Grades Of Students

Objective

Taking marks of a student and to output his/her grade as per the given rules.

Flow Chart

Code (Grade.pas)

```
program grade_;
var marks:integer;var grade:char;
begin
write('Please Enter Marks : ‘);
readln(marks);
grade:='F';
if marks>=33 then grade:='E';
if marks>=45 then grade:='D';
if marks>=65 then grade:='C';
if marks>=75 then grade:='B';
if marks>=90 then grade:='A';
writeln ( 'The Grade is ',grade);
readln;
end.
```

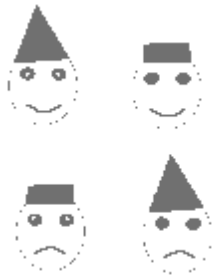
Output

```
Please Enter Marks : 33
The Grade is E

Please Enter Marks : 90
The Grade is A

Please Enter Marks : 57
The Grade is D
```

Swapping Of Variables

Objective

Taking values of two variables and to exchange their values without using a temporary variable.



Flow Chart

Code (Swap.pas)

```
program swap;
var a,b:integer;
begin
writeln('Enter A,B:');
readln(a,b);
writeln('Swapping..');
b:=b-a;
a:=a+b;
b:=a-b;
writeln('A=',a);
writeln('B=',b);
readln;
end.
```

Output

```
Enter A,B=10
30
Swapping..
A=30
B=10

Enter A,B=45
10
Swapping..
A=10
B=45
```

Reversing A Number

Objective

Taking a number from the user and to print it in the reverse order.

Flow Chart

Code (Reverse.pas)

```
program reverse;
var i,j,k:integer;
begin
k:=0;
Write('Enter the Number: ');readln(i);
while i>0 do begin
j:= i mod 10;
i:=i div 10;
k:= k*10 +j;
end;
writeln(k);
readln;
end.
```

Output

```
Enter the Number: 123
321

Enter the Number: 1002
2001

Enter the Number: 321
123
```

The Fibonacci Series…

Objective

Print N terms of the Fibonacci Series starting from 0 to Nth term; where N is a number provided by the user.

0
1 1
2 3 5
8 13 21 34
55 89 144 233

Flow Chart

Code (Fibonac.pas)

```
program fibonacci;
var i,j,k,m,l:integer;
begin
write('Enter Number Of terms: ');
readln(i);
j:=0;
k:=1;
write('0..');
for m:=2 to i do
begin
l:=j+k;
write(j+k,'..');
j:=k;
k:=l;
end;
readln;
end.
```

Output

```
Enter Number Of terms: 5
0..1..2..3..5..

Enter Number Of terms: 3
0..1..2..

Enter Number Of terms: 2
0..1..
```

The ASCII Table.

Objective

To draw the entire ASCII Character Set from 0 to 255th character.

Flow Chart

Code (Ascii.pas)

```
Program aciiitable;
var a:integer;
begin
for a:=0 to 255 do writeln( 'Ascii ',a,'=',chr(a));
readln;
end.
```

Output…

```
Ascii 236=_
Ascii 237=_
Ascii 238=_
Ascii 239=_
Ascii 240=_
Ascii 241=±
Ascii 242=_
Ascii 243=_
Ascii 244=_
Ascii 245=_
Ascii 246=÷
Ascii 247=_
Ascii 248=°
Ascii 249=•
Ascii 250=·
Ascii 251=_
Ascii 252=n
Ascii 253=²
Ascii 254=_
Ascii 255=_
```

The Maximum & the Minimum.

Objective

To find the Maximum and the minimum elements from a user defined array.

