**ABSTRACT**

Our project provides a user friendly interface which allows the user to view available books in the library.

We have considered Android as our operating system. Because it is a powerful open Source that also runs on smart phones and Tablets. we use SQLite Database is embedded with in it.

The first step is when user downloads the app from the Google play store and the application will appear on their android device the users will also be able to check the availability of the books. The admin has all permissions to add the new user and add new books.

The main advantage of the application is to make it easier to access. the library system using smartphones

**ACKNOWLEDGEMENT**

I wish to express my sincere gratitude and salutations to His Holiness **Dr.SreeSreeSivakumaraswamigalu** for his unlimited blessings showered on us.

I express my sincere thanks to **Dr. M.N. Channabasappa** Director, Siddaganga institute of Technology, for providing all the facilities to carry out this mini project work.

I express my sincere thanks to **Dr.Shivakumaraiah** Principal Siddaganga institute of Technology, for providing all the facilities to carry out this mini project work.

I record my gratitude to **Dr.M.A.Jayaram** Director , Lab coordinator **Mr. T M KiranKumar** Asst Prof, Dept of MCA and **Mr. C BhanuPrakash** Asst Prof, Dept of MCA and the guide **Dr.M.A Jayaram** Director, Department of Master of Computer Applications for their guidance and suggestions in every aspect for the completion of our Mini Project.

i express my deepest thanks to all other staff members for their valuable suggestions and help during this mini- project time.

Finally, my whole thanks to my mother and Friends who were very supportive and provided us invaluable inputs and continuous help during whole course of the project work.

**INDEX**

|  |  |  |
| --- | --- | --- |
| Sl No. | Contents | Page No |
|  | Objective | 4 |
|  | Introduction | 5 |
|  | System Requirements | 6 |
| 4. | Screen Shots | 28-46 |
| 5. | Conclusion | 47 |
| 6. | References | 48 |
|  |  |  |

**INTRODUCTION**

The main purpose of the project is managing the entire activity of a library. The key is to keep track of all the information related to every book in the library. Information such as: the cost of the book, the author, the title, the total number of books available and so on. This system is much easier to work with rather than writing down each detailed description. The system will be a useful tool that tracks all the books in the library and handle the necessary information.

We were able to build an android application that helps to reduce the manual work for managing the books, its authors and readers. It helps to track all the information about books, readers and if the books issued have been issued or not. It helps in searching facilities based on factors like: books, readers, authors and publishers. It helps in increasing the efficiency of the managing books, its readers, authors and so on.

Library app is mobile app designed and developed for the use of staff.The Mca staff can use login credentials to login mobile app. keeping into mind that Library app enables the quick view of essential information such as – issued books , to check the availability books in library ,list the books for particular Mca department ,it also helps to search the books in the library by using the barcode number or by using book name .

Use: Instead of going library directly we check in app only.

**OBJECTIVE**

The basic idea of Bubble Sort is to compare two adjoining values and exchange if they are not in proper order. This Process is repeated for every pair of Adjoining values in the Array **.**Bubble sort, also known as sinking sort, is a simple sorting algorithm that works by repeatedly stepping through the list to be sorted, comparing each pair of adjacent items and swapping them if they are in the wrong order.

The pass through the list is repeated until no swaps are needed, which indicates that the list is sorted. The algorithm gets its name from the way smaller elements "bubble" to the top of the list. Because it only uses comparisons to operate on elements, it is a comparison sort. Although the algorithm is simple, it is not efficient for sorting large lists; other algorithms are better.

**SYSTEM REQUIREMENTS**

**Hardware Requirements:**

* Processor -Pentium 4, Core i3
* RAM -2 GB
* Hard Disk -40 GB
* Processor Speed -2.4 GHz

**Software Requirements:**

* Turbo C++.

**CODING**

#include<stdio.h>

#include<conio.h>

#include<graphics.h>

#include<ctype.h>

#include<stdlib.h>

#include<dos.h>

float b[10],temp1=0.0;

char d[10],s[10][10];

int a[10],num;

int gd,gm,t1,t2,b1,b2,n,HORTZ\_DIR,temp=0,left,top,right,bottom,n=10,left1,top1,right1,bottom1;

void module(void);

void module0(void);

void intmodule1(void);

void intmodule2(int num,int i);

void intsorting(void);

void intmodule3(void);

void intmodule3a(void);

void floatmodule1(void);

void floatmodule2(float num,int i);

void floatsorting(void);

void floatmodule3(void);

void floatmodule3a(void);

void charmodule1(void);

void charmodule2(char num, int i);

void charsorting(void);

void charmodule3(void);

void charmodule3a(void);

void sort(char [50][10],int);

void findMeaning(char [50][10],int);

void module4(void);

main()

{

int i,ch;

gd=DETECT;

initgraph(&gd,&gm,"c:\\turboc3\\bgi");

module0();

setbkcolor(4);

settextstyle(TRIPLEX\_FONT,HORTZ\_DIR,6);

while(1)

{

cleardevice();

setcolor(14);

settextstyle(TRIPLEX\_FONT,HORTZ\_DIR,3);

outtextxy(10,10,"1.ABOUT THE PROJECT");

outtextxy(10,40,"2.INTEGER SORTING");

outtextxy(10,70,"3.FLOAT SORTING");

outtextxy(10,100,"4.CHARACTER SORTING");

outtextxy(10,130,"5.STRING SORTING");

outtextxy(10,160,"6.EXIT");

outtextxy(10,190,"Enter your choice");

scanf("%d",&ch);

cleardevice();

switch(ch)

{

case 1: cleardevice();

module();

cleardevice();

break;

case 2:cleardevice();

outtextxy(10,10,"Enter The Array Size:");

scanf("%d",&n);

cleardevice();

intmodule3();

intmodule1();

intsorting();

cleardevice();

break;

case 3:cleardevice();

outtextxy(10,10,"Enter The Array Size:");

scanf("%d",&n);

cleardevice();

floatmodule3();

floatmodule1();

floatsorting();

cleardevice();

break;

case 4:cleardevice();

outtextxy(10,10,"Enter The Array Size:");

scanf("%d",&n);

cleardevice();

charmodule3();

charmodule1();

charsorting();

cleardevice();

break;

case 5:cleardevice();

left=getmaxx()/1-300;

top=getmaxy()/2-220;

right=getmaxx()/4+450;

bottom=getmaxy()/4+250;

rectangle(left,top,right,bottom);

setcolor(2);

outtextxy(350,20,"Abort");

outtextxy(500,20,"Blind");

outtextxy(350,60,"Caution");

outtextxy(500,60,"Delay");

outtextxy(350,100,"Eminent");

outtextxy(500,100,"Faint");

outtextxy(350,140,"Harsh");

outtextxy(500,140,"Giant");

outtextxy(350,180,"Misery");

outtextxy(500,180,"Naive");

outtextxy(350,220,"Keen");

outtextxy(500,220,"Labour");

outtextxy(350,260,"Rescue");

outtextxy(500,260,"Scare");

outtextxy(350,300,"Task");

outtextxy(500,300,"Unique");

setcolor(14);

outtextxy(10,10,"Enter any 10 words as shown");

outtextxy(10,50," in the box ");

printf("\n\n");

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

{

scanf("%s",s[i]);

}

setcolor(10);

settextstyle(TRIPLEX\_FONT,HORTZ\_DIR,3);

outtextxy(10,380,"Hit the key to see the meaning of above words");

delay(1000);

cleardevice;

settextstyle(TRIPLEX\_FONT,HORTZ\_DIR,6);

setcolor(14);

outtextxy(10,70," Array After Sorting ");

sort(s,n);

findMeaning(s,n);

delay(15000);

cleardevice();

break;

case 6: cleardevice();

module4();

delay(1000);

exit(0);

default:setcolor(14);

outtextxy(50,250,"Invalid choice");

break;

}

}

getch();

return;

}

void module(void)

{

setcolor(14);

settextstyle(4,0,5);

outtextxy(150,30,"About the project");

setcolor(1);

settextstyle(1,0,3);

outtextxy(80,100,"Bubble sort");

setcolor(15);

outtextxy(80,160,"->Simplest sorting algorithm");

outtextxy(80,190,"->It works by repeatedly swapping the adjacent");

outtextxy(90,210," elements if they are in wrong order");

outtextxy(80,240,"->Here is the source code of the c program");

outtextxy(100,260,"to sort");

outtextxy(150,290,"\* Integers");

outtextxy(150,320,"\* Floating Point Numbers");

outtextxy(150,350,"\* Characters");

setcolor(1);

outtextxy(450,380,"By");

outtextxy(450,410,"Shruthi.P");

outtextxy(450,440,"Vineetha");

getch();

}

void module0(void)

{

setbkcolor(10);

settextstyle(TRIPLEX\_FONT,HORTZ\_DIR,3);

setcolor(1);

outtextxy(100,130,"Practical Implementation of BubbleSort");

delay(2000);

cleardevice();

}

//--------------------------------------------------//

void intmodule1(void)

{

int i;

settextstyle(TRIPLEX\_FONT,HORTZ\_DIR,1);

setcolor(RED);

outtextxy(5,5,"Enter the integer elements");

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

{

scanf("%d",&a[i]);

cleardevice();

intmodule2(a[i],i);

intmodule3();

}

}

void intmodule2(int num,int i)

{

int x;

char str[5];

x=(i+1)\*50;

itoa(num,str,10);

outtextxy(x+20,75,str);

}

void intsorting(void)

{

int i,j,k,x,ipos,jpos;

char p[2],c;

char str[5];

cleardevice();

c=24;

p[0]=c;

p[1]=000;

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

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

{

cleardevice();

intmodule3();

ipos=(k+1)\*60;

jpos=(j+1)\*50+80;

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

delay(2000);

if(a[j]>=a[j+1])

{

settextstyle(1,HORIZ\_DIR,4);

setcolor(14);

outtextxy(400,160,"A[i]>=A[j]");

settextstyle(0,HORIZ\_DIR,1);

setcolor(15);

delay(2000);

outtextxy(400,200,"Put A[i] in temp");

setfillstyle(1,5);

floodfill(ipos,60,WHITE);

getch();

setfillstyle(1,1);

floodfill(120,420,WHITE);

delay(2000);

temp=a[j];

cleardevice();

intmodule3();

setfillstyle(1,3);

floodfill(120,420,WHITE);

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

outtextxy(400,200,"Put A[j] in A[i]");

setfillstyle(1,5);

floodfill(jpos,60,WHITE);

getch();

setfillstyle(1,1);

floodfill(ipos,60,WHITE);

delay(2000);

a[j]=a[j+1];

cleardevice();

intmodule3();

setfillstyle(1,3);

floodfill(ipos,60,WHITE);

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

outtextxy(400,200,"Put TEMP in A[j]");

setfillstyle(1,5);

floodfill(120,420,WHITE);

getch();

setfillstyle(1,1);

floodfill(jpos,60,WHITE);

delay(2000);

a[j+1]=temp;

cleardevice();

intmodule3();

setfillstyle(1,3);

floodfill(jpos,100,WHITE);

settextstyle(2,HORIZ\_DIR,8);

setcolor(2);

outtextxy(350,200,"Swapping Completed");

settextstyle(0,HORIZ\_DIR,1);

setcolor(15);

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

delay(2000);

}

}

settextstyle(2,HORIZ\_DIR,8);

setcolor(2);

outtextxy(5,250,"Iteration Completed");

printf("\n\n\n\n");

printf("Pass %d",i);

intmodule3a();

delay(2000);

}

cleardevice();

intmodule3();

intmodule3a();

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

settextstyle(2,HORIZ\_DIR,8);

setcolor(2);

outtextxy(300,200,"Sorting Completed");

delay(2000);

}

void intmodule3(void)

{

int i,x;

char ch[5];

char str[5];

settextstyle(4,HORTZ\_DIR,4);

setcolor(2);

outtextxy(20,50,"A");

settextstyle(0,HORTZ\_DIR,1);

setcolor(15);

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

{

t1=(i+1)\*50;

t2=50;

b1=t1+50;

b2=100;

rectangle(t1,t2,b1,b2);

x=(i+1)\*50+20;

itoa(a[i],str,10);

outtextxy(x,75,str);

itoa(i,ch,10);

outtextxy(x,30,ch);

}

rectangle(100,400,150,450);

outtextxy(110,470,"TEMP");

itoa(temp,str,10);

outtextxy(120,425,str);

}

void intmodule3a(void)

{

int i,x;

char ch[5];

char str[5];

settextstyle(4,HORTZ\_DIR,4);

setcolor(2);

outtextxy(-5,180,"A1");

settextstyle(0,HORTZ\_DIR,1);

setcolor(15);

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

{

t1=(i+1)\*50;

t2=180;

b1=t1+50;

b2=220;

rectangle(t1,t2,b1,b2);

x=(i+1)\*50+20;

itoa(a[i],str,10);

outtextxy(x,200,str);

itoa(i,ch,10);

outtextxy(x,160,ch);

}

}

void floatmodule1(void)

{

int i;

settextstyle(TRIPLEX\_FONT,HORTZ\_DIR,1);

textcolor(MAGENTA+BLINK);

outtextxy(5,5,"Enter the Floating elements");

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

{

scanf("%f",&b[i]);

cleardevice();

floatmodule2(b[i],i);

floatmodule3();

}

}

void floatmodule2(float num,int i)

{

int x;

char str[5];

x=(i+1)\*50;

itoa(num,str,10);

outtextxy(x+20,75,str);

}

void floatsorting(void)

{

int i,j,x,k,ipos,jpos;

char p[2],c;

char str[5];

cleardevice();

c=24;

p[0]=c;

p[1]=000;

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

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

{

cleardevice();

floatmodule3();

ipos=(k+1)\*60;

jpos=(j+1)\*50+80;

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

delay(2000);

if(b[j]>=b[j+1])

{

settextstyle(1,HORIZ\_DIR,4);

setcolor(14);

outtextxy(400,160,"A[i]>=A[j]");

settextstyle(0,HORIZ\_DIR,1);

setcolor(15);

delay(2000);

outtextxy(400,200,"Put A[i] in temp");

setfillstyle(1,5);

floodfill(ipos,60,WHITE);

getch();

setfillstyle(1,1);

floodfill(120,420,WHITE);

delay(2000);

temp1=b[j];

cleardevice();

floatmodule3();

setfillstyle(1,3);

floodfill(120,420,WHITE);

