/\*

\* PROBLEM STATEMENT- A digital periodic table that searches an element using search by atomic name, atomic number, atomic weight and atomic symbol.

Also displays an element's metallic property and the block in which the element is present.

\*/

/\*Header files\*/

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

/\*Function prototypes\*/

int get\_option(void);

int atomic\_number(void);

int atomic\_symbol(void);

int atomic\_name(void);

int atomic\_weight(void);

char block(int);

char property(int);

void show(int,char,char);

/\*Global Array Declarations and initialization.\*/

char name[112][20]={"Hydrogen","Helium","Lithium","Beryllium","Boron","Carbon","Nitrogen","Oxygen","Flourine","Neon",

"Sodium", "Magnesium", "Aluminium", "Silicon", "Phosphorus", "Sulphur", "Chlorine", "Argon",

"Potassium","Calcium","Scandium","Titanium","Vanadium","Chromium","Manganese","Iron","Cobalt","Nickel","Copper","Zinc",

"Gallium","Germanium","Arsenic","Selenium","Bromine","Krypton",

"Rubidium","Strontium","Yttrium","Zirconium","Niobium","Molybdenum","Technetium","Ruthenium","Rhodium","Palladium","Silver","Cadmium",

"Indium","Tin","Antimony","Tellurium","Iodine","Xenon",

"Cesium","Barium","Lanthanum","Cerium","Praseodymium","Neodymium","Promethium","Samarium","Europium","Gadolinium","Terbium","Dysprosium",

"Holmium","Erbium","Thulium","Ytterbium","Lutetium",

"Hafnium","Tantalum","Tungsten","Rhenium","Osmium","Iridium","Platinum","Gold","Mercury","Thallium","Lead","Bismuth","Polonium","Astatine","Radon",

"Francium","Radium","Actinium","Thorium","Protactinium","Uranium","Neptunium","Plutonium","Americium","Curium","Berkelium",

"Californium","Einsteinium","Fermium","Mendelevium","Nobelium","Lawrencium",

"Rutherfordium","Dubnium","Seaborgium","Bohrium","Hassium","Meitnerium","Darmstadtium","Roentgenium","Copernicium"};

char symbol[112][5]={"H","He","Li","Be","B","C","N","O","F","Ne",

"Na","Mg","Al","Si","P","S","Cl","Ar",

"K","Ca","Sc","Ti","V","Cr","Mn","Fe","Co","Ni","Cu","Zn","Ga","Ge","As","Se","Br","Kr",

"Rb","Sr","Y","Zr","Nb","Mo","Tc","Ru","Rh","Pd","Ag","Cd","In","Sn","Sb","Te","I","Xe",

"Cs","Ba","La","Ce","Pr","Nd","Pm","Sm","Eu","Gd","Tb","Dy","Ho","Er","Tm","Yb","Lu",

"Hf","Ta","W","Re","Os","Ir","Pt","Au","Hg","Tl","Pb","Bi","Po","At","Rn",

"Fr","Ra","Ac","Th","Pa","U","Np","Pu","Am","Cm","Bk","Cf","Es","Fm",",Md","No","Lr",

"Rf","Db","Sg","Bh","Hs","Mt","Ds","Rg","Cn"};

double weight[112]={1.00,4.00,6.94,9.01,10.81,12.01,14.00,16.00,20.00,20.18,23.00,24.31,26.99,28.08,30.97,32.06,35.45,39.09,40.07,44.95,47.86,50.94,

51.99,54.93,55.84,58.69,58.93,63.54,65.39,69.7,72.64,74.92,78.96,79.90,83.80,85.46,87.62,88.90,91.22,92.90,95.94,98.00,101.07,

102.90,106.42,107.86,112.41,114.81,118.71,121.76,127.6,126.90,131.29,13.90,137.32,138.90,140.11,140.90,144.24,145.00,

150.36,151.96,157.25,158.92,162.50,164.93,167.5,168.93,173.04,174.96,178.49,180.94,183.84,186.20,190.23,192.21,195.07,196.97,

200.59,204.38,207.20,208.98,209.00,210.00,222.00,223.00,226.00,227.00,231.03,232.03,237.00,238.03,243,244,247,247,251,252,257,258,

259,261,262,264,266,268,272,277,276,281,280,285};

/\*Main method\*/

int

main(void)

{

int i=999,option;

do

{

option=get\_option();

char ch1,ch2;

switch (option)

{

case 1: i=atomic\_number();

break;

case 2: i=atomic\_symbol();

break;

case 3: i=atomic\_name();

printf("%d",i);

break;

case 4: i=atomic\_weight();

break;

case 5: return (0);

default: system("cls");

printf("\n\n\t\t\t\tWrong Choice!\n");

printf("\n\t\t\t\tLet's try it again, Shall we ?\n\n");

