**“DIGITAL PERIODIC TABLE”**

***A***

***Minor Project-1 Report***

*Submitted in partial fulfillment of the*

*Requirements for the award of the degree of*

# BACHELOR OF TECHNOLOGY

# in

# COMPUTER SCIENCE & ENGINEERING

**By**

|  |  |  |
| --- | --- | --- |
| Ashutosh Sharma | 500069142 | R164218018 |
| Akhilesh Tomar | 500068246 | R164218008 |
| Sushant Kumar | 500068977 | R164218077 |

***Under the guidance of***

**Dr. Rashmi Sharma**

**Assistant Professor (SG)**

**Department of Systemics**



**Department of Systemics**

**School of Computer Science**

**University of Petroleum & Energy Studies**

**Bidholi, Via Prem Nagar, Dehradun, UK**

**2020-21**

# CANDIDATE’S DECLARATION

We here by certify that the project work entitled **“Digital Periodic Table”** in partial fulfillment of the requirements for the award of the Degree of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE AND ENGINEERING** with specialization in **“Internet of Things and Smart Cities”** submitted to the School of Computer Science, Department of Systemics, University of Petroleum & Energy Studies, Dehradun, is an authentic record of our work carried out during a period from **August**-**2020** to**November**-**2020** under the supervision of **Dr. Rashmi Sharma, Assistant Professor(SG), Dept. of Systemics**.

Ashutosh SharmaR164218018

Akhilesh TomarR164218008

Sushant Kumar R164218077

The matter presented in this project has not been submitted by us for the award of any other degree of this or any other University.

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

**Dr. Rashmi Sharma Dr. Neelu J. Ahuja**

Project Guide HoD- Systemics

UPES, Dehradun UPES, Dehradun

# ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to our teacher, **Dr. Rashmi Sharma,** as well as our Activity Coordinator, **Dr. Alok Aggarwal,** who gave us the opportunity to do this project on the topic **DIGITAL PERIODIC TABLE**. This work would not have been possible without their support and valuable suggestions.

We sincerely thank to our respected HoD of the Department, **Dr.Neelu Jyoti Ahuja**,for her support in doing our project at School of Computer Science.

We are also grateful to **Dr. Manish Prateek,** Dean-SoCS, UPES for giving us the necessary facilities to carry out our project work successfully.

We would like to thank all our **friends** for their help and constructive criticism during our project work. Finally we have no words to express our sincere gratitude to our **parents** who have shown us this world and for every support they have given us.

**ABSTRACT**

**Background:** The periodic table is fundamental to understand the behavior of the elements and their subsequent reactions. But many people finds searching through a traditional periodic table time consuming and unable to find the required information easily.

**Objective:** A program that helps a user to retrieve information about any element. In it, a user can update the information of any element and can also add the information about any new element. It shows properties of every element separately. It helps to find the information about elements easily and quickly just by a simple search. There are various options in the search menu to make it easy for user to find the information.

**Methods:** This program will use the concept of file handling to store and search the information of different elements. It will be implemented in C language.

**Results:** Program to search and store the information about various elements in periodic table.

**Conclusion:** This project aims to develop a periodic table which is easy to use and give information quickly to the user. By using this project anyone can find the properties of an element rapidly with maximum efficiency.

**TABLE OF CONTENTS**

**S. No. Contents Page No.**

1. Introduction 6-7

2. Literature Review 9

3. Methodology 10-13

3.1 Algorithm 12-13

3.2 Flow Chart 13

3.3 Code 14-25

3.4 Outputs 25-29

4. Conclusion 30

5. References 31

**LIST OF FIGURES**

**S. No. Name of the Figure Page No**

1. Iterative Waterfall Model 11

2. Flowchart 13

3. Screenshots 25-29

## 1.INTRODUCTION

**1.1 A brief Introduction about Digital Periodic Table.**

The project is a simple console application built without the use of graphics. It is developed using the C programming language for the purpose of storing name, symbol, atomic number, atomic weight, and some important properties as well as to display them as per requirement of the user.This project will help students to learn properties of all the elements such as atomic number, atomic mass etc. in a easier way. It provide information of only that element which a user wants, whereas in a traditional Periodic Table all elements are there at one place which creates a lot of confusion for searching but here a user can just enter only element name and get the required Information. It will help in learning Periodic Table properties in an efficient and digital mode.