Flow Chart

Code (Maxmin.pas)

```
Program Max_Min_from_list;
var a:array[1..10] of integer;
var i,j,k,l:integer;
begin
write('Enter number of elements: ');readln(i);
writeln('Now enter elements: ');
for k:=1 to i do readln(a[k]);
j:=a[1];k:=a[1];
for k:=1 to i do
if a[k]<j then j:=a[k];
if a[k]>l then l:=a[k];
writeln('Max in List= ' ,l);
writeln('Min in List= ' ,J);
readln;
end.
```

Output…

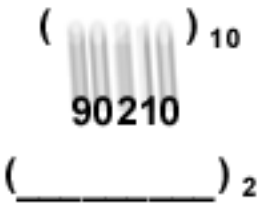
```
Enter number of elements: 8
Now enter elements:
2
3
4
5
6
7
8
9
Max in List= 9
Min in List= 2
```

Base To Base Conversion-I

Objective

To change the base of given decimal base number to a user defined base.

Flow Chart



Code (D2abase.PAS)

```
program decil2Abase;
var i,j,k,l,m,n:integer;
begin
k:=0;l:=0;m:=1;j:=1;
write('Enter the decimal Number');
readln(i);
while j<=1 do
begin
Write('Enter Target Base');
readln(j);
if j<=1 then writeln('Invalid Base Retry..');
end;
while i>0 do
begin
k:=i mod j;
l:=l+k*m;
i:=i div j;
m:=m*10;
end;
writeln(l);
readln;
end.
```

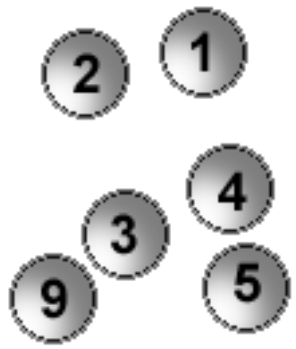
Output…

```
Enter the decimal Number 29
Enter Target Base 2
11101

Enter the decimal Number 2
Enter Target Base 2
10
```

Bubble Sort.

Objective

To sort a user-defined array using the Bubble Sort Algorithm.



Flow Chart

Code (Bubbles.pas)

```
program bubblesort;
procedure swap(var a,b:integer);
var temp:integer;
begin
temp:=a;
a:=b;
b:=temp;
end;
var a:array[1..10] of integer;
var i,j,k:integer;
begin
Write('Enter Number Of Elements:- ');
Readln(i);
writeln('Now enter elements:-')
for k:=1 to i do readln(a[k]);
for k:=i downto 2 do
for j:= 1 to (k-1) do
if a[j]>a[j+1] then swap(a[j],a[j+1]);
writeln;
Writeln('Sorted List:-');
for k:=1 to i do writeln(a[k]);
readln;
end.
```

Selection Sort.

Objective

To sort a user-defined array using the Selection Sort Algorithm.

Flow Chart

Code (Selesort.pas)

```
program selectoionsort;
var a:array[1..5] of integer;
var i,j,k,l,minp,min:integer;
begin
writeln('Enter No of Elements; ');
readln(i);
writeln('Now Enter Elements:-');
for k:=1 to i do readln(a[k]);
for k:=1 to i do
begin
min:=a[k];
minp:=k;
for l:=k+1 to (i-1) do
if a[l]<min then
begin
min:=a[l];
minp:=l;
end;
j:=a[minp];
a[minp]:=a[k];
a[k]:=j;
end;
writeln;
writeln('Sorted List:');
for k:=1 to i do writeln (a[k]);
readln;
end.
```

Junk Sort!!

Objective

To sort a user-defined array using my own sort algorithm as defined below.

Flow Chart

Code (Sort.pas)

```
Program Junk_Sort;
var a,b,c:array[1..10] of integer;
var j,k :integer;
var done:boolean;
begin
done:=true;
for  j:= 1 to 10 do
readln (c[j]);
for  j:= 1 to 10 do
      a[j]:=c[j];
 while(done) do
 begin
      for  j:= 1 to 10 do
       a[j]:=a[j]-1;
      for  j:= 1 to 10 do
       if a[j]=0 then
        begin
        k:=k+1;
        b[k]:=c[j];
        if (k>=10) then
        done:=false;
        end;
     end;
writeln;
for  j:= 1 to 10 do
writeln (b[j]);
readln;
end.
```

Linear Search.