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

outtextxy(400,200,"Put A[j] in A[i]");

setfillstyle(1,5);

floodfill(jpos,60,WHITE);

getch();

setfillstyle(1,1);

floodfill(ipos,60,WHITE);

delay(2000);

b[j]=b[j+1];

cleardevice();

floatmodule3();

setfillstyle(1,3);

floodfill(ipos,60,WHITE);

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

outtextxy(400,200,"Put TEMP in A[j]");

setfillstyle(1,5);

floodfill(120,420,WHITE);

getch();

setfillstyle(1,1);

floodfill(jpos,60,WHITE);

delay(2000);

b[j+1]=temp1;

cleardevice();

floatmodule3();

setfillstyle(1,3);

floodfill(jpos,60,WHITE);

settextstyle(2,HORIZ\_DIR,8);

setcolor(2);

outtextxy(350,200,"Swapping Completed");

settextstyle(0,HORIZ\_DIR,1);

setcolor(15);

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

delay(2000);

}

}

settextstyle(2,HORIZ\_DIR,8);

setcolor(2);

outtextxy(5,250,"Iteration Completed");

printf("\n\n\n\n");

printf("Pass %d",i);

floatmodule3a();

delay(2000);

}

cleardevice();

floatmodule3();

floatmodule3a();

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

settextstyle(2,HORIZ\_DIR,8);

setcolor(2);

outtextxy(300,200,"Sorting Completed");

delay(2000);

}

void floatmodule3(void)

{

float i;

int x;

char ch[5];

char str[5];

char wholeNumStr[10];

char decNumStr[5];

int wholenum;

int decinum;

settextstyle(4,HORIZ\_DIR,4);

setcolor(2);

outtextxy(20,50,"A");

settextstyle(0,HORTZ\_DIR,1);

setcolor(15);

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

{

t1=(i+1)\*50;

t2=50;

b1=t1+50;

b2=100;

rectangle(t1,t2,b1,b2);

x=(i+1)\*50+20;

itoa(b[i],str,10);

wholenum=b[i];

decinum=(float)((b[i]-wholenum)\*10);

itoa(wholenum,wholeNumStr,10);

itoa(decinum,decNumStr,10);

strcpy(str,wholeNumStr);

strcat(str,".");

strcat(str,decNumStr);

outtextxy(x,75,str);

itoa(i,ch,10);

outtextxy(x,30,ch);

}

rectangle(100,400,150,450);

outtextxy(110,470,"TEMP");

itoa(temp1,str,10);

wholenum=temp1;

decinum=(float)((temp1-wholenum)\*10);

itoa(wholenum,wholeNumStr,10);

itoa(decinum,decNumStr,10);

strcpy(str,wholeNumStr);

strcat(str,".");

strcat(str,decNumStr);

outtextxy(120,425,str);

}

void floatmodule3a(void)

{

float i;

int x;

char ch[5];

char str[5];

char wholeNumStr[10];

char decNumStr[5];

int wholenum;

int decinum;

settextstyle(4, HORIZ\_DIR, 4);

setcolor(2);

outtextxy(-5,180,"A1");

settextstyle(0, HORIZ\_DIR, 1);

setcolor(15);

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

{

t1=(i+1)\*50;

t2=180;

b1=t1+50;

b2=220;

rectangle(t1,t2,b1,b2);

x=(i+1)\*50+20;

itoa(b[i],str,10);

wholenum =b[i];

decinum =(float)((b[i] - wholenum)\*10);

itoa(wholenum, wholeNumStr,10);

itoa(decinum, decNumStr,10);

strcpy(str, wholeNumStr);

strcat(str, ".");

strcat(str, decNumStr);

outtextxy(x,200,str);

itoa(i,ch,10);

outtextxy(x,160,ch);

}

}

void charmodule1(void)

{

int i;

settextstyle(TRIPLEX\_FONT,HORTZ\_DIR,1);

textcolor(MAGENTA+BLINK);

outtextxy(5,5,"Enter the Characters");

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

{

scanf(" %c",&d[i]);

charmodule2(d[i],i);

charmodule3();

}

}

void charmodule2(char num,int i)

{

int x;

char str[5];

char temp1[2];

temp1[0] = num;

temp1[1] = '\0';

x=(i+1)\*50;

itoa(num,str,10);

outtextxy(x+20,75,temp1);

}

void charsorting(void)

{

int i,j,x,k,ipos,jpos;

char p[2],c;

char str[5];

cleardevice();

c=24;

p[0]=c;

p[1]=000;

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

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

{

cleardevice();

charmodule3();

ipos=(k+1)\*60;

jpos=(j+1)\*50+80;

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

delay(2000);

if(d[j]>=d[j+1])

{

settextstyle(1, HORIZ\_DIR, 4);

setcolor(14);

outtextxy(400,160,"A[i] >= A[j]");

settextstyle(0, HORIZ\_DIR, 1);

setcolor(15);

delay(2000);

outtextxy(400,200,"Put A[i] in TEMP");

setfillstyle(1,5);

floodfill(ipos,60,WHITE);

getch();

setfillstyle(1,1);

floodfill(120,420,WHITE);

delay(2000);

temp=d[j];

cleardevice();

charmodule3();

setfillstyle(1,3);

floodfill(120,420,WHITE);

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

outtextxy(400,200,"Put A[j] in A[i]");

setfillstyle(1,5);

floodfill(jpos,60,WHITE);

getch();

delay(2000);

setfillstyle(1,1);

floodfill(ipos,60,WHITE);

delay(2000);

d[j]=d[j+1];

cleardevice();

charmodule3();

setfillstyle(1,3);

floodfill(ipos,60,WHITE);

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

outtextxy(400,200,"Put TEMP in A[j]");

setfillstyle(1,5);

floodfill(120,420,WHITE);

getch();

delay(2000);

setfillstyle(1,1);

floodfill(jpos,60,WHITE);

delay(2000);

d[j+1]=temp;

cleardevice();

charmodule3();

setfillstyle(1,3);

floodfill(jpos,60,WHITE);

settextstyle(2, HORIZ\_DIR, 8);

setcolor(2);

outtextxy(350,200,"Swapping Completed");

settextstyle(0, HORIZ\_DIR, 1);

setcolor(15);

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

delay(2000);

}

}

settextstyle(2,HORIZ\_DIR,8);

setcolor(GREEN);

outtextxy(5,250,"Iteration Completed");

printf("\n\n\n\n");

printf("Pass %d",i);

charmodule3a();

delay(2000);

}

cleardevice();

charmodule3();

charmodule3a();

outtextxy(ipos,120,p);

outtextxy(jpos,120,p);

outtextxy(ipos,140,"i");

outtextxy(jpos,140,"j");

settextstyle(2, HORIZ\_DIR, 8);

setcolor(2);

outtextxy(300,200,"Sorting Completed");

delay(2000);

}

void charmodule3(void)

{

int i,x;

char ch[5];

char str[5];

char temp1[2];

settextstyle(4, HORIZ\_DIR, 4);

setcolor(2);

outtextxy(20,50,"A");

settextstyle(0, HORIZ\_DIR, 1);

setcolor(15);

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

{

t1=(i+1)\*50;

t2=50;

b1=t1+50;

b2=100;

rectangle(t1,t2,b1,b2);

x=(i+1)\*50+20;

itoa(d[i],str,10);

temp1[0] = d[i];

temp1[1] = '\0';

outtextxy(x,75,temp1);

itoa(i,ch,10);

outtextxy(x,30,ch);

}

rectangle(100,400,150,450);

outtextxy(110,470,"TEMP");

itoa(temp,str,10);

temp1[0] = temp;

temp1[1] = '\0';

outtextxy(120,425,temp1);

}

void charmodule3a(void)

{

int i,x;

char ch[5];

char str[5];

char temp1[2];

settextstyle(4, HORIZ\_DIR, 4);

setcolor(2);

outtextxy(-5,180,"A1");

settextstyle(0, HORIZ\_DIR, 1);

setcolor(15);

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

{

t1=(i+1)\*50;

t2=180;

b1=t1+50;

b2=220;

rectangle(t1,t2,b1,b2);

x=(i+1)\*50+20;

itoa(d[i],str,10);

temp1[0] = d[i];

temp1[1] = '\0';

outtextxy(x,200,temp1);

itoa(i,ch,10);

outtextxy(x,160,ch);

}

}

void sort(char s[50][10],int n)

{

int i,j,cmp;

char tmp[1][10];

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

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

{

cmp=strcmp(s[j],s[j+1]);

if(cmp>0)

{

strcpy(tmp[0],s[j+1]);

strcpy(s[j+1],s[j]);

strcpy(s[j],tmp[0]);

}

}

}

void findMeaning(char s[50][10],int n)

{

FILE \*fp;

char line[124];

int i;

char \*search="=";

char \*word;

char \*meaning;

printf("\t<<Sorted Order>>\t\t\t<<Meaning>>");

printf("\n\t");

for(i=0;i<n;i++){

fp = fopen("Dict.txt","r");

setbkcolor(4);

setcolor(15);

left1=getmaxx()/1-600;

top1=getmaxy()/2+242;

right1=getmaxx()/4+400;

bottom1=getmaxy()/3+20;

rectangle(left1,top1,right1,bottom1);

while(fgets(line,sizeof(line),fp)!=NULL)

{

if(strstr(line, s[i])!=NULL)

{

word = strtok(line,search);

meaning = strtok(NULL,search);

printf("\n\t%s\t->%s",word,meaning);

}

}

fclose(fp);

}

}

void module4(void)

{

setbkcolor(10);

setcolor(1);

settextstyle(7,HORIZ\_DIR,4);

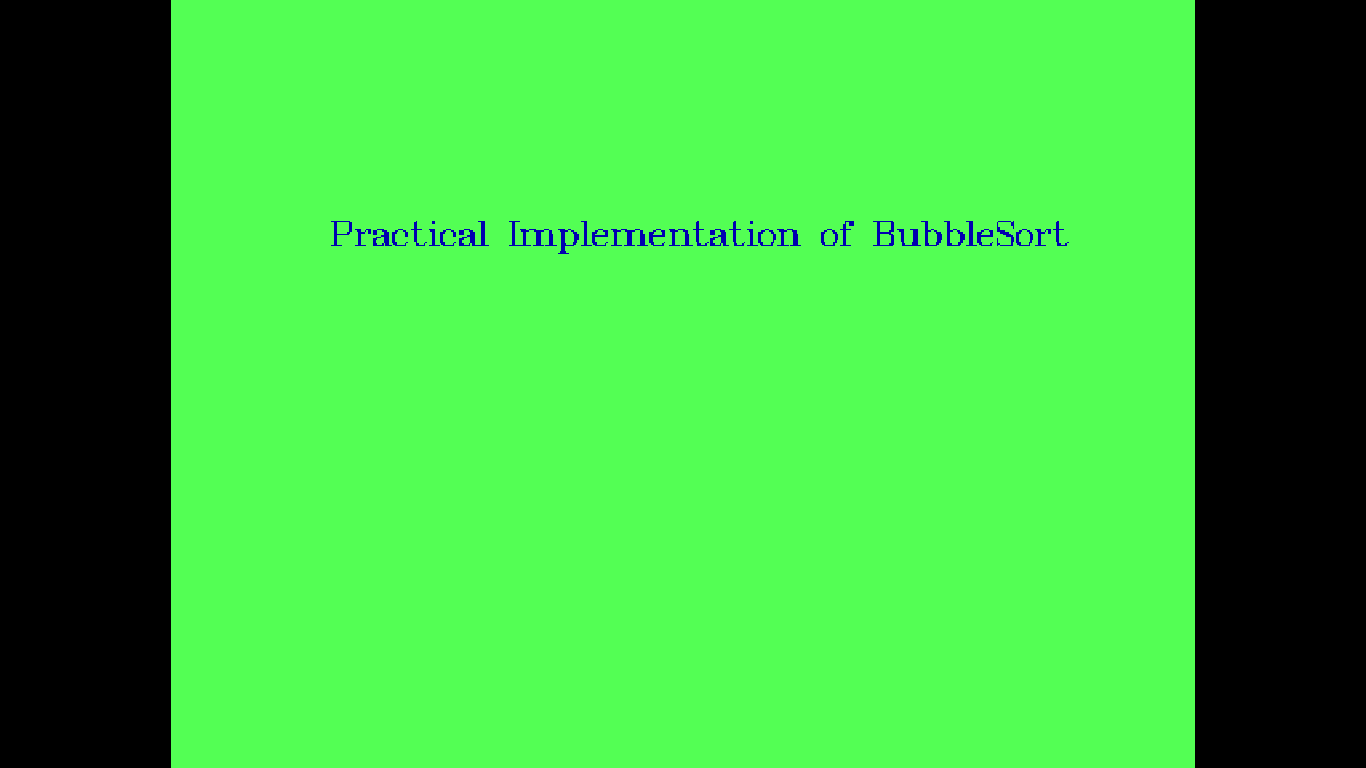
outtextxy(160,180,"!! THANK YOU !!");

outtextxy(120,280," Press any key to EXIT");