### 1.2 Motivation of Work

The Periodic Table is for many the symbol of Chemistry. It is a single image that contains all of the known elements in the universe combined into an easily readable table. There are many patterns present in the table as well. It is very useful for students and many people working in the field of chemistry. The motivation to do this project is to provide a fast and efficient way to search about the information of different elements. Through this anyone can find properties of elements easily without any confusion.

**1.3 Objectives**

* **Storage of Element Information:** In the project, you can add information of any element with its name, symbol, atomic number, atomic weight and its some important properties. When new element information is to be added to this Modern Periodic Table, you have to enter 1 in the main menu and input information in given format.
* **Exploration of element Information:** Another main function of this project is to explore or to display the stored information. You can search an element by using any of the following method:
  1. By name of element
  2. By symbol of element
  3. By atomic number of element
  4. By atomic weight of element

### 1.4 Problem Statement

* Many students got confused in Inorganic Chemistry due to lack of information regarding elements properties including both chemical and physical properties.
* Nowadays, everyone wants the information quickly and in an easy way whereas reading a periodic table is time consuming.
* Since there are many numbers of elements in the periodic table with their numerous properties it is very difficult to learn all of them.
* If a new element got discover then we have to create the periodic table again and again in order to add the new element.

**Organization of the Report**

Apart from this introductory chapter which covered introduction, motivation of the work, objectives and problem statements, chapter 2 covers the few of the major works done earlier by students/academicians who have done similar works. Chapter 3 gives the methodology of the proposed work. Flow charts and Algorithms are given in this chapter. Chapter 4 gives the obtained results and future work. Finally, chapter 5 concludes the work.

**2. Literature Review**

* In [1], the Periodic Table conceived in 1869 by Mendeleev (1834–1907) became the basic tool for studying the chemical elements and their compounds. The Table improved through the years to become an essential guide for chemists, physicists, and metallurgists in their work. The history of the Periodic Table is the history of chemistry and physics during the nineteenth and twentieth centuries. Mendeleev predictions of undiscovered elements were seriously considered by chemists and as a result a number of elements were discovered during his life time and few years later.
* In [2], there has been a further change to the overall form of the periodic table. This change is somewhat analogous to the change from the short to the medium-long form in that the inner transition elements, formerly called the rare earths, have been removed to form the f-block, which is inserted between the s- and d-blocks, once again to preserve the order of increasing atomic number or often displayed as a footnote. The recent synthesis of elements up to and including element -118, with the exception of element 117, has led to speculation that the periodic table is due to undergo a further expansion to accommodate the g-block elements which will begin, at least formally, at element -121.
* In [3], a digital periodic table that searches an element using search by atomic name, atomic number, atomic weight and atomic symbol. The project is built using arrays in C language to create digital periodic table showing information about different elements.

## 3. Methodology

This project is implemented using file handling in C i.e. creating a file and accessing the stored data in the file, modifying and removing the stored data.

The main **functions used** in designing the Digital Periodic Table project are:

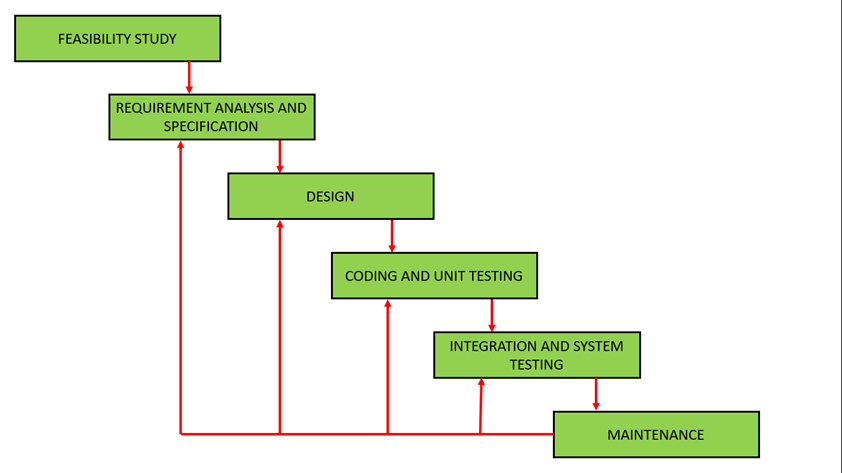
* **void add():** This function is used to input or add the information of new element to the program.
* **void explore():** This function is used to explore the stored information in the file created.
* **void mainscreen():** This function is used to print the main screen or menu of the project.