Objective

To search a user provided element in a user defined array using Linear Search Algorithm.

Flow Chart

Code (Linears.pas)

```
Program linearserch;
var I:array[1..10] of integer;
var j,k,l:integer;
begin
write('Enter Number of elements: ');
readln(j);
write('Now Enter Elements:-');
for k:=1 to j do readln(i[k]);
writeln;
write('Enter Target Element');readln(l);
k:=1;
while k<=j do
if i[k]=l then k:=j+k+1 else k:=k+1;
if k>j+1 then writeln('Element Found at ', k-j-1) else
writeln('Element not Found');
readln;
end.
```

Output

```
Enter Number of elements: 3
Now Enter Elements:-4
5
6

Enter Target Element:-6
Element Found at 3
```

Binary Search.

Objective

To search a user provided element in a user defined array using Binary Search Algorithm.

Flow Chart

Code (Binarys.pas)

```
Program binaryserch;
var a:array[1..10] of integer;
var I,j,k,ul,ll:integer;
begin
write('Enter number of elements: ');readln(i);
writeln('Now enter elements: ');
for k:=1 to i do readln(a[k]);
Write('Search For: ');readln(j);
ul:=1;ll:=i;
while ul<>ll do
if a[((ul+ll) div 2)] < j then  ul:=(ul+ll) div 2+1
else ll:=(ul+ll) div 2;
if a[ll]= j then writeln('Search Element Found') else
writeln('Search Element Does Not Exist');
readln;
end.
```

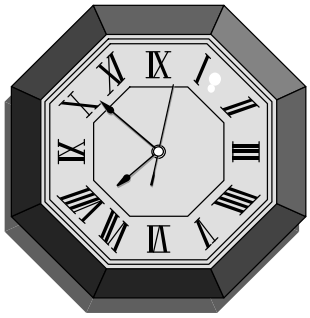
Output

```
Enter number of elements: 4
Now enter elements:
6
7
8
9
Search For: 4
Search Element Does Not Exist
```

Stop Watch.

Objective

To create a stopwatch in Pascal.



Flow Chart

Code (Stopwatc.pas)

```
Program stopwatch;
Var h,m,s,ms,k:integer;
Begin
h:=0;m:=0;s:=0;ms:=0;
while true do begin
ms:=ms+1;
if ms>999 then begin
ms:=0; s:=s+1;
if s>59 then begin
s:=0;m:=m+1;
if m>59 then begin
m:=0;h:=h+1;
end;
end;
end;
writeln (h,':',m,':',s,':',ms);
for k:=1 to 25000 do;
end;
end.
```

Output

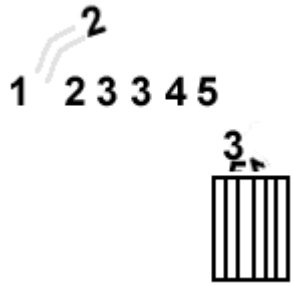
```
0:0:1:155
0:0:1:156
0:0:1:157
0:0:1:158
0:0:1:159
0:0:1:160
0:0:1:161
0:0:1:162
^C
```

Elimination of Duplicates-I.

Objective

To eliminate duplicate elements in a user defined array-using algorithm defined below.

Flow Chart



Code (Dup.pas)

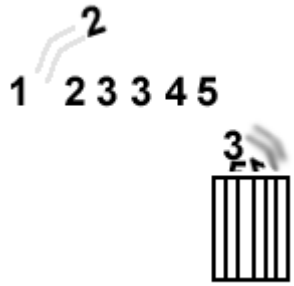
```
Program dup1;
var a,b : array[ 1..10] of integer;
var I,j,k:integer;
var dup:boolean;
begin
writeln('Enter Elements: ');
for I:=1 to 10 do readln (a[i]);

for I:=1 to 10 do begin
dup:=false;
    for j:=1 to k do
       if a[i]=b[j] then dup:=true;
    if not dup then begin
    k:=k+1;
    b[k]:=a[i];
    end;
end;
writeln('Unduplicated List');for i:=1 to k do writeln
(b[i]);
readln;
end.
```

Elimination of Duplicates-II.