}

if (i== -1)

{

system("cls");

printf("\t\t\t\t\t\tOops ! \n\t\t\t\t\tNo Such element present !\n");

printf("\t\t\t\t Make a right choice this time !\n\n");

}

else if (i==999)

continue;

else

{

system("cls");

ch1=block(i);

ch2=property(i);

show(i,ch1,ch2);

}

i=999;

}

while (option!=5);

return (0);

}

/\*Gets the choice of search from the user\*/

int

get\_option(void)

{

int option;

printf("\n\nEnter the corresponding number :\n\n");

printf("1. Search by ATOMIC NUMBER\n");

printf("2. Search by ATOMIC SYMBOL\n");

printf("3. Search by ATOMIC NAME\n");

printf("4. Search by ATOMIC WEIGHT\n");

printf("5. QUIT\n");

printf("\nEnter Your Choice : ");

scanf("%d",&option);

return option;

}

/\*Method to search element by atomic number\*/

/\*Returns the index of the element\*/

int

atomic\_number(void)

{

int n;

printf("\nEnter the Atomic Number you want to search for : ");

scanf("%d",&n);

n--;

if (n>=0 && n<112)

return n;

else

return -1;

}

/\*Method to search element by atomic name\*/

/\*Returns the index of the element\*/

int

atomic\_name(void)

{

char x[20];

int i,flag=0;

printf("\nEnter the Atomic Name you want to search for : ");

scanf("%s",x);

for (i=0;i<112;i++)

{

if (strcmp(x,name[i])==0)

{

flag=1;

break;

}

}

if (flag==1)

return i;

else

return -1;

}

/\*Method to search element by atomic symbol\*/

/\*Returns the index of the element\*/

int

atomic\_symbol(void)

{

char x[5];

int i,flag=0;

printf("\nEnter the Atomic Symbol you want to search for : ");

scanf("%s",x);

for (i=0;i<112;i++)

{

if (strcmp(x,symbol[i])==0)

{

flag=1;

break;

}

}

if (flag==1)

return i;

else

return -1;

}

/\*Method to search element by their corresponding atomic weight \*/

/\*Returns the index of the element\*/

int

atomic\_weight(void)

{

double x;

int i,flag=0;

printf("\nEnter the Atomic Weight you want to search for : ");

scanf("%lf",&x);

for (i=0;i<112;i++)

{

if (weight[i]==x)

{

flag=1;

break;

}

}

if (flag==1)

return i;

else

return -1;

}

/\*Method to find the block in which the element falls\*/

/\*Returns the index of the element\*/

char

block(int i)

{

char ch=' ';

int n=i+1;

if(n==1 || n==3 || n==4 || n==11 || n==12 || n==19 || n==20 || n==37 || n==38 || n==55 || n==56 || n==87 || n==88)

ch = 's';

else if(n==2 || (n>=5 && n<=10) || (n>=13 && n<=18) || (n>=31 && n<=36) || (n>=49 && n<=54) || (n>=81 && n<=86))

ch = 'p';

else if((n>=21 && n<=30) || (n>=39 && n<=48) || (n>=72 && n<=80) || (n>=104 && n<=112))

ch = 'd';

else

ch='f';

return ch;

}

/\*Method to find the metallic property of the element\*/

/\*Returns the index of the element\*/

char

property(int i)

{

int n=i+1;

char ch=' ';

if (n==1 || n==2 || (n>=6 && n<=10) || (n>=15 && n<=18) || (n>=34 && n<=36) || n==53 || n==54 ||n==86)

ch='n';

else if (n==5 || n==14 || n==32 || n==33 || n==51 ||n==52 ||n==85)

ch='s';

else

ch='m';

return ch;

}

/\*Displays the search result\*/

/\*Displays the Atomic number, name, symbol, weight, metallic property and block\*/

void

show(int i,char ch1,char ch2)

{

printf("\n\t\t\t\t\t HERE'S YOUR SEARCH RESULT\n\n");

printf("\n\t\t\t\t\tAtomic Number : %d\n",(i+1));

printf("\t\t\t\t\tAtomic Symbol : %s\n",symbol[i]);

printf("\t\t\t\t\tAtomic Name : %s\n",name[i]);

printf("\t\t\t\t\tAtomic Weight : %.2f\n",weight[i]);

printf("\t\t\t\t\tBlock : %c\n",ch1);

printf("\t\t\t\t\tMetallic property : ");

if (ch2=='n')

printf("Non-Metal\n");

else if (ch2=='s')

printf("Metalloid\n");

else

printf("Metal\n");

printf("\n\t\t\t\t\t\tThank You ! \n\n\n");

}