardization (ISO), with the latest standard version ratified and published by ISO in December 2017 as ISO/IEC 14882:2017 (informally known as C++17). The C ++ programming language was initially standardized in 1998 as ISO/ IEC 14882:1998, which was amended by the C++03, C++11 and C++14 standards. The current C ++17 standard supersedes these with new features and an enlarged standard library. C++20 is the next plan standard thereafter.

C++ is an object oriented language and thus, it support object oriented programming(OOP) concepts which are:

- Data Abstraction
- Data Encapsulation
- Modularity
- Inheritance
- Polymorphism

### **Data Abstraction:**

Data abstraction refers to providing only essentially information to the outside world and hiding their background details, i.e., to represent the needed information in program without presenting the details.

### **Data Encapsulation:**

Data encapsulation is defined as wrapping up of data and information under a single unit. Encapsulation is an object oriented programming concept that binds together the data and functions that manipulate the data, and that keeps both safe from outside interference and misuse

### **Modularity:**

Modularity refers to the concept of making multiple modules first then linking and combining them to form a complete system modularity enables reusability and minimizes duplication modularity is the property of a system that has been decomposed into a set of cohesive and loosely coupled modules it is the process of subdividing a computer program into separate subprograms

### **Inheritance:**

Inheritance is the capability of one class of things to derive capabilities or properties from another class inheritance enables new objects to take on the properties of existing objects.

### **Polymorphism:**

Polymorphism is the ability of an object to assume or become many different forms of objects it is the ability for a message or data to be processed in more than one form

**ABOUT**  
**THE**  
**PROJECT**



# SOFTWARE ON CONTACT DIARY

A contact diary is a record containing some specific details of a person. The result is a facility carrying a range of data under one roof, including full address and general information. In theory, contact diary allows users to fill required information of a contact as per their needs.

Our project is a small working model of a contact diary. It consists of four operations the user can choose from.

1. **Add A New Contact:** The user can choose this option to add a new contact in the file by feeding information such as name, phone number and address of a contact.
2. **Show All The Names:** The user can choose to view all the existing contact names of the file in a tabular form, further user can choose to view details of specific contact that exist in file.
3. **Modify A Contact:** The user can choose this option to manually find a particular contact using the unique parameters of contact, i.e. contact name and then change the parameters of contact such as contact name, contact phone number and contact address.
4. **Delete A Contact:** The user can choose this option to manually find a particular contact using the unique parameters of contact, i.e. contact name and then delete the entire record corresponding to that particular contact.

# **SYSTEM REQUIREMENTS**

## Hardware Requirements:

RAM: 1 GB or more

Disk Capacity: 512 MB or more

Processor: Intel Pentium III or higher

Clock Speed: 1.2GHz or higher

## Software Requirements:

Application: Borland Turbo C++ (v3.0)

Application Version: 3.0

Operating System: Windows 7 Ultimate or higher

# RESOURCE

## **Header Files used:**

- `fstream.h` : for file handling input/output functions
- `conio.h` : for console input/output functions
- `process.h` : for function declarations and macros working with threads and processes
- `stdlib.h` : for C++ standard library functions
- `stdio.h` : for standard input and output operations
- `iomanip.h` : for input/output manipulator functions
- `string.h` : for string functions
- `graphics.h` : for building different shapes, altering text fonts and size, changing colors.
- `dos.h` : for handling interrupts, producing sounds, date and time functions

## **Objects used:**

- File objects: `f`
- Class objects: `h1`

## **Data files used:**

CONTACTA, contact2

# SOURCE CODE

```

#include<stdio.h>
#include<conio.h>
#include<fstream.h>
#include<stdlib.h>
#include<dos.h>
#include<string.h>
#include<graphics.h>
#include<iomanip.h>

//*****
//    CLASS NAME: CONTACT DIARY
//*****
class contact
{
public:
    char name[20];
    char add[100];
    long phno;
    void getdata();
    void show();
    void details();
};

//*****
//    FUNCTIONS TO GET AND SHOW DATA
//*****

void contact ::getdata()
{
    cout<<"\n\n\n\n ENTER DETAILS ABOUT CONTACT YOU WANT
TO ADD :=>";
    cout<<"\n\n\n\t\tEnter Name of contact :=> ";
    cin>>name;
    //fflush(stdin);
    cout<<"\n\t\tEnter Phone/Mobile number :=> ";
    cin>>phno;
    cout<<"\n\t\tEnter Address of contact :=> ";
    cin>>add;
    fflush(stdin);