Objective

To eliminate duplicate elements in a user defined array using the second algorithm defined below.

Flow Chart



Code (Dup2.pas)

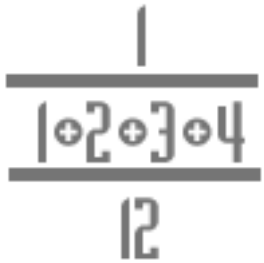
```
Program remmove_duplicates;
var a:array[1..10] of integer;
var I,j,k,l:integer;
begin
l:=10;
writeln('Enter Elements');
for I:=1 to 10 do readln (a[i]);
for I:=1 to 10 do
        for j:=(i+1) to l do
           if a[i]=a[j] then
              begin
              for k:= j to l do a[k]:=a[k+1];
              l:=l-1;
              end;

writeln;
writeln('Un-Duplicated Elements');
for I:=1 to l do writeln (a[i]);
readln;
end.
```

Hcf & Lcm Of Two Numbers.

Objective

To find the LCM & the HCF of the user provided two numbers.

Flow Chart



Code (Hcf.pas)

```
Program lcm_;
var a,b,c,hcf,lcm:integer;
begin
writeln('enter a,b');
readln (a,b);
c:=a;
hcf:=1;
while (c>=1) do
begin
     if a mod c=0 then
       if b mod c=0 then
          begin
               hcf:=c;
               c:=1;
          end;
c:=c-1;
end;
writeln (hcf);
while (c>=0) do
begin
     c:=c+1;
     if (a*c) mod b=0 then
          begin
          lcm:=a*c;
          c:=-1;
          end;
end;
writeln( lcm);
readln;
end.
```

Nearest Prime Number ?

Objective

To find the nearest Prime Number of the user provided number.

Flow Chart

Code (Ner.pas)

```
program ner;
var I,a,pri,c:integer;
function isprime( num:integer): boolean;
begin
    c:=1;
    while c<(num div 2) do  begin
        c:=c+1;
        if num mod c =0 then c:=num;
    end;
    if c=num then  isprime:=false
    else  isprime:=true;
end;
begin
readln(i); a:=0; pri:=0;
  while pri=0 do  begin
      a:=a+1;
           if isprime(i+a) then  pri:=i+a
           else
          if isprime(i-a) then   pri:=i-a;
  end;
writeln('nearest prime is',pri); readln;
end.
```

Output

```
3
nearest prime is 2

15
nearest prime is 17
```

Sum Even, Odd & Prime?

Objective

To find the sum of all the Even, Odd & Prime Numbers up to a user provided number.

Flow Chart

Code (Suoep.pas)

```
program sum_odd_even_prime;
var i,j,k,l,m:integer;
function isprime( num:integer): boolean;
var c:integer;
begin
   if (num=1) or  (num=2) then begin
   isprime:=true;exit;end;
   c:=1;while c<(num div 2) do begin
   c:=c+1;if num mod c =0 then c:=num;
   end;
   if c=num then isprime:=false else isprime:=true;
end;
begin
j:=0;l:=0;m:=0;
write ('Sum all Even,Prime & Odd till number : ');
readln(i);
for k:=1 to i do
begin
if isprime(k) then j:=j+k;
if k mod 2 =0 then l:=l+k else m:=m+k;
end;
write ('Sum Prime=');writeln(j);
write ('Sum Odd  =');writeln(m);
write ('Sum Even =');writeln(l);
readln;
end.
```

Drawing A Pattern.

Objective

To draw the pattern as illustrated on the right.