You can search an element by using any of the following method:

* 1. By name of element
  2. By symbol of element
  3. By atomic number of element
  4. By atomic weight of element

If you press 3 in the main menu, the program will be terminated.

**Iterative Waterfall Model-**



**Figure 1: Iterative Waterfall Model**

Iterative Waterfall Model

* Iterative Waterfall Model is the extension of the Waterfall model with some modifications made to improve the performance of the software development.
* The iterative waterfall model provides customer’s feedback paths from each phase to its previous phases in order to change the requirements and perform some modifications, if necessary.
* This model reduces the developer’s effort and time required to detect and correct the errors.
* In iterative waterfall model, next phase can only begin when the previous phase is completed as waterfall model.
* This model is very easy to understand and use.
* Customer involvement is not required during the software development

### Algorithm

**START**

Step 1. Create new structure element

Step 1.1 Declare structure members: name, symb, atms, block, prop

Step 1.2 Declare structure variables p and q

Step 2. Declare a file pointer fp

Step 3. function mainscreen():

Step 3.1 Calls system(“cls”) to clear screen

Step 3.2 Display “Digital Periodic Table”

Step 4. function add():

Step 4.1 Calls mainscreen()

Step 4.2 Declare char ch

Step 4.3Display “Enter details of the element”

Step 4.4 Take input of name, symbol, at\_num, at\_wt, at\_conf, block, prop

Step 4.5 fp opens file and writes in file

Step 4.6 Display “Enter ‘y’ for next record(y/n)”

Step 4.7 if ch=y, go to step 4.3

Step 4.8 else go to step 4.1

Step 5. function print():

Display “Name”, “Symbol”, “At\_no”, “At\_wt”, “At\_conf”, “Block”, “Properties” of elements

Step 6. function explore():

Step 6.1 Calls mainscreen()

Step 6.2 Display:

Enter the corresponding no.

1.Search by NAME

2.Search by SYMBOL

3.Search by ATOMIC NUMBER

4.Search by ATOMIC WEIGHT

5.Elements of Diff. Blocks

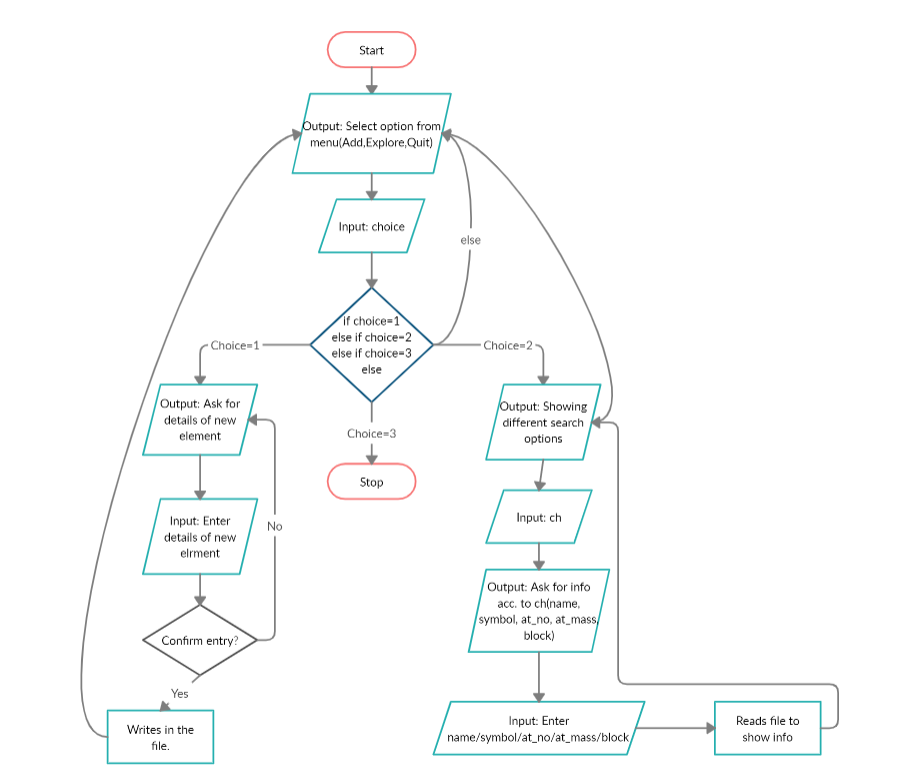
6.Return to main menu

Step 6.3 Takes the input in ‘d’