```

```

}
void contact ::show()
{
    int i;
    cout<<" \n";
    cout<<"\t\t"<<name;
    for(i=strlen(name);i<14;i++)
        cout <<" ";
    fflush(stdin);
}

void contact::details()
{
    gotoxy(60,25);
    clearviewport();
    textbackground(CYAN);
    textcolor(CYAN);
    clrscr();
    char ans;
    ifstream f;
    contact h1;
    char name[20];
    char add[100];
    long phno;
    f.open("CONTACTA",ios::in|ios::binary);
    settextstyle(7,0,2);
    setcolor(YELLOW);
    outtextxy(200,20,"DETAILS OF CONTACT");
    cout<<"\n\n\n\n Enter contact name for checking details of it :- ";
    cin>>name;
    do
    {
        f.read((char *)&h1,sizeof(h1));
        if(f.eof()==1) {break;}
        if(strcmp(h1.name,name)==0)
        {
            cout<<"\n          Name   :- "<<h1.name;
            cout<<"\n          Phone number :- "<<h1.phno;
            cout<<"\n          Address :- "<<h1.add;

```

```

        cout<<"\n";
        getchar();
    }
}while(f);
f.close();
setcolor(GREEN);
settextstyle(7,0,1);
outtextxy(250,410,"Check more details(y/n) :- ");
ans=getchar();
if(ans=='Y' || ans=='y')
    h1.details();
else if(ans=='N' || ans=='n')
    exit(0);
}
//*****
//    TO INVOKE WELCOME SCREEN
//*****

void graph()
{
    setcolor(BLUE);
    settextstyle(7,0,5);
    outtextxy(115,200,"CONTACT DIARY");
    getch();
    clearviewport();
}

void main()
{
    clrscr();
    int gdriver=DETECT,gmode;
    initgraph(&gdriver,&gmode,"C:\\TC\\BGI");
    graph();
    settextstyle(TRIPLEX_FONT,0,4);
    char r[]="CREATED By:-";
    for(int i=0;r[i]!='\0';i++)

    {

```

```

        cout<<r[i]<<" ";
        delay(100);

    }
    settextstyle(10,0,3);
    setcolor(RED);
    outtextxy(100,80," SHUBHAM MEHADIA  XII-B  ");
    setcolor(YELLOW);
    delay(600);
    outtextxy(100,180," NAVEEN KASWAN  XII-B  ");
    setcolor(CYAN);
    delay(600);
    outtextxy(100,280," ABHISHEK TILWAR  XII-B  ");
    setcolor(GREEN);
    delay(600);
    outtextxy(100,380," DHEERAJ RATHI    XII-C  ");
    delay(600);

    getch();
    char s;
    contact h1;
    z:
    fstream f;

    f.open("CONTACTA",ios::in|ios::out|ios::app|ios::ate|ios::binary)
    ;
    clearviewport();
    settextstyle(10,HORIZ_DIR,3);
    setcolor(LIGHTGREEN);
    outtextxy(50,15,"WHAT IS YOUR NEED:");
    settextstyle(TRIPLEX_FONT,0,3);
    setcolor(CYAN);
    outtextxy(50,100,"1 .  ADD NEW CONTACTS");
    setcolor(LIGHTBLUE);
    outtextxy(50,150,"2 .  SHOW NAMES OF SAVED CONTACTS");
    setcolor(LIGHTRED);
    outtextxy(50,200,"3 .  MODIFY CONTACTS");
    setcolor(LIGHTMAGENTA);