Flow Chart

Code (Pat.pas)

```
var a,b,c:integer;
begin
a:=1;
c:=1;
while (true) do
begin
    a:=a+c;
    if ((a>5) or (a<1)) then
    c:=c* -1;
    for b:= 0 to (5-a) do
        write (' ');
    for b:= 0 to (2*a) do
       write ('*');
 for b:= 0 to 29000 do;
 for b:=0 to 29000 do;
 writeln;
end;
readln;
end.
```

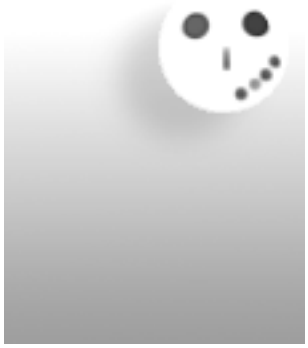
Output

```
**********
 ********
  ******
   *****
    ***
     *
    ***
   *****
  ******
 ********
**********
```

Drawing A Circle.

Objective

To draw a Circle of user given radius.



Flow Chart

Code (Cir.pas)

```
Program Circle;
var x,y,r,temp:real;
begin
write('Enter the radius of Circle : ');readln(r);
y:=0;writeln;
while y>=(-1*2*r) do begin
   x:=0;
    while x<= 2*r do
    begin
    temp:=  sqrt((r*r) - ((y+r)*(y+r)));
    if (x>temp+r) or (x<r-temp) then
    write(' ')
    else
     write ('+');
    x:=x+0.4;
    end;
 writeln;
 y:=y-1;
end;
readln;
end.
```

Output

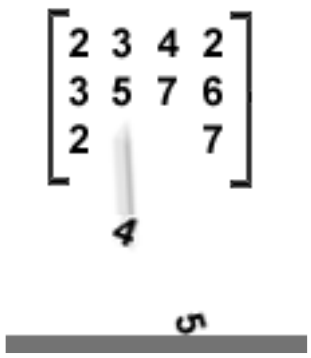
```
Enter the radius of Circle : 3


  ++++++++++++
 ++++++++++++++
++++++++++++++++
 ++++++++++++++
  ++++++++++++
```

Transpose Of A Matrix.

Objective

To calculate the transpose of a user defined matrix.

Flow Chart



Code (Transpos.pas)

```
var a,b:array[1..10,1..10] of integer;
var m,n:array[1..2] of integer;
var i,j:integer;
begin
writeln('enter order of a');
read(m[1],m[2]);
writeln('enter A');
for I:=1 to m[1] do
for j:=1 to m[2] do
read(a[i,j]);
for I:= 1 to m[2] do
for j:= 1 to m[1] do
b[i,j]:=a[j,i];
Writeln('Transposed Matrix');
for I:= 1 to m[2] do
begin
for j:= 1 to m[1] do
write( b[i,j],'|');
writeln;
end;
end.
```

Output

```
enter order of a
2
2
enter A
3
4
5
6
Transposed Matrix
3|5|
4|6|
```

Average Of A List.

Objective

To calculate the average of a list of given numbers using arrays.

Flow Chart

Code (Average.pas)

```
program average;
var list:array[1..20] of integer;
var length,temp1:integer;
var sum:real;
begin
write('Enter Number Of Elements:');
read(length);
writeln('Enter Elements:-');
for temp1:=1 to length do
begin
write('Enter Element ',temp1,':');
read(list[temp1]);
end;
for temp1:=1 to length do sum:=sum+list[temp1];
sum:=sum/length;
writeln('Average is ',sum:6:2);
readln;
readln;
end.
```

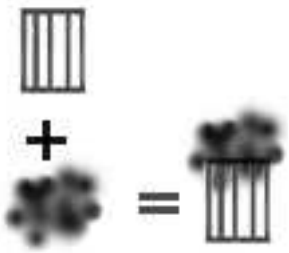
Output

```
Enter Number Of Elements:4
Enter Elements:-
Enter Element 1:1
Enter Element 2:2
Enter Element 3:3
Enter Element 4:4
Average is   2.50
```

Junk Screen.