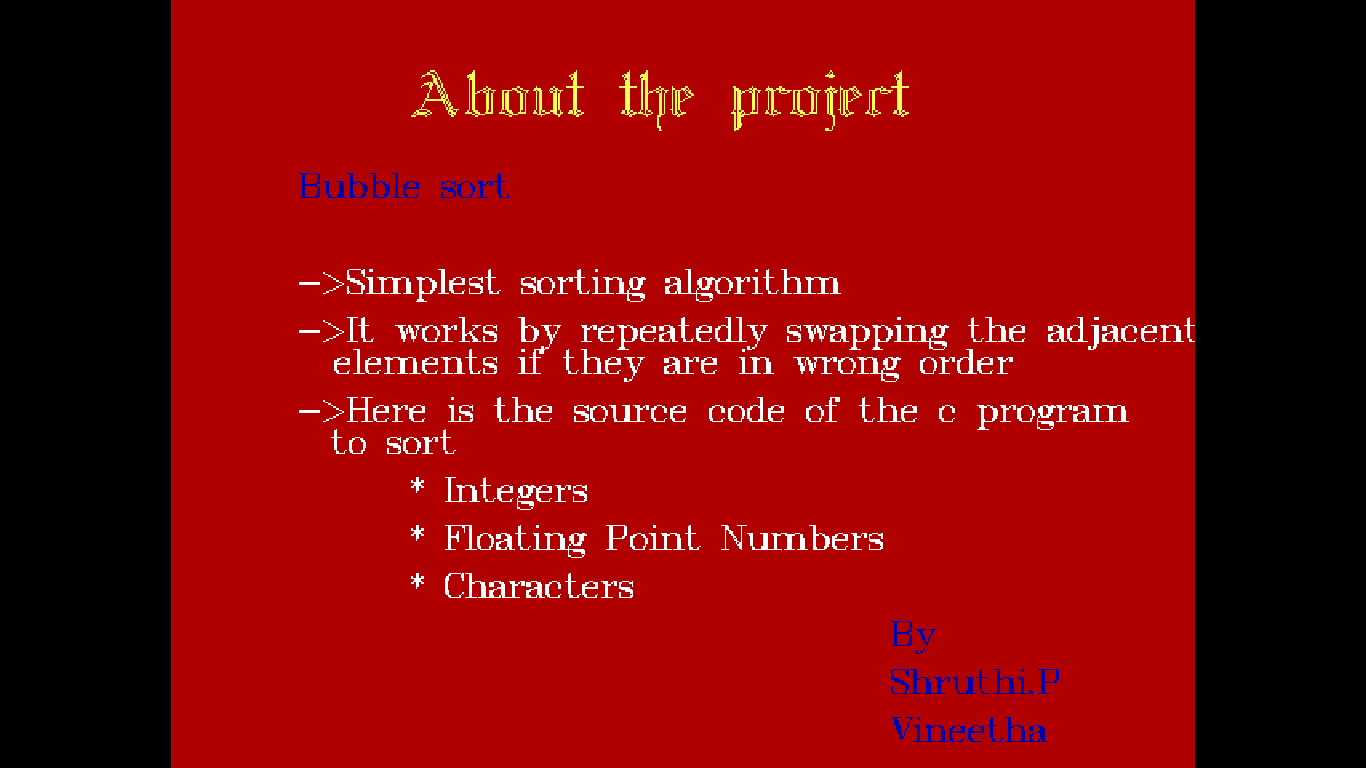
delay(2000);