```



```

outtextxy(50,250,"4 . DELETE CONTACTS");
setcolor(LIGHTCYAN);
outtextxy(50,300,"5 . EXIT");
int a;
settextstyle(7,0,5);
setcolor(BROWN);
outtextxy(20,400, "ENTER YOUR CHOICE:");
f.seekg(0);
gotoxy(55,25);
cin>>a;
char x;
gotoxy(1,1);
switch (a)
{
//*****
// CASE : 1
// DETAILS : TO ADD NEW CONTACTS
//*****
case 1:
b:
{gotoxy(1,1);
clearviewport();
textbackground(RED);
textcolor(BLUE);
clrscr();
fstream f;
f.open("CONTACTA",ios::in|ios::out|ios::ate|ios::app|ios::binary)
;
settextstyle(7,0,2);
setcolor(CYAN);
outtextxy(200,20,"NEW CONTACT");
char ans;
h1.getdata();
f.write((char *)&h1,sizeof(h1));
cout << "\nCONTACT SAVED :\n\n";
setcolor(GREEN);
settextstyle(7,0,1);
outtextxy(250,410,"Want to add more(y/n):");
ans=getchar();

```

```

    if(ans=='Y' || ans=='y')
        goto b;
    else if(ans=='N' || ans=='n')
        goto z;
    else
        goto z;
}
//*****
// CASE      : 2
// DETAILS : TO SHOW NAMES OF ALL CONTACTS
//*****
case 2:
{
    clearviewport();
    cout<<"\n\n";
    fstream f;

f.open("CONTACTA",ios::in|ios::out|ios::ate|ios::app|ios::binary)
;
    settextstyle(7,0,2);
    setcolor(CYAN);
    outtextxy(200,20,"NAMES OF CONTACTS");
    char ans;
    f.seekg(0);
    int ctr=0;
    cout <<"\t\n          Name          \n";
    cout <<"\t\n -----\n";
    while(f.read((char *)&h1,sizeof(h1)))
    {
        ctr=ctr+1;
        if(ctr==12) // For displaying 10 records per screen
        {
            cout <<"\t\n -----\n";
            getchar();
            clrscr();
            cout <<"\n\t          Name          \n";
            cout <<"\t\n -----\n";
            ctr=0;
        }
    }
}

```

```

    h1.show();

    if(f.eof()==1)
    {
        break;
    }
}
f.close();
setcolor(GREEN);
settextstyle(7,0,1);
outtextxy(50,410,"Want to see details of a specific contact(y/n)
:\n");
cin>>ans;
if(ans=='y' || ans=='Y')
    h1.details();
else if(ans=='n' || ans=='N')
{
    goto z;
}
else
{
    goto z;
}
}

//*****
//  CASE    : 3
//  DETAILS  : TO MODIFY A CONTACT
//*****
case 3:
{
    clearviewport();
    textcolor(YELLOW);
    clrscr();
    char ans;
    d:
    fstream f;
    contact h1;
    char name[20];

```

```

char add[100];
long phno;

f.open("CONTACTA",ios::in|ios::binary);
settextstyle(7,0,2);
setcolor(CYAN);
outtextxy(200,30,"MODIFYING CONTACT");
cout<<"\n\n\n\n";
cout<<" Enter contact name you want to modify : ";
cin>>name;
do
{
    f.read((char *)&h1,sizeof(h1));
    if(f.eof()==1)
    {
        break;
    }
    if(strcmp(h1.name,name)==0)
    {
        cout<<"\n      Name :-"<<h1.name;
        cout<<"\n      Phone Number :-"<<h1.phno;
        cout<<"\n      Address :-"<<h1.add;
        getchar();
        cout<<"\n\n Enter New Contact :=>" ;
        cout<<"\n\n Enter the contact name :-";
        cin>>name;
        cout<<"\n Enter Phone Number of Contact :-";
        cin>>phno;
        cout<<"\n Enter Address of contact :-";
        cin>>add;
        strcpy(h1.name,name);
        h1.phno=phno;
        strcpy(h1.add,add);
        int l=f.tellg();
        f.close();
        f.open("CONTACTA",ios::out|ios::binary|ios::ate);
        f.seekg(l-sizeof(h1));
        f.write((char *)&h1,sizeof(h1));
    }
}