Objective

To fill the console screen with random characters from the ASCII table.



Flow Chart

Code

```
program junkscreen;
var i,j,k:integer;
begin
for j:=1 to 2000 do
begin
k:=random(255);
write (chr(k));
end;
end.
```

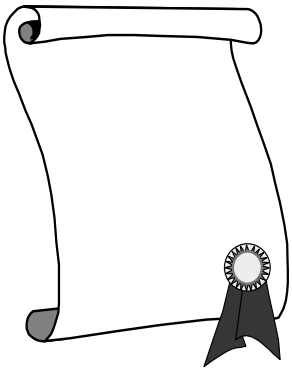
Output

```
_Váé_├ n− ¥Zéñ+ó_  {²<2o»£¦v*í‼ _+¦m  8ô
ßx_¦_+·6'¿-0┘ +@    Xß_o?k^Ñmó--
të¥N=h+º-_  R_wâ_½iö$°w'O£-{"v$
`µå_¦Lù← WZkÆ_¡+²[↑ fF_<b¶ååê91i_Çô^_+
_¡Æå_:│ •~_4-↑
E-+kÇH¦Uln°\ò¦{,Rj{Zv┼ ._ît_-ûjL+ôGñ_@ƒ
x-
¦8•¬_D_<zY¦_Bñ_¦↑ [_3_ò_+b−│Ü+wü+_G)
"²!V'j~`KF◀ K+=_òCd‼ -i¦
_¶Câ+¦rb+¶x#_Pû┬ q
_"1├ YƒoÆM_h┬ ¦_7-l¦·+← ù£_¢+Q
ZK_i7=    _+└ .àHáïÅv+-
¦>)ß_┤ B__-jmn+_i¶É┐ _öd__ôÑ
&äIÖüOS¦§«¶
c_zi-┤ ¡¦G!_+┬ >_c+¦_ G¦<f¿•+_îU¼Ä]Re↑ J
#+−  _  nI_¤┼ =M-
d++°\¦3_\_F^_èDkF+_ê+┼ 3y+0--+«¼_FT
+à+f− U¦²É^K_â·ycS-← − ¦UñX;$#¦_ìS_Mín◀
ö+d+WÿFm
```

Ranking Of Students...

Objective

To generate a result report using user provided marks of each student and to find their ranks.



Code (Students.pas)

```
program students_avarage;
type student=record
name:string[10];
mphy:integer;
mmat:integer;
mche:integer;
mave:real;
end;
{Global Declarations}
var a:array[1..10] of student;
var i,j,k,l,prea,prek:integer;
function concate2(x:string;y:integer):string;
var i:integer;begin
while length(x)<y do x:=x+' ';
concate2:=x;
end;
function concate(x,y:integer):string;
var te:string;var temp:integer; begin
str(x,te);
while length(te)<y do te:=te+' '; concate:=te;
end;
procedure sort;
var minp,j,k:integer;var min,temp:real;
var l:student;
begin
l.name:='';l.mmat:=0;l.mave:=0;l.mche:=0;l.mphy:
=0;
for K:=1 to (i-1) do begin
minp:=k;min:=a[k].mave;
        for j:= (k+1) to i do
        if a[j].mave>min then begin
        minp:=j;min:=a[j].mave;
        end;
l:=a[k];
a[k]:=a[minp];
a[minp]:=l;
end;
end;
{Main Module}
begin
Write('Enter Number of Students');readln(i);
Writeln;
for k:=1 to i do with a[k] do
begin                                    {Cont…}
```