}

**SCREENSHOTS**

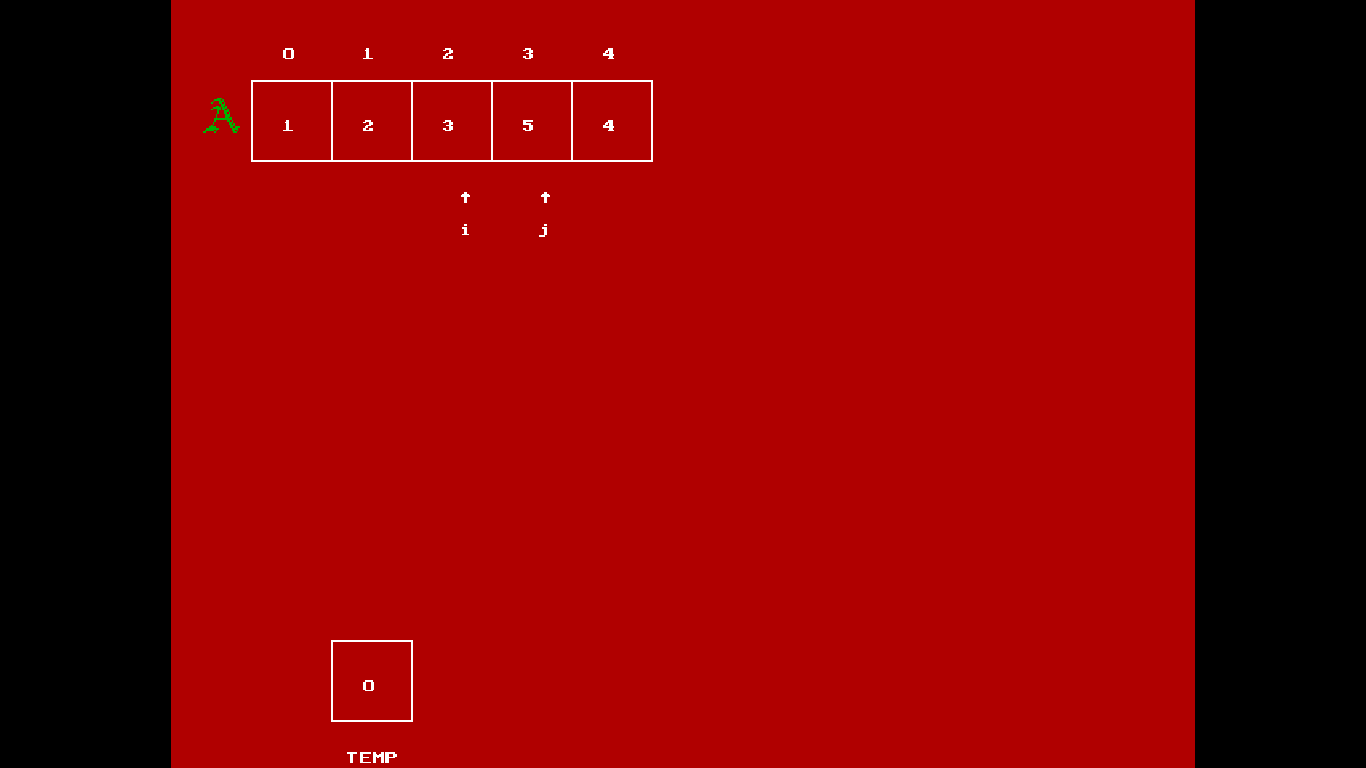


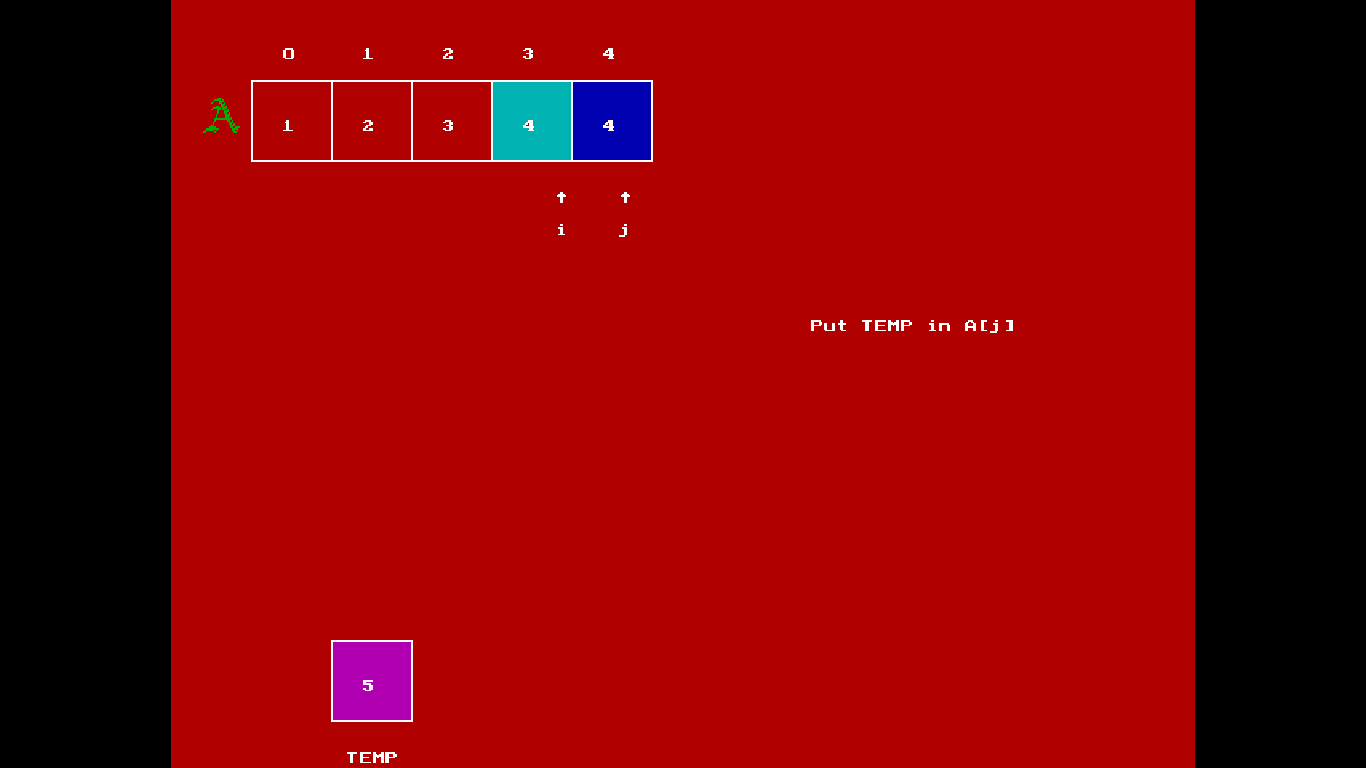




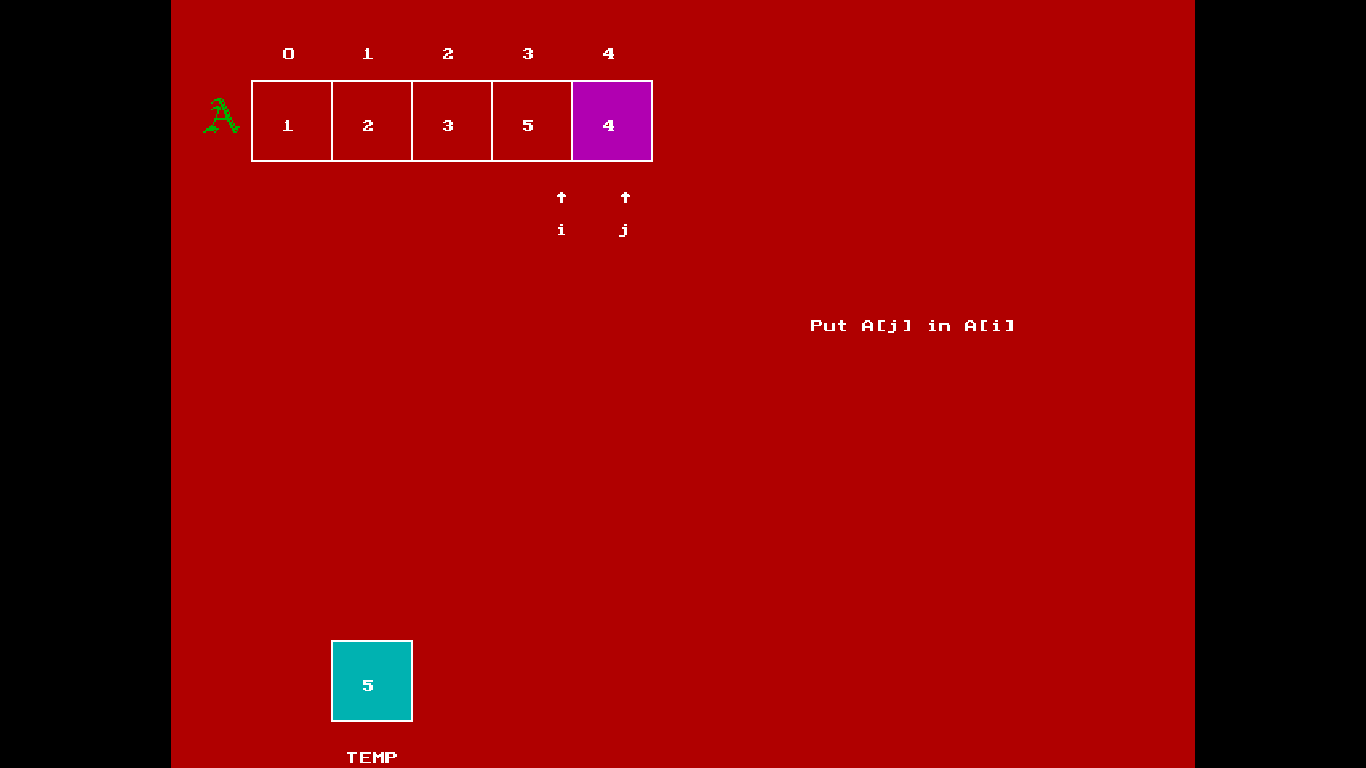


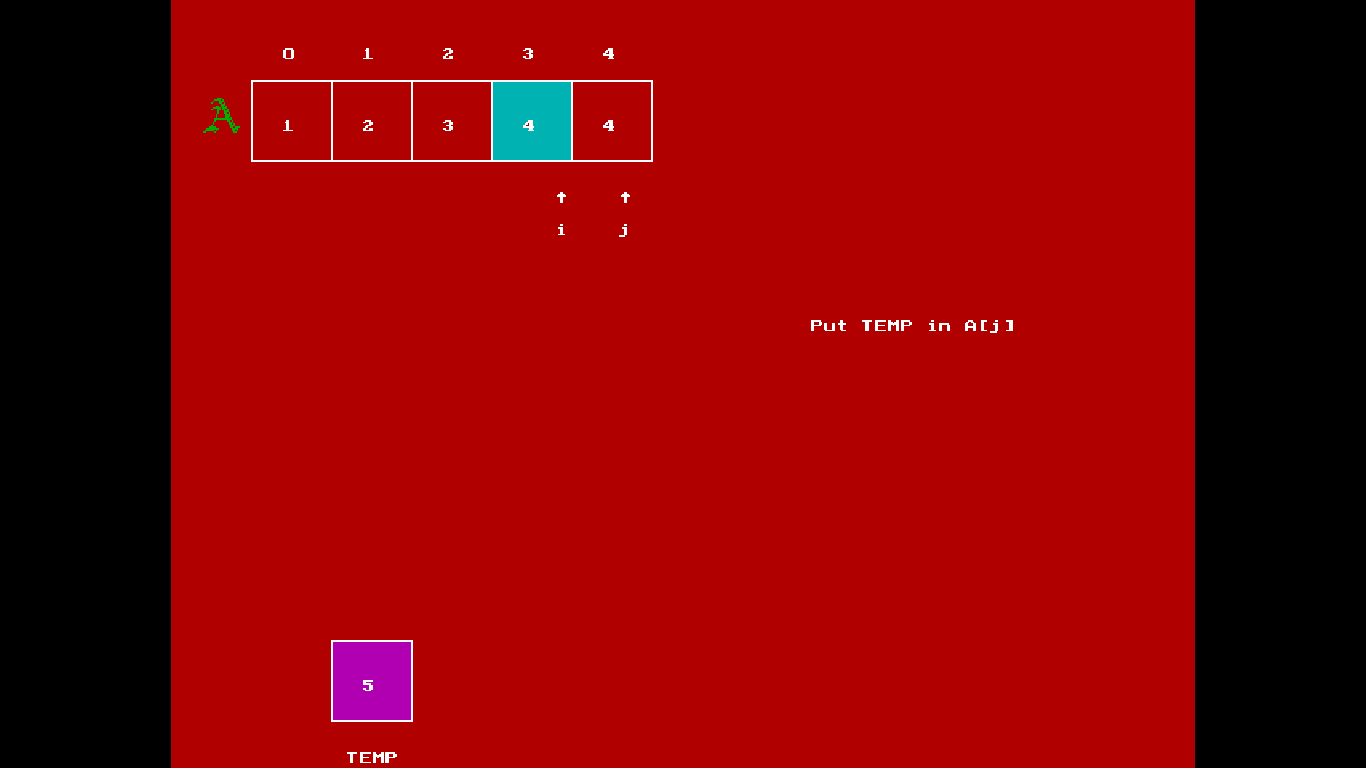


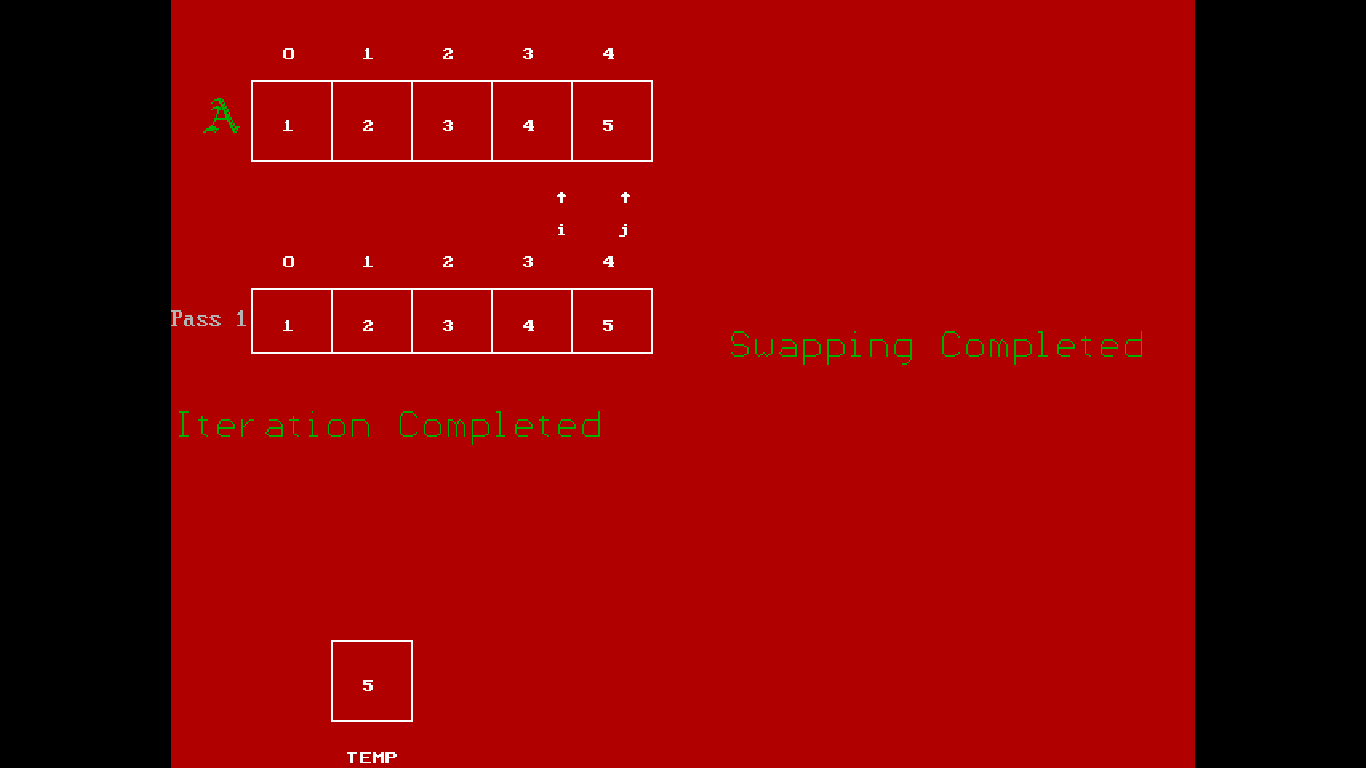


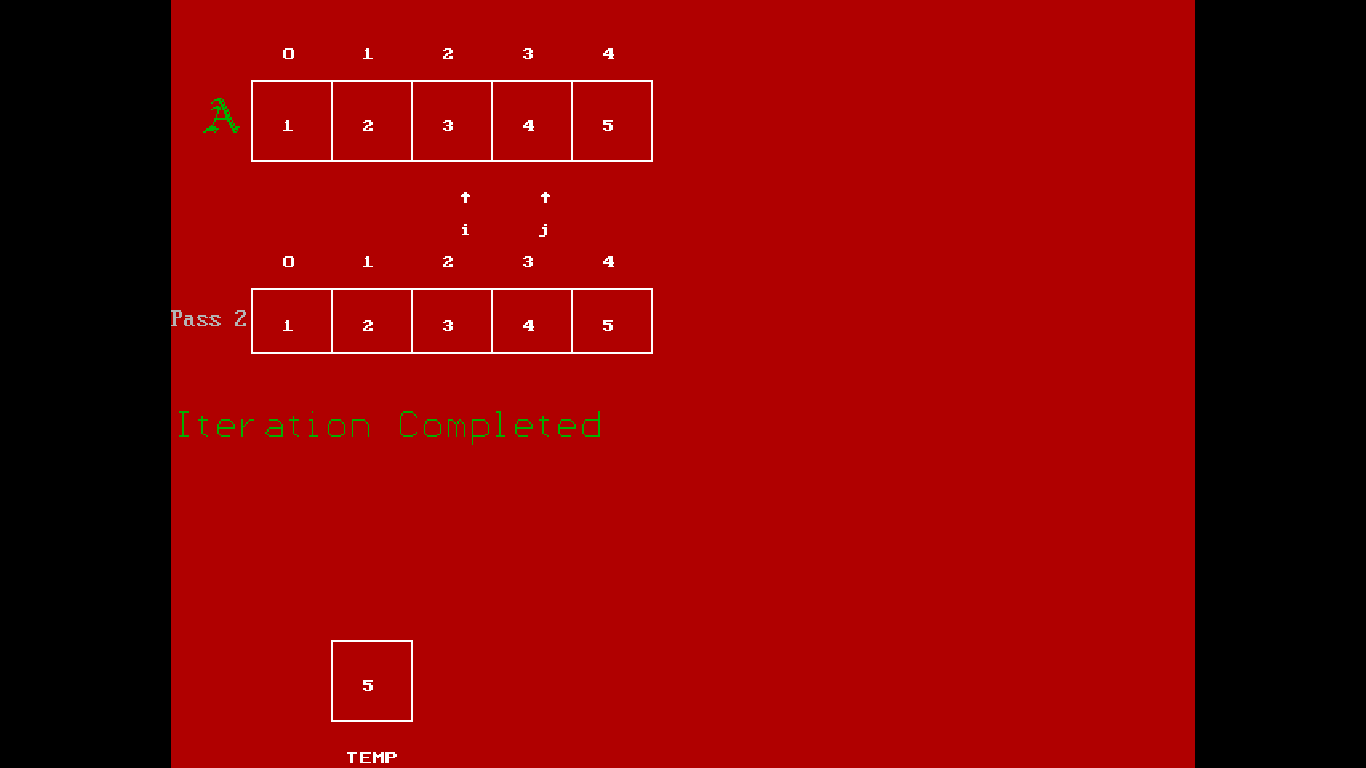


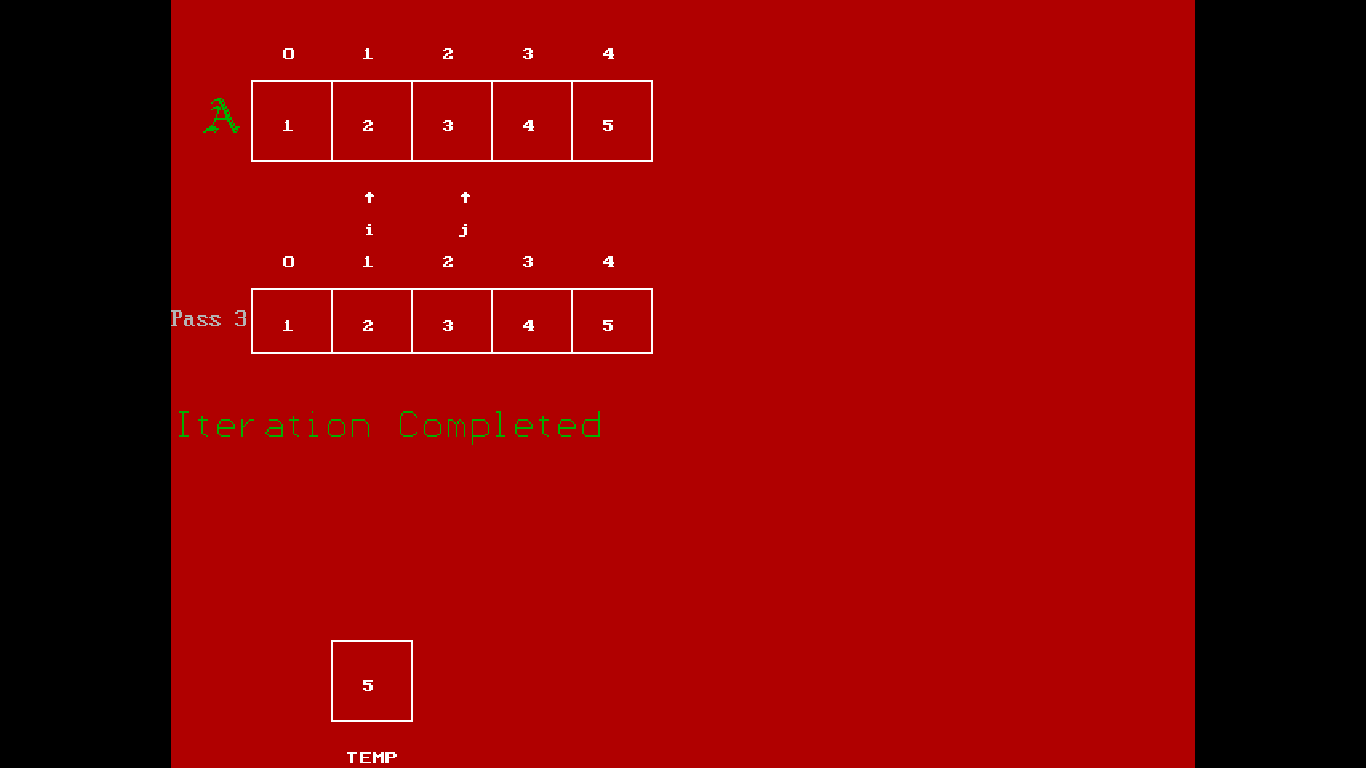
\

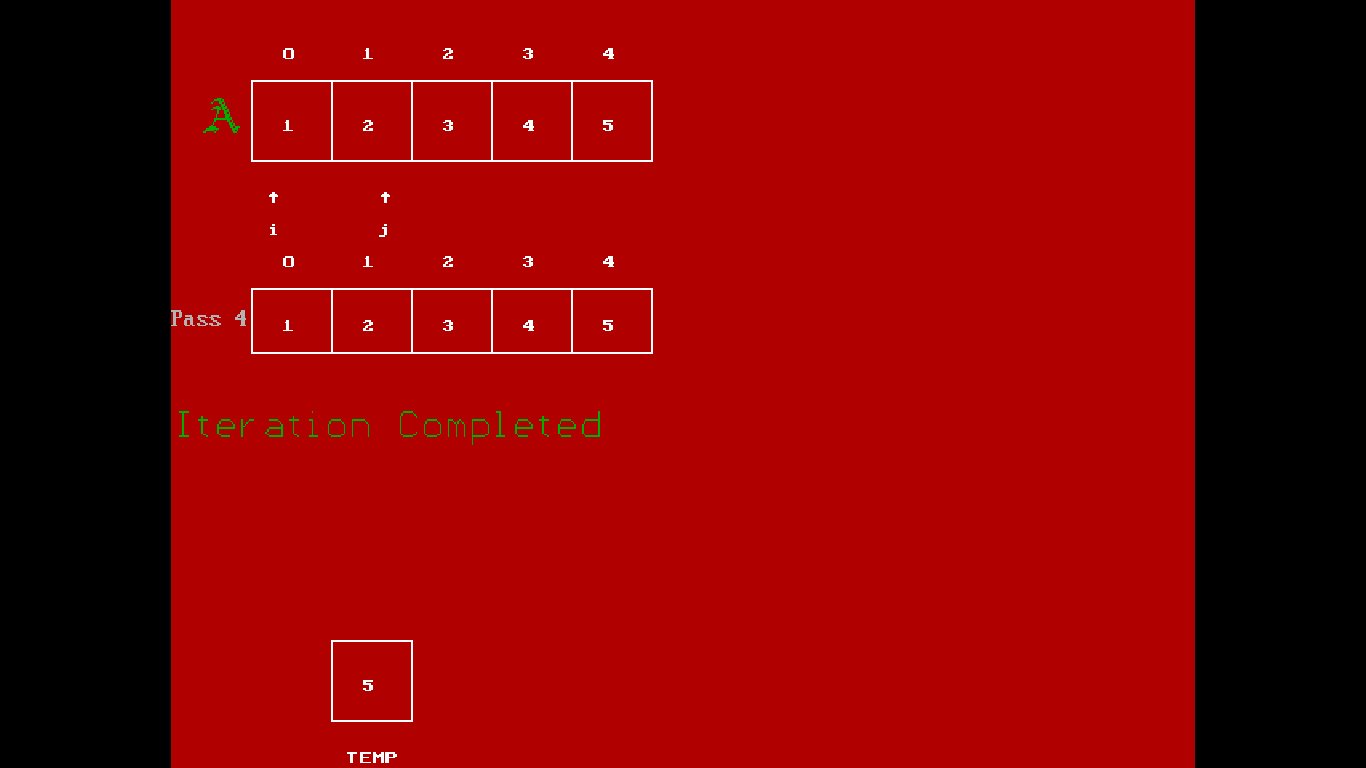


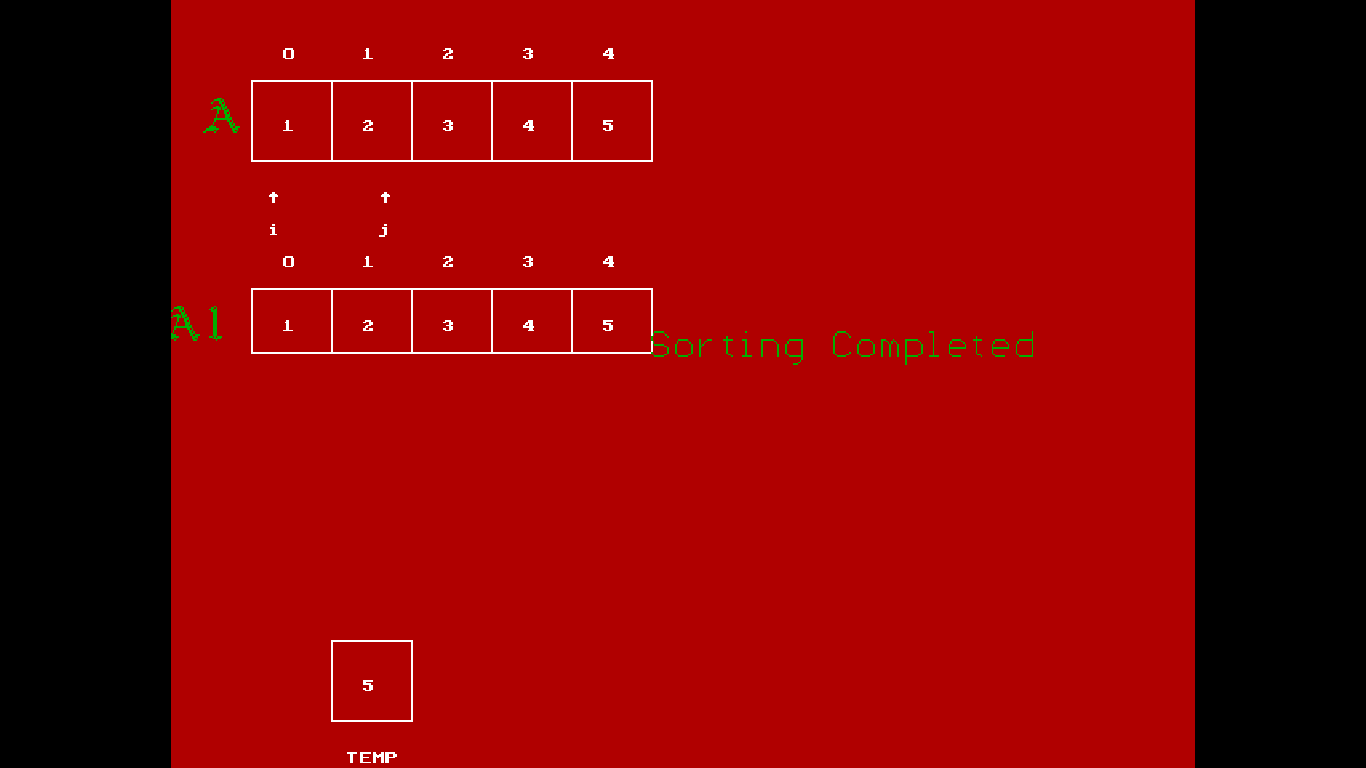