Step 6.4 Display information by choice reading from file

**STOP**

**3.2 FLOW CHART**

****

**Figure 2: Flow Chart**

* 1. **Code**

#include<stdio.h>

#include<conio.h>

#include<string.h>

#include<stdlib.h>

#include<windows.h>

#include<ctype.h>

struct element{

char name[20];

char sb[5];

int atm;

float atms;

char block;

char atc[20];

char prop[250];

}p,q;

int rw,cl;

FILE\*fp;

void add();

void explor();

void print();

void mainscreen();

void gotoxy(int x, int y)

{ static HANDLE h = NULL;

if(!h)

h = GetStdHandle(STD\_OUTPUT\_HANDLE);

COORD c = { x, y };

SetConsoleCursorPosition(h,c);

}

void textcolor(int ForgC)

{

WORD wColor;

HANDLE hStdOut = GetStdHandle(STD\_OUTPUT\_HANDLE);

CONSOLE\_SCREEN\_BUFFER\_INFO csbi;

//We use csbi for the wAttributes word.

if(GetConsoleScreenBufferInfo(hStdOut, &csbi))

{//Mask out all but the background attribute, and add in the forgournd color

wColor = (csbi.wAttributes & 0xF0) + (ForgC & 0x0F);

SetConsoleTextAttribute(hStdOut, wColor);

}

return;

}

int main()

{char d;

system("cls");

mainscreen();

label1:

textcolor(15);

gotoxy(22,15);textcolor(14);

printf("Enter the corresponding no");gotoxy(22,19);textcolor(10);

printf("1.Add new Element Information");gotoxy(22,21);

printf("2.Explore");gotoxy(22,23);

printf("3.Quit");gotoxy(22,25);

fflush(stdin);

d=getch();

switch(d)

{

case '1':

{ add();

break;

}

case '2':

{ explor();

break;

}

case '3':

{ system("cls");

mainscreen();

textcolor(14); gotoxy(30,24);

printf("THANK U");gotoxy(30,26);

printf("BYE...........");

getch();

exit(1);

break;

}

default:

{ system("cls");

mainscreen();

textcolor(12+128);gotoxy(22,11);

printf("Wrong choice");gotoxy(22,13);textcolor(15);

printf("Retype choice");

goto label1;

}

}

system("cls");

mainscreen();

goto label1;

return 0;

}

void mainscreen()

{ int i,j;

system("cls");

for(i=2,j=2;i<rw;j++)

{

if(j>15)

j=2;

textcolor(j);

gotoxy(i,2);

printf("%c",'\*');

gotoxy(i,cl-1);

printf("%c",'\*');

i++;

}

for(i=2,j=2;i<cl;i++,j++)

{

if(j>15)

j=2;

textcolor(j);

gotoxy(2,i);

printf("%c",'\*');

gotoxy(rw-1,i);

printf("%c",'\*');

}

gotoxy(30,4);textcolor(3);

printf("Modern Periodic Table");

gotoxy(37,6);textcolor(6);

printf("Digital");

gotoxy(35,7); textcolor(15);

printf("-----------");

}

void add()

{

char ch;

label1:

system("cls");

mainscreen();

gotoxy(15,14);textcolor(10);

printf("Enter the Information of Elements:");

gotoxy(15,16);

printf("Name:");

gotoxy(15,18);

printf("Symbol:");

gotoxy(15,20);

printf("Atomic No: ");

gotoxy(15,22);

printf("Atomic Wt: ");

gotoxy(15,24);

printf("Atomic Config:");

gotoxy(15,26);

printf("Block:");

gotoxy(15,28);

printf("Properties:");

textcolor(15);

fflush(stdin);

gotoxy(20,16);

scanf("%[^\n]",p.name);

p.name[0]=toupper(p.name[0]);

fflush(stdin);

gotoxy(23,18);

scanf("%[^\n]",p.sb);

p.sb[0]=toupper(p.sb[0]);

fflush(stdin);

gotoxy(25,20);

scanf("%d",&p.atm);

fflush(stdin);

gotoxy(25,22);

scanf("%f",&p.atms);

fflush(stdin);

gotoxy(29,24);

scanf("%[^\n]",p.atc);

fflush(stdin);

gotoxy(21,26);

scanf("%c",&p.block);

p.block=toupper(p.block);

if(p.block!='S'&&p.block!='P'&&p.block!='D'&&p.block!='F')

p.block=' ';

fflush(stdin);

gotoxy(26,28);

scanf("%[^\n]",p.prop);

p.prop[0]=toupper(p.prop[0]);

if((fp=fopen("data","ab+"))==NULL)