```

```

}while(f);
f.close();
cout << "\nCONTACT MODIFIED :\n\n";
setcolor(GREEN);
settextstyle(7,0,1);
outtextxy(250,410,"Modify more(y/n):-");
outtextxy(250,410,"\t");
ans=getchar();
if(ans=='Y' || ans=='y')
{ clrscr();
  goto d;}
else if(ans=='N' || ans=='n')
goto z;
else
goto z;
}
//*****
// CASE : 4
// DETAILS : TO DELETE RECORD
//*****
case 4:
{
  clearviewport();
  setcolor(RED);
  char name[20];
  f.close();
  f.open("video",ios::in|ios::at|ios::binary);
  settextstyle(7,0,3);
  outtextxy(200,30,"DELETING CONTACT");
  textcolor(CYAN);
  cout<<"\n\n\n\n\n\n\n Enter contact name to be deleted:\t";
  fflush(stdin);
  cin>>name;
  f.seekg(0);
  int k=0;
  do
  {
    f.close();

```

```

f.open("CONTACTA",ios::in|ios::ate|ios::binary);
f.seekg(k);
fflush(stdin);

f.read((char *)&h1,sizeof(h1));

if(f.eof())
    break;
if(strcmp(h1.name,name)==0)
{
    k+=46;
    continue;
}
else
{
    f.close();
    f.open("contact2",ios::out|ios::ate|ios::binary);
    f.write((char*)&h1, sizeof h1);
    k+=46;
}
}while(f);
f.close();
remove ("CONTACTA");
rename ("contact2","CONTACTA");
cout << "\nCONTACT DELETED :\n\n";
cout<<"\n\n\n\n\tPress Any Key...";
getch();
goto z;
}

//*****
// CASE : 5
// DETAILS : TO EXIT
//*****
case 5:
{
    fflush(stdin);
    clearviewport();
    setbkcolor(BLACK);

```

```
    setcolor(EGA_WHITE);
    window(10,10,40,11);
    settextstyle(8,0,4);
    outtextxy(50,200,"***** THANKS FOR USING *****");

    outtextxy(20,250,"#### MEET YOU AGAIN LATER ####");

    getch();
    closegraph();
    exit(1);

}
default:
{
    setcolor(DARKGRAY);
    settextstyle(0,0,2);
    outtextxy(50,350,"OUT OF ORDER !!!!") ;
    getch();
    clrscr();
    goto z;
}
}
}
```

**OUTPUT**



## *CONTACT DIARY*

CREATED By :-

SHUBHAM MEHADIA XII-B

NAVEEN KASWAN XII-B

ABHISHEK TILWAR XII-B

DHEERAJ RATHI XII-C

## WHAT IS YOUR NEED:

- 1 . ADD NEW CONTACTS
- 2 . SHOW NAMES OF SAVED CONTACTS
- 3 . MODIFY CONTACTS
- 4 . DELETE CONTACTS
- 5 . EXIT

*ENTER YOUR CHOICE:*

### *NEW CONTACT*

ENTER DETAILS ABOUT CONTACT YOU WANT TO ADD :=>

Enter Name of contact :=> Ankush

Enter Phone/Mobile number :=> 98345672

Enter Address of contact :=> Itwari

CONTACT SAVED :

*Want to add more(y/n):*

## *NAMES OF CONTACTS*

Name

-----

Ankush  
Abhi jeet  
Govinda  
Hitvi  
Jai  
Nikunj  
Parineeta  
Sophia  
Varun  
Vickey

*Want to see details of a specific contact(y/n) :*

## *DETAILS OF CONTACT*

Enter contact name for checking details of it :- Ankush

Name :- Ankush  
Phone number :- 98345672  
Address :- Itwari

*Check more details(y/n) :-*

## *MODIFYING CONTACT*

Enter contact name you want to modify : Ankush

Name :-Ankush  
Phone Number :-98345672  
Address :-Itwari

Enter New Contact :=>

Enter the contact name :-Arvind

Enter Phone Number of Contact :-88898765

Enter Address of contact :-Gokulpeth

CONTACT MODIFIED :

*Modify more(y/n):-*

## *DELETING CONTACT*

Enter contact name to be deleted:- Arvind

CONTACT DELETED

Press Any Key...

\*\*\*\*\* THANKS FOR USING \*\*\*\*\*  
#### MEET YOU AGAIN LATER ####

# **BIBLIOGRAPHY**

**The content required for the completion of the project has been taken from:-**

- 1. Introduction To C++ ( by Sumita Arora )**
- 2. [www.cplusplus.com](http://www.cplusplus.com)**