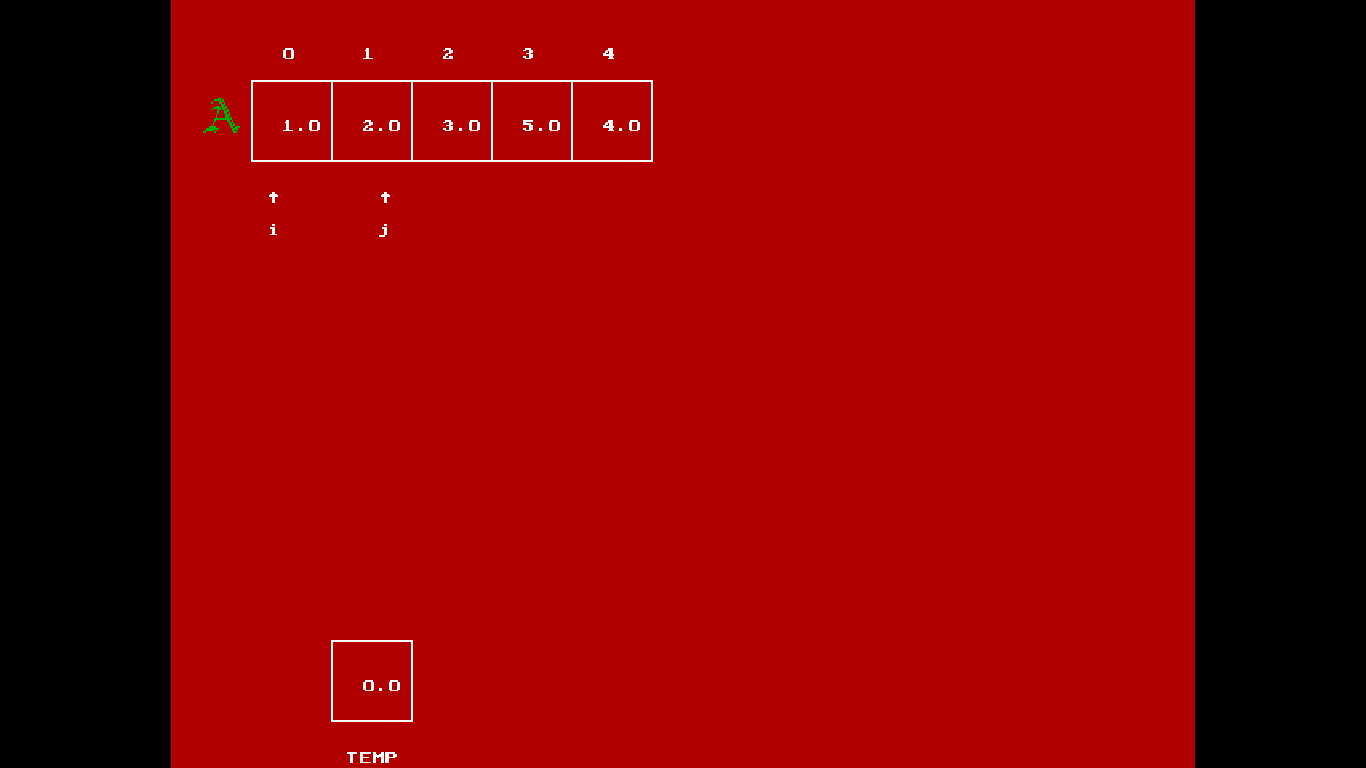


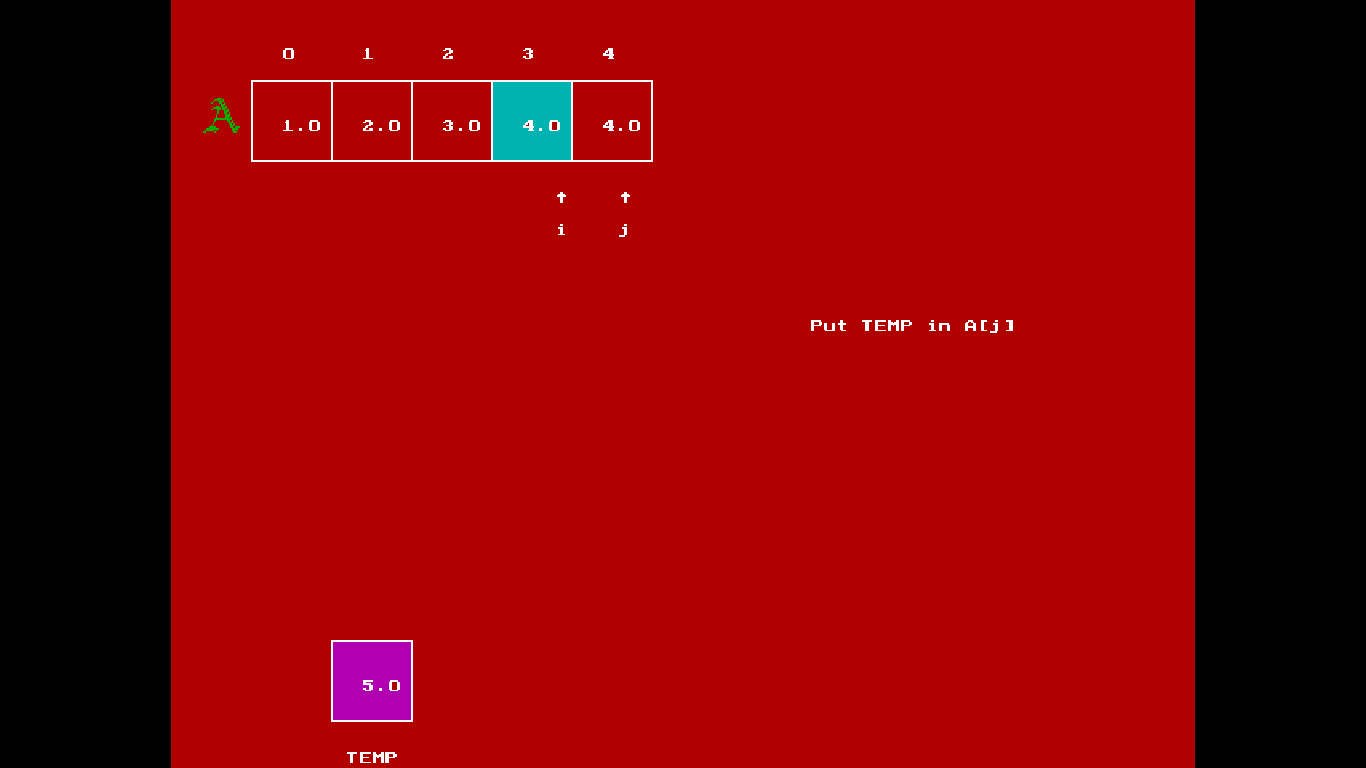


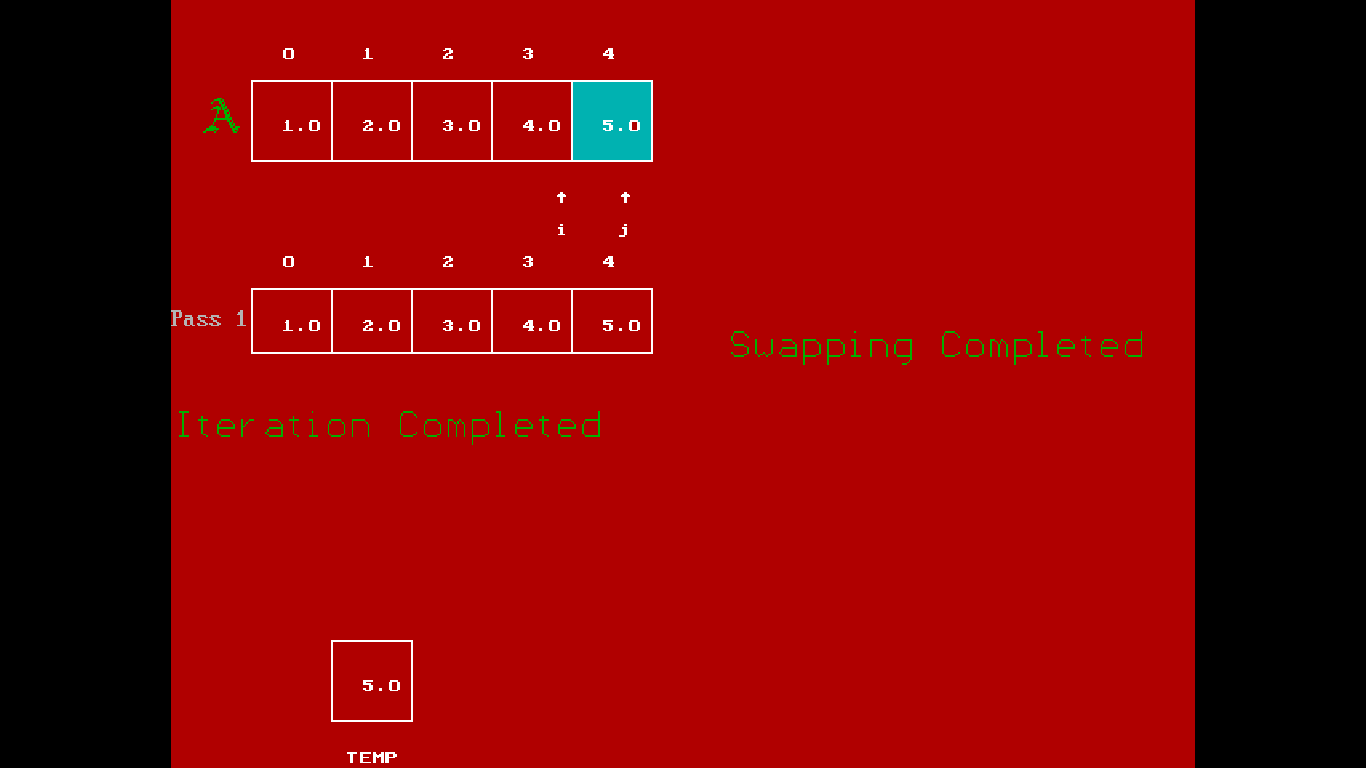


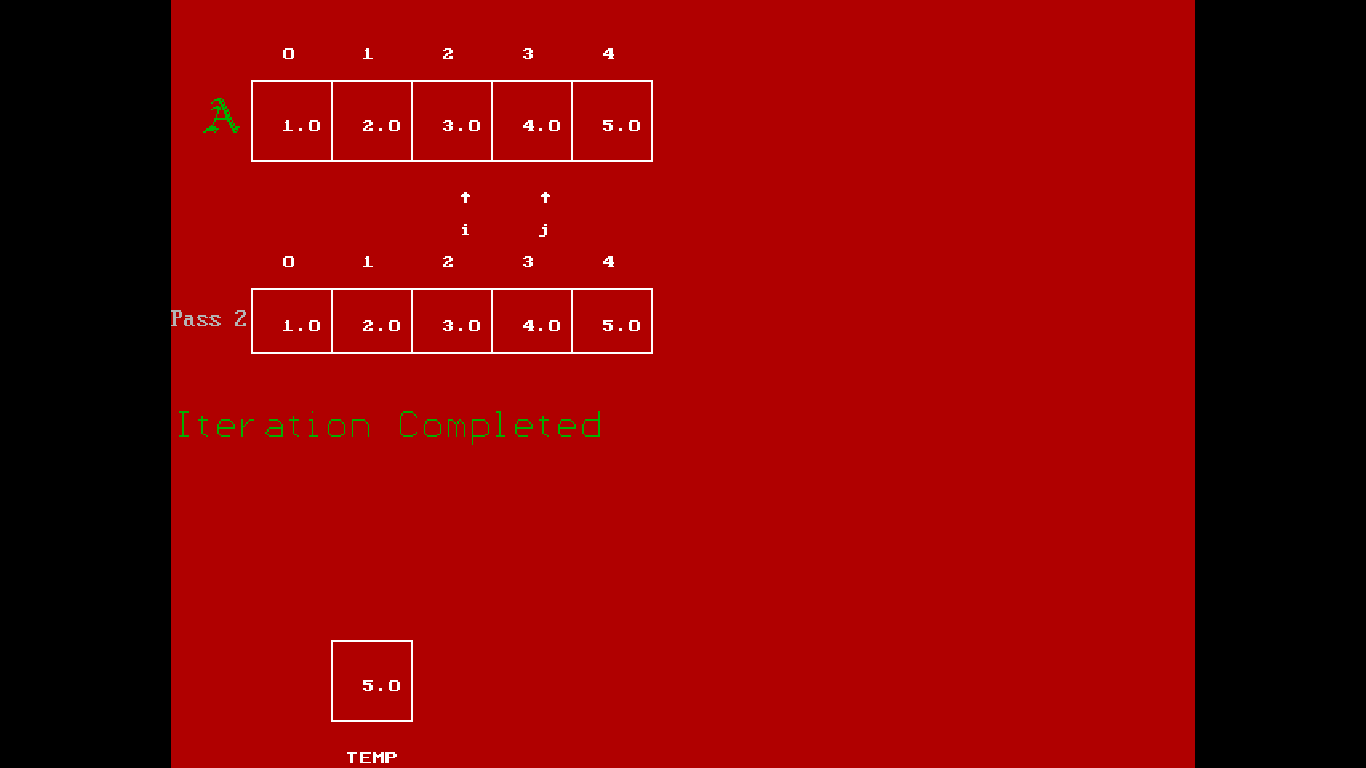


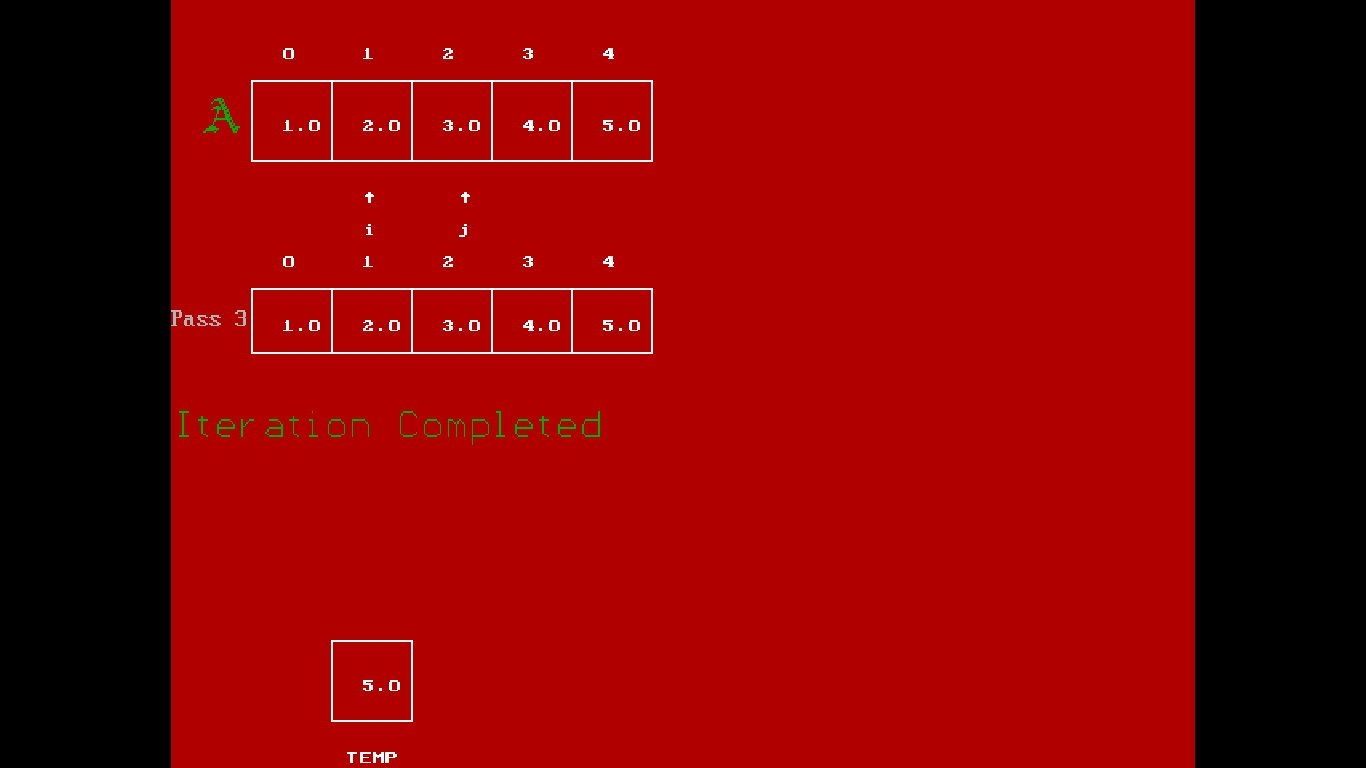


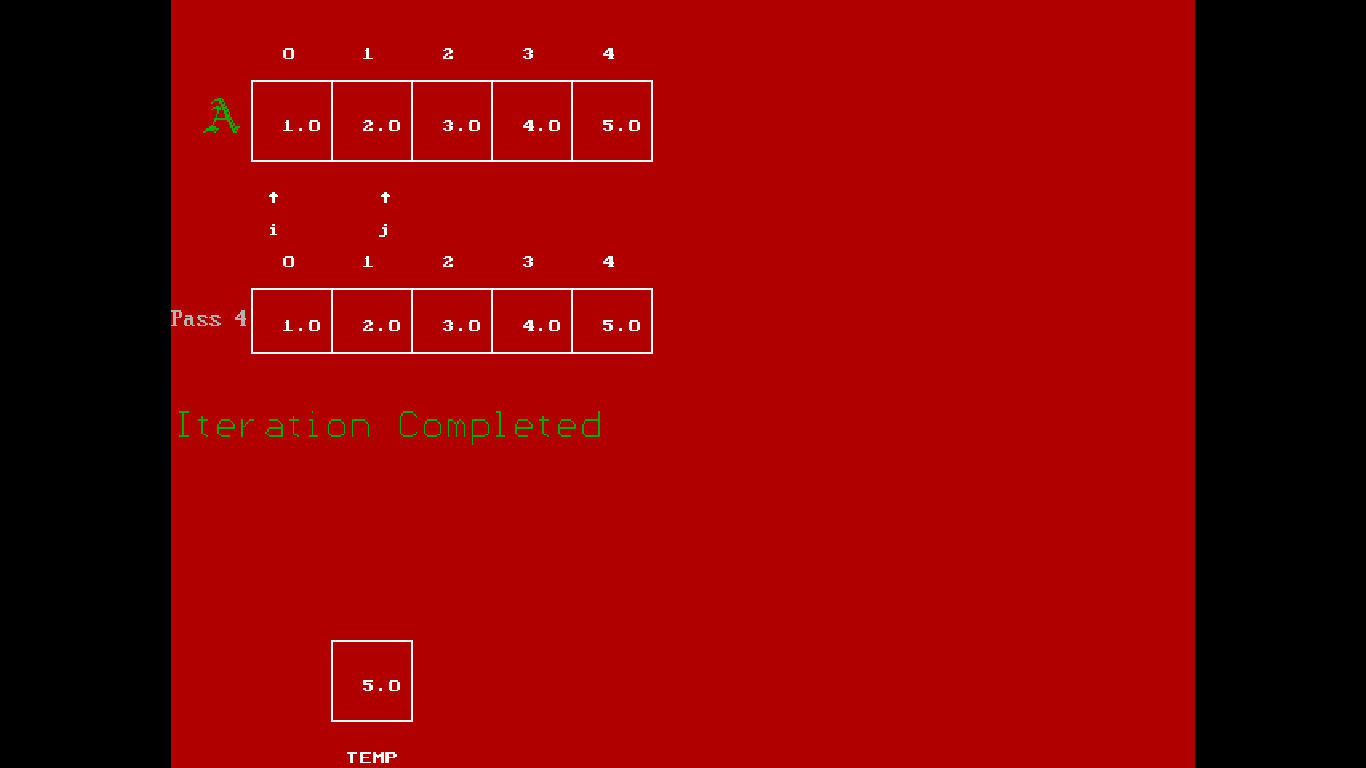


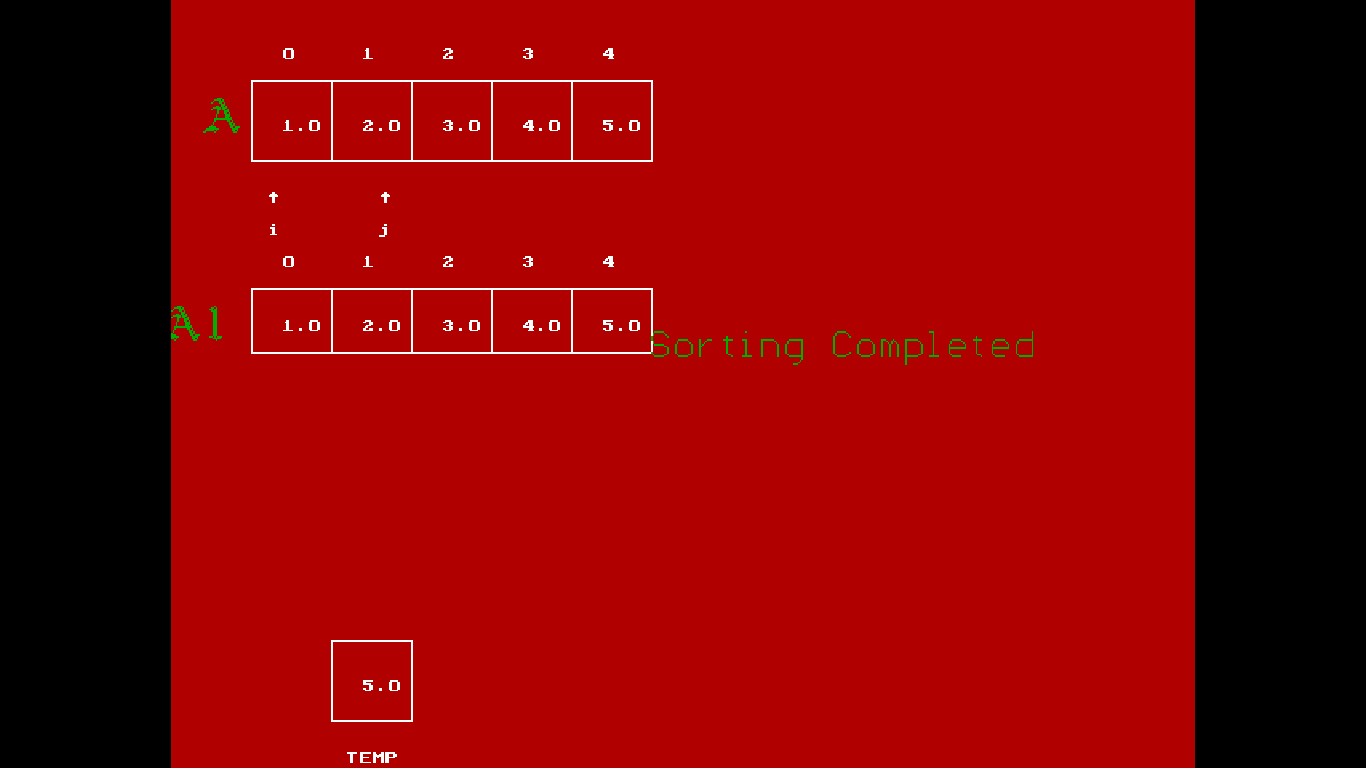






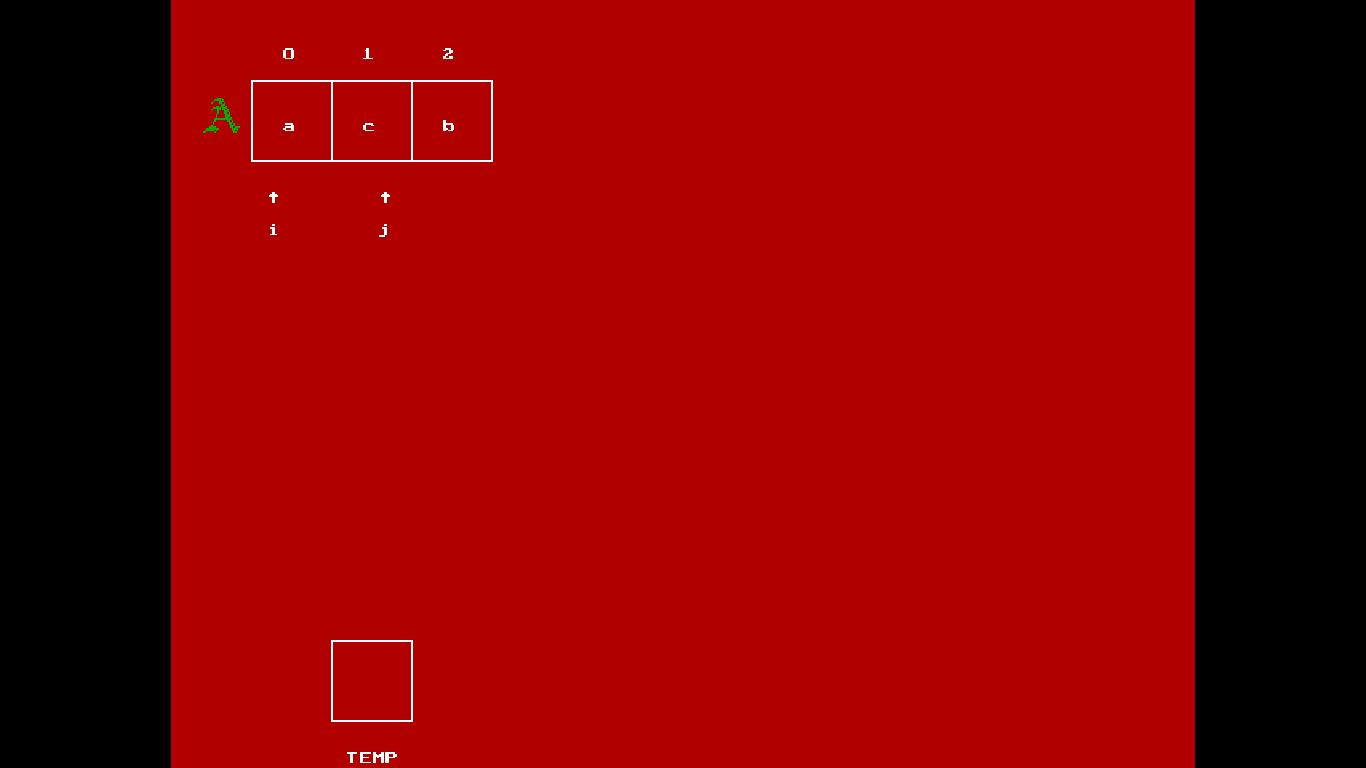