{

printf("Cannot open the file f1");

getch();

exit(1);

}

fwrite(&p,sizeof(p),1,fp);

fclose(fp);

printf("\n\n\n\t\tEnter 'y' for next record(y/n):");

ch=getch();

if(ch=='y')

{

goto label1;

}

}

void explor()

{

char d,c;

FILE \*f;

int given\_atmic\_no,a,i,tsz,n;

float given\_atmic\_mass;

int flag;

char string[20];

startofexplore:

system("cls");

mainscreen();

label6:

gotoxy(22,15);textcolor(12);

printf("Enter the corresponding no");gotoxy(22,19);textcolor(3);

printf("1.Search by 'NAME'");gotoxy(22,21);

printf("2.Search by SYMBOL");gotoxy(22,23);

printf("3.Search by ATOMIC NUMBER");gotoxy(22,25);

printf("4.Search by ATOMIC WEIGHT");gotoxy(22,27);

printf("5.Elements of Diff. Blocks");gotoxy(22,29);

printf("6.Return to main menu");

gotoxy(25,32);

fflush(stdin);

d=getch();

switch(d)

{

case '1':

{ system("cls");

mainscreen();

gotoxy(15,25);

textcolor(12);

printf("Enter the Name of Element:");

textcolor(3);

fflush(stdin);

scanf("%[^\n]",string);

printf("%s",string);

string[0]=toupper(string[0]);

if((fp=fopen("data","rb+"))==NULL)

{ system("cls");

printf("\n cannot open the record file 1");

getch();

exit(1);

}

flag=1;

while(fread(&p,sizeof(p),1,fp))

{

if(strcmp(p.name,string)==0)

{

print();

flag=0;

break;

}

}

if(flag==1)

{ system("cls");

mainscreen();

gotoxy(25,25);

textcolor(12);

printf("::No Element Available::");

}

fclose(fp);

getch();

break;

}

case '2':

{ system("cls");

mainscreen();

gotoxy(22,15);

textcolor(12);

printf("Enter the symbol:");

textcolor(3);

fflush(stdin);

scanf("%[^\n]",string);

printf("%s",string);

string[0]=toupper(string[0]);

if((fp=fopen("data","rb+"))==NULL)

{system("cls");

printf("\n cannot open the record file 1");

getch();

exit(1);

}

flag=1;

while(fread(&p,sizeof(p),1,fp))

{

if(strcmp(p.sb,string)==0)

{

print();

flag=0;

break;

}

}

if(flag==1)

{ system("cls");

mainscreen();

gotoxy(25,25);

textcolor(12);

printf("::No Element Available::");

}

fclose(fp);

getch();

break;

}

case '6':

{

return;

}

case '3':

{ system("cls");

mainscreen();

gotoxy(15,25);

textcolor(12);

printf("Enter the Atomic No. Element:");

textcolor(3);

fflush(stdin);

scanf("%d",&given\_atmic\_no);

if((fp=fopen("data","rb+"))==NULL)

{system("cls");

printf("\n cannot open the record file 1");

getch();

exit(1);

}

flag=1;

while(fread(&p,sizeof(p),1,fp))

{

if(p.atm==given\_atmic\_no)

{

print();

flag=0;

break;

}

}

if(flag==1)

{ system("cls");

mainscreen();

gotoxy(25,25);

textcolor(12);

printf("::No Element Available::");

}

fclose(fp);

getch();

break;

}

case '4':

{ system("cls");

mainscreen();

gotoxy(15,22);

textcolor(12);

printf("Enter the Atomic mass of Element:");

textcolor(3);

fflush(stdin);

scanf("%f",&given\_atmic\_mass);

if((fp=fopen("data","rb+"))==NULL)

{system("cls");

printf("\n cannot open the record file 1");

getch();

exit(1);

}

flag=1;

while(fread(&p,sizeof(p),1,fp))

{

if(p.atms==given\_atmic\_mass)

{

print();

flag=0;

break;

}

}

if(flag==1)