Output

```
Enter Number of Students3

1.Enter Student name: vaibhav
 Enter vaibhav`s marks in Physics  : 20
 Enter vaibhav`s marks in Maths    : 20
 Enter vaibhav`s marks in Chemistry: 20

2.Enter Student name: ronak
 Enter ronak`s marks in Physics  : 30
 Enter ronak`s marks in Maths    : 30
 Enter ronak`s marks in Chemistry: 30

3.Enter Student name: ammar
 Enter ammar`s marks in Physics  : 40
 Enter ammar`s marks in Maths    : 40
 Enter ammar`s marks in Chemistry: 40


Report:
Rank  Name  Physics  Maths  Chemistry  Average
-----------------------------------------------------------
1     ammar   40      40      40         40.00
2     ronak   30      30      30         30.00
3     vaibhav 20      20      20         20.00
```

Ranking Of Students .

Code(Students.pas)

```
write(k,'.Enter Student name: ');read(name);
write(' ','Enter ',name,'`s marks in Physics  : ');readln(mphy);
write(' ','Enter ',name,'`s marks in Maths    : ');readln(mmat);
write(' ','Enter ',name,'`s marks in Chemistry: ');readln(mche);
mave:=(mphy+mmat+mche) / 3;
writeln;
end;
writeln;
writeln('Report:');
 sort;
write(concate2('Rank',10));
write(concate2('Name',10));
write(concate2('Physics',10));
write(concate2('Maths',10));
write(concate2('Chemistry',10));
writeln(concate2('Average',10));
for k:=1 to 78 do write('-');
writeln;
prea:=1;
prek:=1;
for k:=1 to i do with a[k] do
with a[k] do
begin
if a[k].mave<>a[prea].mave then begin
prek:=prek+1;prea:=k end;
write(concate(preK,10));
write(concate2(a[k].name,10));
write(concate(a[k].mphy,10));
write(concate(a[k].mmat,10));
write(concate(a[k].mche,10));
writeln(mave:3:2);
end;
readln;
end.
{Code ConCludes}
```

Array Merging & Partitioning.

Objective

To Partition or Merge one and two user-defined arrays as per the users wish.

Code(Partitio.pas)

```
Program Parition;
var a,b,c:array[1..10] of integer;
var i,j,k,l:integer;
begin
writeln('1.Merge Arrays');writeln('2.Partition Arrays');
writeln('Select Option');readln(l);
if l=2 then begin
write('Enter No. of Elements in A: ');readln(i);
writeln('Enter Elements:');
for k:=1 to i do readln(a[k]);
writeln('Partion From Element: '); readln(j);
for k:=1 to i do
if k<j then b[k]:=a[k] else c[k-j+1]:=a[k];
writeln('Partitioned Array');
Writeln('B    C');
Writeln('------');
for k:=1 to i do
begin
if k<j then write(b[k]) else write(' ');
Write('   ');
if k<=i-j+1 then write(c[k]) else write(' ');
writeln;
end;
readln;
end
else
begin
write('Enter No. of Elements in A:-');readln(i);
writeln('Enter Elements:');
for k:=1 to i do readln(a[k]);
write('Enter No. of Elements in B:-');readln(j);
writeln('Enter Elements:');
for k:=1 to j do readln(b[k]);
for k:=1 to (i+j) do
if k>i then c[k]:=b[k-i] else c[k]:=a[k];
writeln('Merged List');
for k:=1 to i+j do writeln(c[k]);
readln;
end
end.
```

Output

```
1.Merge Arrays
2.Partition Arrays
Select Option
1
Enter No. of Elements in A:-1
Enter Elements:
9
Enter No. of Elements in B:-2
Enter Elements:
11
12
Merged List
9
11
12

1.Merge Arrays
2.Partition Arrays
Select Option
2
Enter No. of Elements in A: 5
Enter Elements:
1
2
3
4
5
Partion From Element:
3
Partitioned Array
B    C
------
1   3
2   4
    5
```

Armstroms, Palindromes & Primes...

Objective

Checking a number for being a Palindrome, Armstrom or Prime.