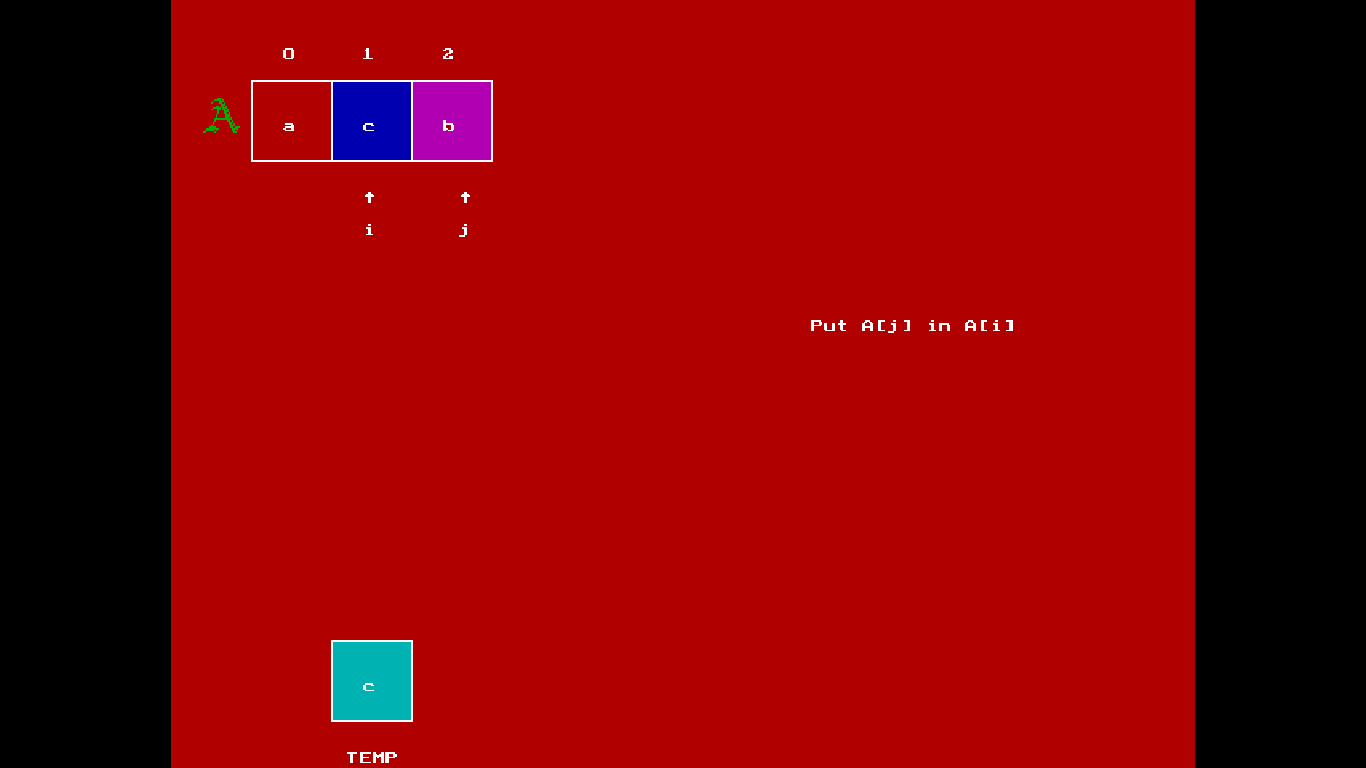


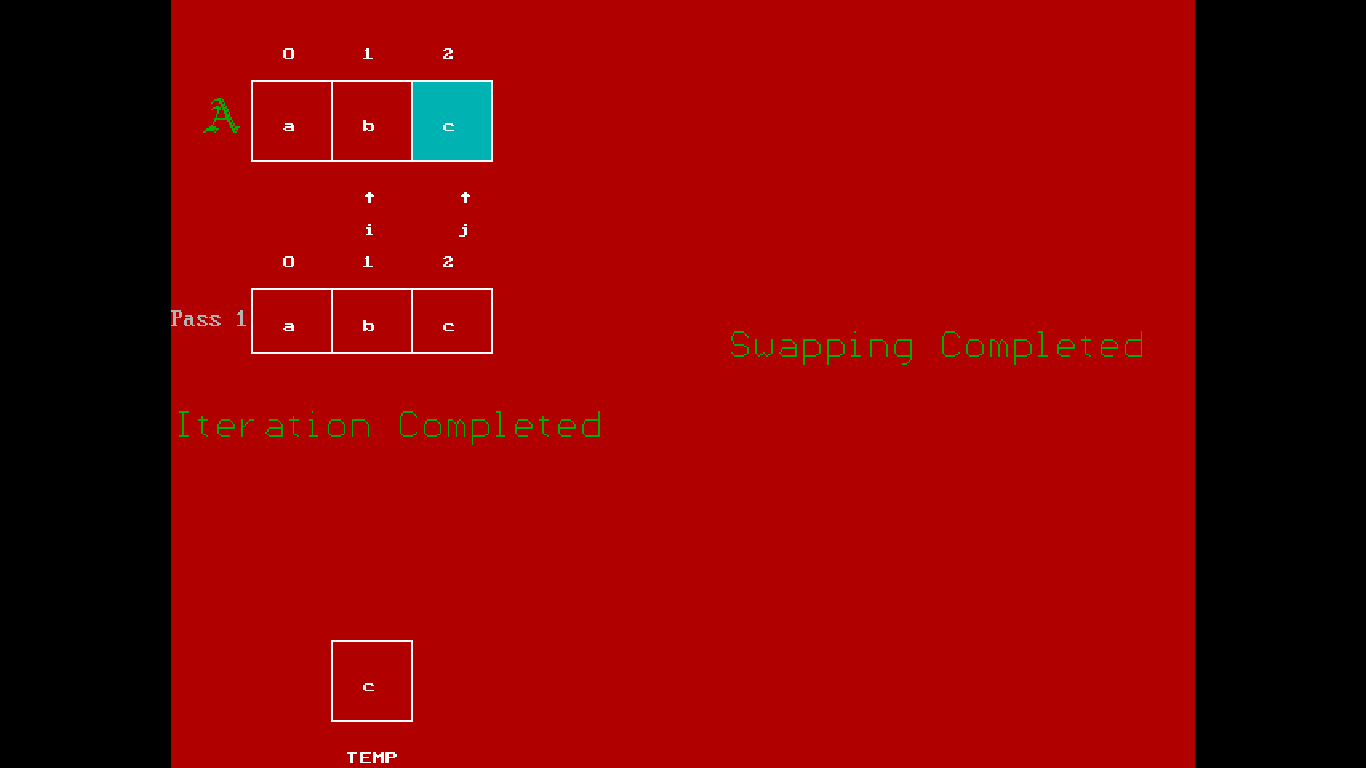


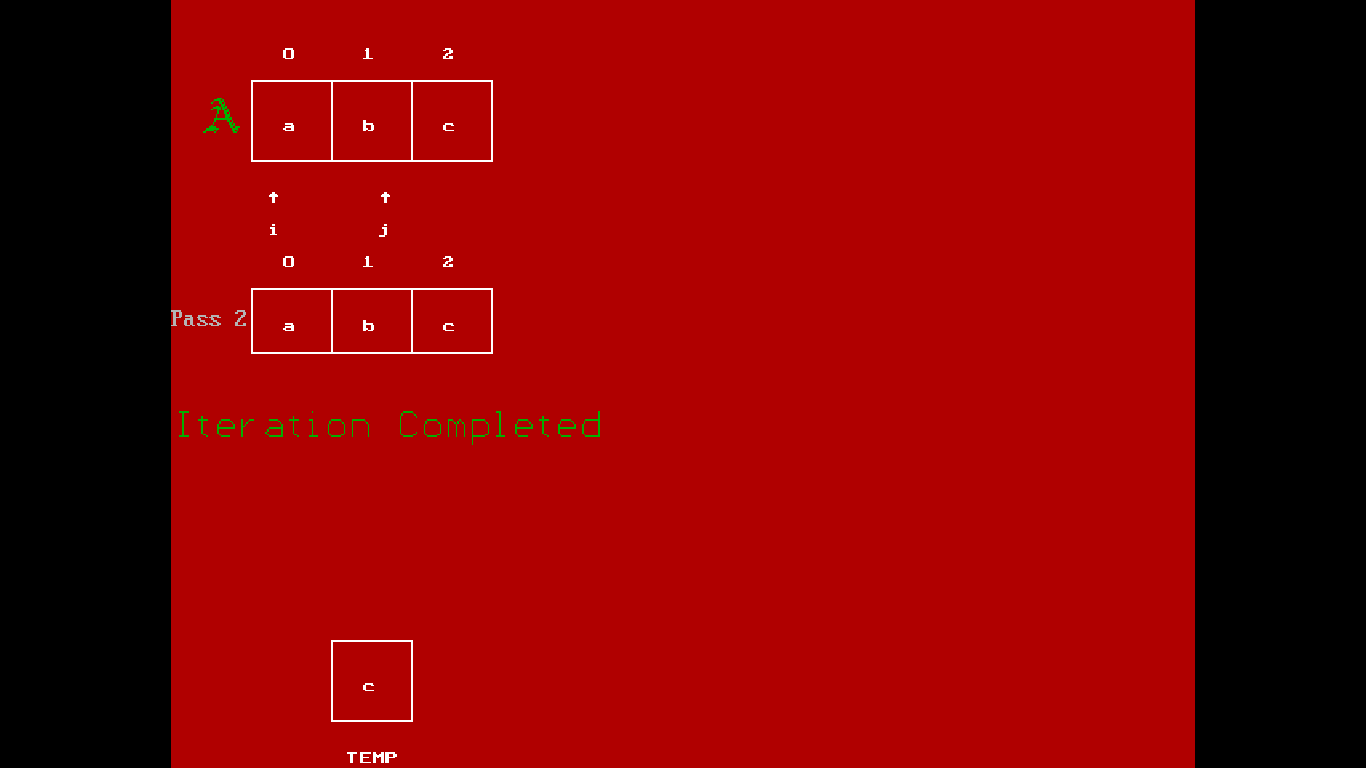


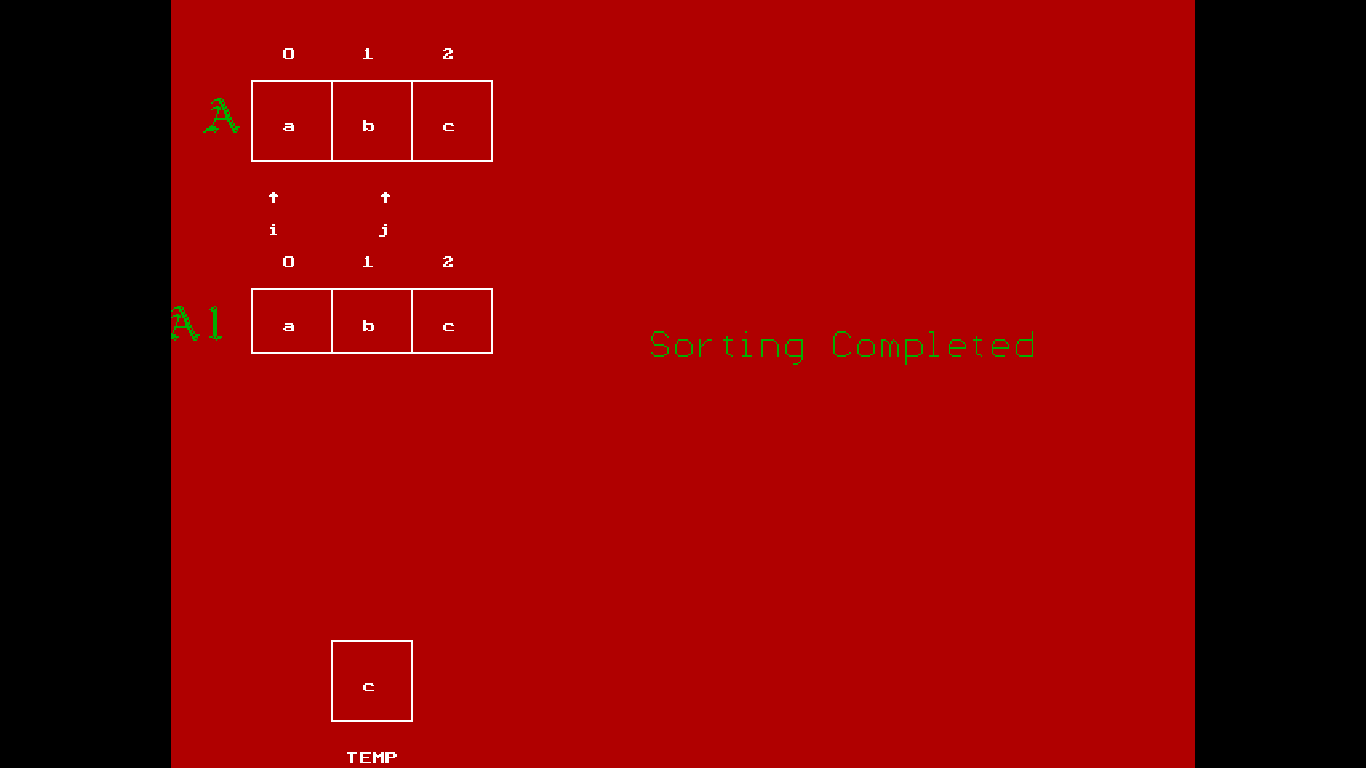




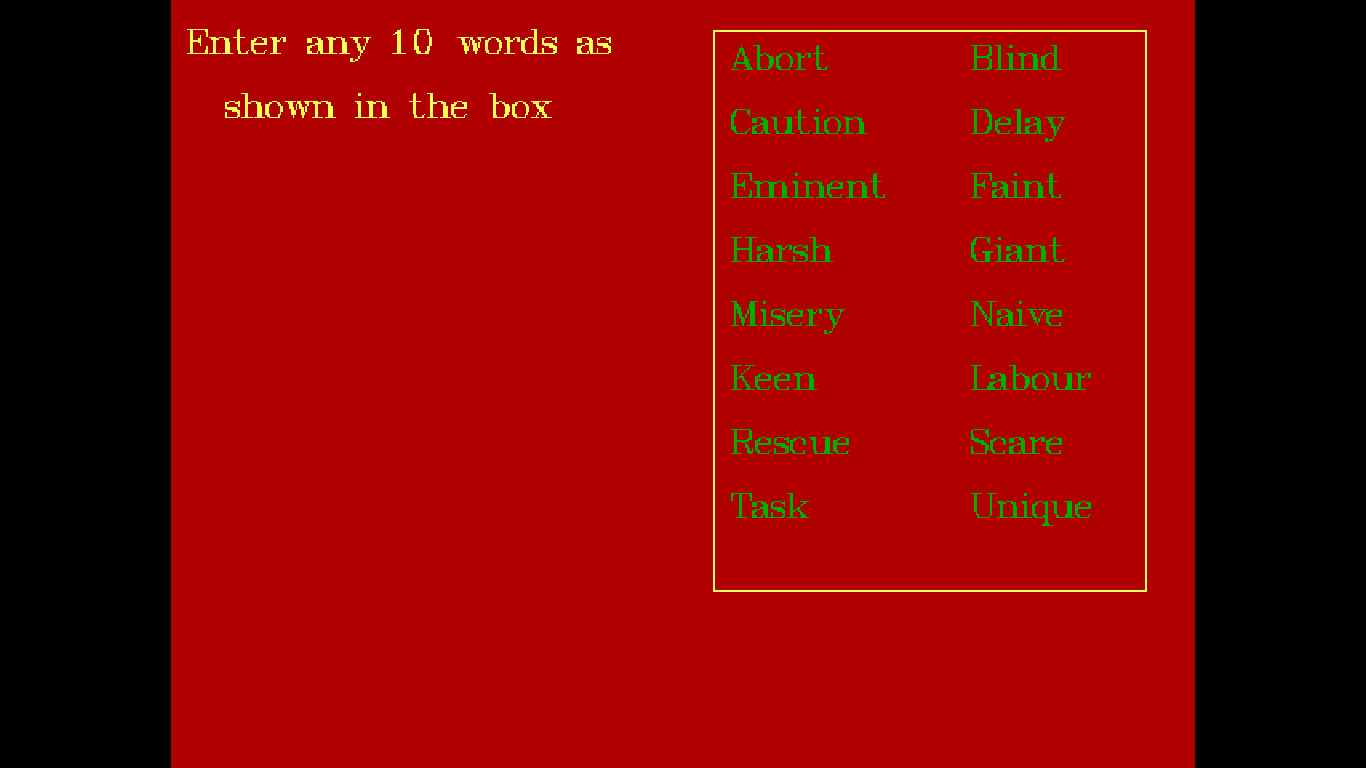


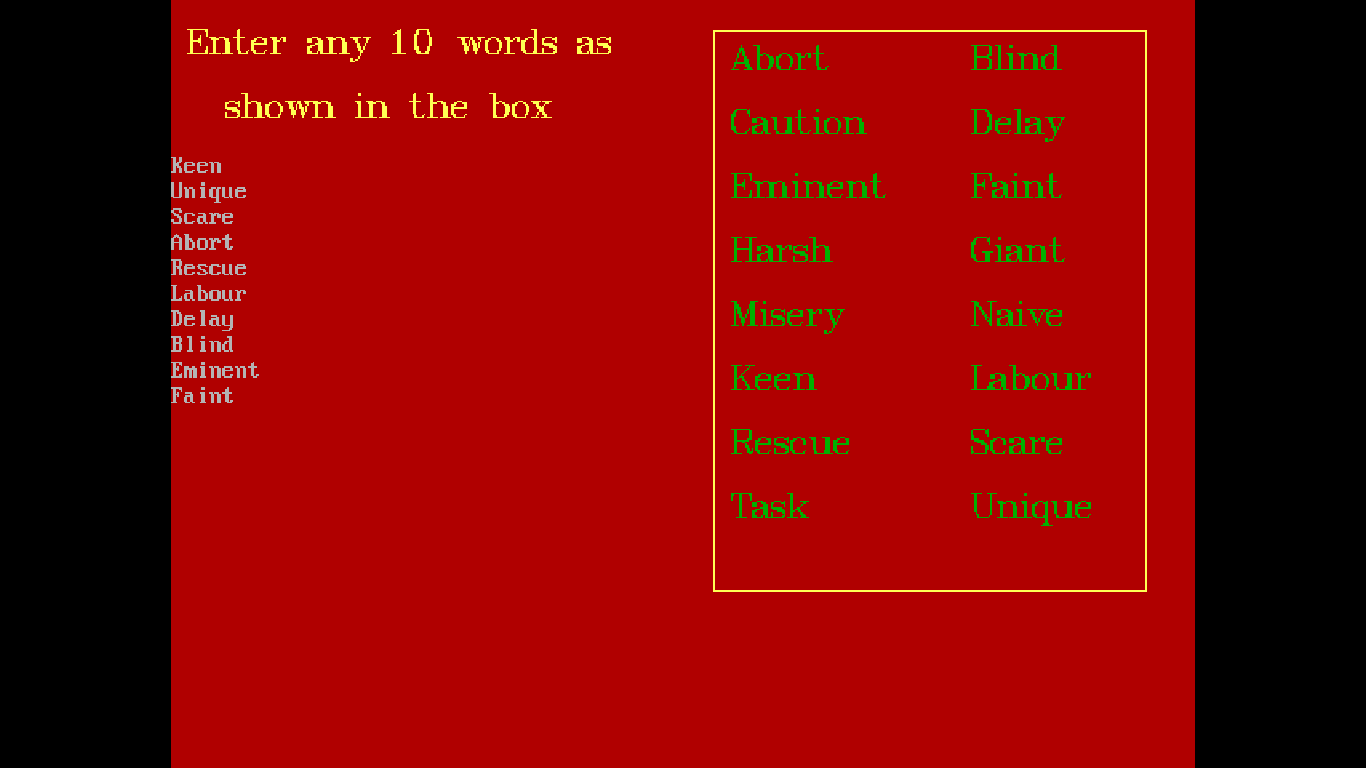


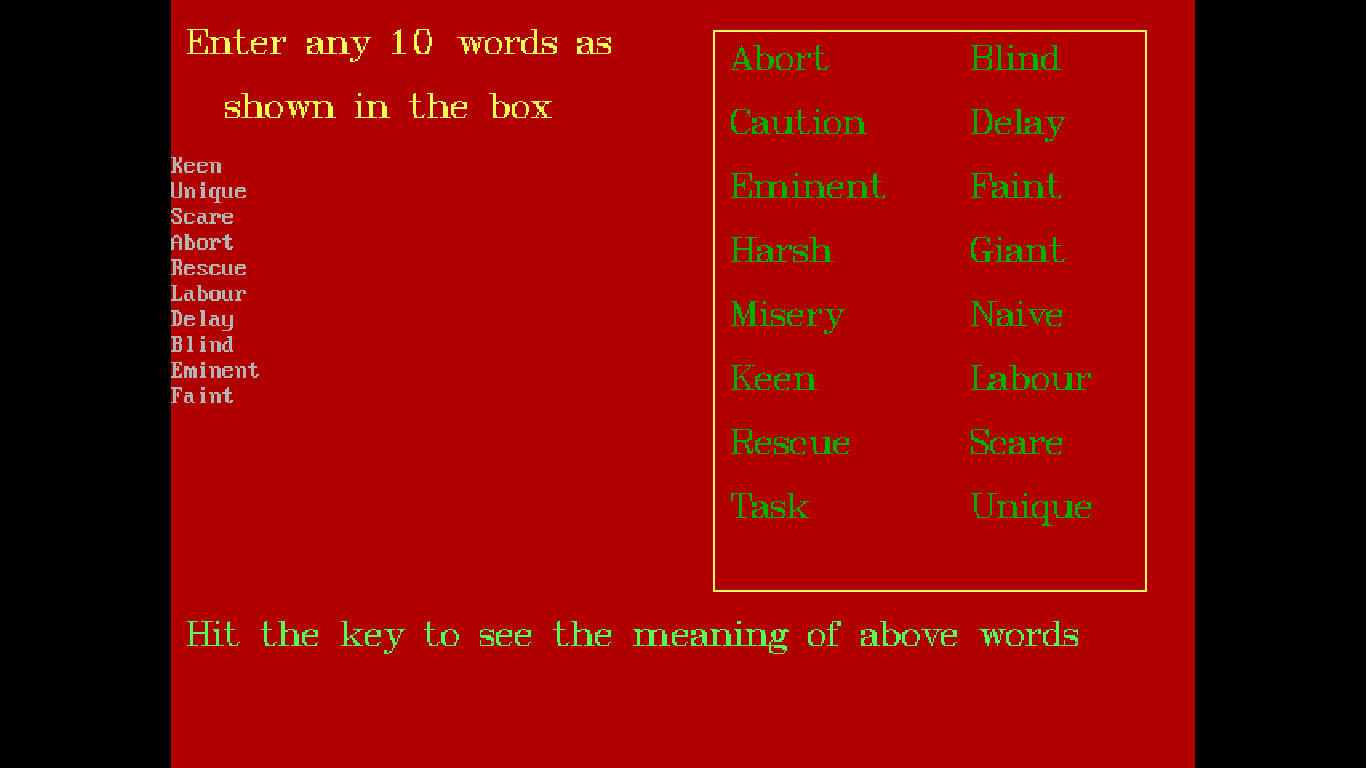


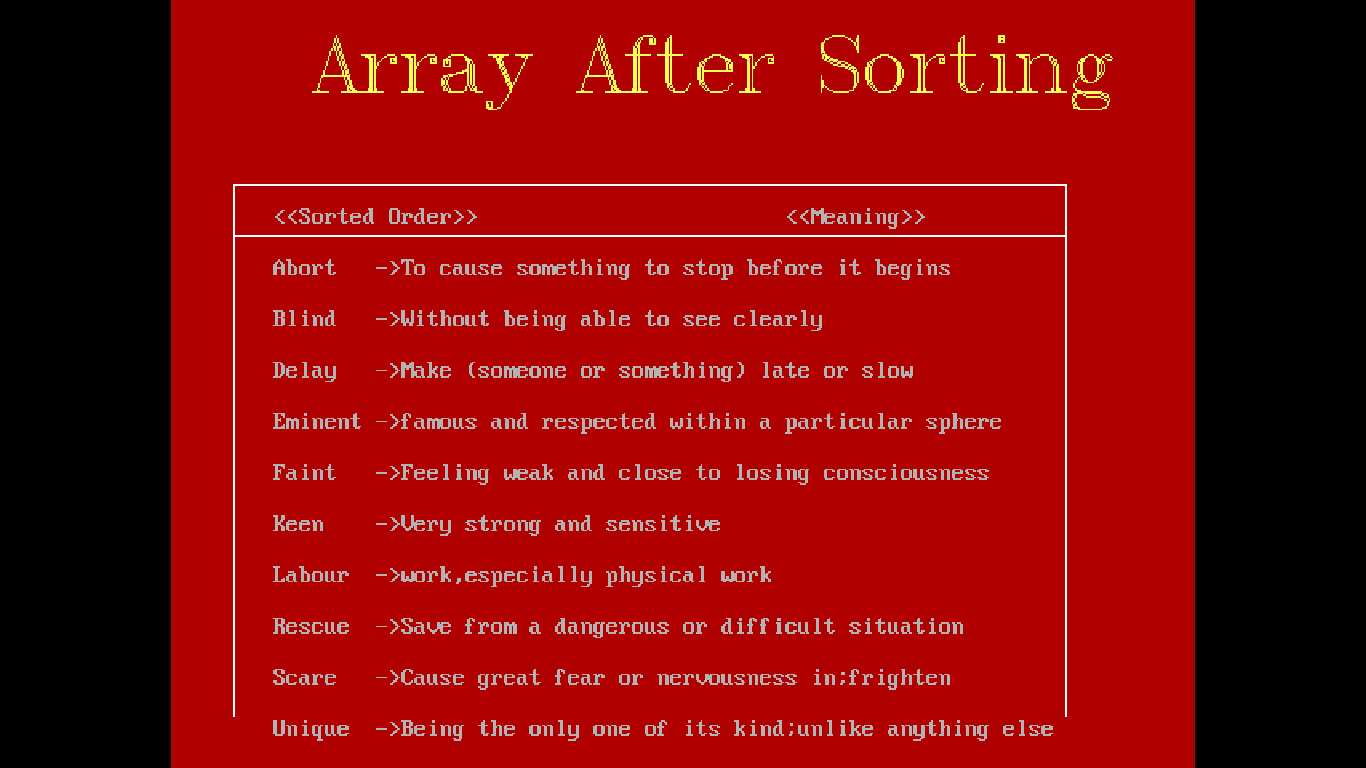
















**CONCLUSION**

In bubble sort, the idea of the algorithm is to move the higher valued elements generally towards the right and lower valued elements to toward left.

Bubble Sort takes several passes to sort elements in an array. Every pass need to do comparisons between elements and exchange the data if the elements are not in the right order. However the complexity of Bubble sort is the same for best case and worse case.

**REFERENCES**

* [*Introduction to Algorithms*](https://en.wikipedia.org/wiki/Introduction_to_Algorithms), Second Edition. MIT Press and McGraw-Hill, 2001.[www.Mathwork.com](http://www.Mathwork.com)
* [Sorting in the Presence of Branch Prediction and Caches](https://www.cs.tcd.ie/publications/tech-reports/reports.05/TCD-CS-2005-57.pdf)
* [www.google.com](http://www.google.com)
* https://www.youtube.com