{ system("cls");

mainscreen();

gotoxy(25,25);

textcolor(12);

printf("::No Element Available::");

}

fclose(fp);

getch();

break;

}

case '5':

{ system("cls");

mainscreen();

gotoxy(15,25);

textcolor(12);

printf("Enter the Block:");

textcolor(3);

fflush(stdin);

scanf("%c",&c);

c=toupper(c);

if((f=fopen("temp","wb+"))==NULL)

{system("cls");

printf("\n cannot open the temp file 1");

getch();

exit(1);

}

if((fp=fopen("data","rb+"))==NULL)

{system("cls");

printf("\n cannot open the record file 1");

getch();

exit(1);

}

flag=1;

while(fread(&p,sizeof(p),1,fp))

{

if(p.block==c)

{

fwrite(&p,sizeof(p),1,f);

}

}

fclose(f);

fclose(fp);

if((f=fopen("temp","rb+"))==NULL)

{

printf("Cannot open the file");

getch();

exit(1);

}

fseek(f,0,SEEK\_END);

tsz=ftell(f);

n=(int)(tsz/sizeof(p));

for(i=0;i<(n-1);i++)

{

for(a=i+1;a<n;a++)

{

fseek(f,i\*sizeof(p),SEEK\_SET);

fread(&p,sizeof(p),1,f);

fseek(f,a\*sizeof(p),SEEK\_SET);

fread(&q,sizeof(p),1,f);

if((p.atm-q.atm)>0)

{

fseek(f,i\*sizeof(p),SEEK\_SET);

fwrite(&q,sizeof(p),1,f);

fseek(f,a\*sizeof(p),SEEK\_SET);

fflush(stdin);

fwrite(&p,sizeof(p),1,fp);

}

}

}

rewind(f);

while(fread(&p,sizeof(p),1,f))

{

print();

getch();

}

system("cls");

mainscreen();

gotoxy(25,25);

textcolor(12);

printf("::No Element Available::");

fclose(f);

getch();

break;

}

default:

{ system("cls");

mainscreen();

textcolor(12+128);gotoxy(22,11);

printf("Wrong choice");gotoxy(22,13);textcolor(15);

printf("Retype choice");

goto label6;

}

}

goto startofexplore;

}

void print()

{ system("cls");

mainscreen();

gotoxy(15,16);

printf("Name:");

gotoxy(15,18);

printf("Symbol:");

gotoxy(15,20);

printf("Atomic No: ");

gotoxy(15,22);

printf("Atomic Wt: ");

gotoxy(15,24);

printf("Atomic Config:");

fflush(stdin);

gotoxy(15,26);

printf("Block:");

gotoxy(15,28);

printf("Properties:");

textcolor(6);

gotoxy(20,16);

printf("%s",p.name);

gotoxy(23,18);

printf("%s",p.sb);

fflush(stdin);

gotoxy(25,20);

printf("%d",p.atm);

fflush(stdin);

gotoxy(25,22);

printf("%f",p.atms);

fflush(stdin);

gotoxy(29,24);

printf("%s",p.atc);

gotoxy(21,26);

printf("%c",p.block);

gotoxy(26,28);

printf("%s",p.prop);

}

**3.4 Output**

****

****

****

****

****

**4.Conclusion**

This digital periodic table design is an important achievement for students in the field of chemistry which provide them the details of elements. It is full of pattern that enable students to better understanding of elements. However, an element’s position in the table reveals a lot more about an element then the number of protons in its nucleus. We have seen that periodic table also contains a great deal of information about an element’s chemical and physical properties. The information gained from the periodic table can open up numerous windows of knowledge about the entire universe we live in. For this activity we should have a much more in depth understanding of the periodic table.

### References

[1] Habashi, Fathi (2011), “*The periodic table since Mendeleev”,* Quebec City, Canada, distributed by Laval University Bookstore.

[2]Scerri, Eric R. (2011),“*A review of research on the history and philosophy of the periodic table”,*Department of Chemistry & Biochemistry, UCLA, Los Angeles.

[3]<https://github.com/SHIKHASINGH1506/Digital-Modern-Periodic-Table/blob/master/prog.c>(Accessed on Oct 23, 2020)

[4] <https://en.wikipedia.org/wiki/Periodic_table>(Accessed on Oct 24, 2020)

[5] <https://creately.com>(Flow Chart)(Accessed on Oct 22, 2020)