Code (Pali.pas)

```
Program APP;
{Global Declarations}
var a,b:integer;
function isarmstrong(i:integer):boolean;
var temp:array[1..3] of integer;
begin
temp[1]:=i;
temp[3]:=0;
while i>0 do
begin
temp[2]:=i mod 10;
i:=(i-temp[2]) div 10;
temp[3]:=temp[3]*10+temp[2];
end;
if temp[3]=temp[1] then isarmstrong:=true else
isarmstrong:=false;
end;
function ispalidrom(i:integer):boolean;
var temp:array[1..3] of integer;
begin
temp[1]:=i;
temp[3]:=0;
while i>0 do
begin
temp[2]:=i mod 10;
i:=(i-temp[2]) div 10;
temp[3]:=temp[3]+temp[2]*temp[2]*temp[2];
end;
if temp[3]=temp[1] then ispalidrom:=true else
ispalidrom:=false;
end;
function isprime(i:integer):boolean;
var temp:array[1..3] of integer;
begin
isprime:=true;
if i<=0 then isprime :=false;
if i>=4 then
     for temp[1]:= 2 to i div 2 do
        if i mod temp[1]=0 then isprime:=false;
end;
function showmenu:integer;
var temp:integer;
begin
writeln;                                    {Cont.....}
```

Output

```
                    Welcome To Pali

Enter Your Choice
1.Check for Prime
2.Check for Armstrong
3.Check for Palidrome
4.Quit
1
Enter Number To Check For Prime:3
3 is a Prime


Enter Your Choice
1.Check for Prime
2.Check for Armstrong
3.Check for Palidrome
4.Quit
3
Enter Number To Check For Palidrome:153
153 is a Palidrome


Enter Your Choice
1.Check for Prime
2.Check for Armstrong
3.Check for Palidrome
4.Quit
2
Enter Number To Check For Armstrong:151
151 is a Armstrong
```

Armstroms, Palindromes & Primes.

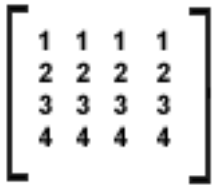
Code (Pali.pas)

```
writeln;
writeln('Enter Your Choice');
writeln('1.Check for Prime');
writeln('2.Check for Armstrong');
writeln('3.Check for Palidrome');
writeln('4.Quit');
read(temp);
if (temp>4) or (temp<1) then
begin
write('Invalid Input');temp:=showmenu;
end;
showmenu:=temp;
end;
{Main Module}
begin
Writeln('                Welcome To Pali              ');
repeat
    b:=showmenu;
    case b of
        3:
          begin
          write('Enter Number To Check For Palidrome:');
          readln(a);
          if ispalidrom(a) then writeln(a,' is a Palidrome')
          else writeln(a,' is not a Palidrome') ;
          end;
        2:
          begin
          write('Enter Number To Check For Armstrong:');
          readln(a);
          if isarmstrong(a) then writeln(a,' is an Armstrong')
          else writeln(a,' is not an Armstrong') ;
          end;
        1:
          begin
          write('Enter Number To Check For Prime:');
          readln(a);
          if isprime(a) then writeln(a,' is a Prime')
          else writeln(a,' is not a Prime') ;
          end;
    end;
until b=4;
writeln ('Bye-Bye');
end.
{Code Concludes}
```

Matrix Operations ...

Objective

To perform Matrix Addition, Subtraction or Multiplication as per the users wish.



Code (Matopr.pas)

```
program matrix_opretions;
type matrix=record
matrix:array[1..3,1..3] of integer;
rows:integer;
columns:integer;
end;
var res:integer;var m1,m2,m3:matrix;
function add(a,b:matrix;var result:matrix):boolean;
var temp:array[1..3] of integer;
begin
add:=false;
if not (a.rows=b.rows) then exit;
for temp[1]:=1 to a.rows do
for temp[2]:=1 to a.columns do
result.matrix[temp[1],temp[2]]:=a.matrix[temp[1],t
emp[2]]+b.matrix[temp[1],temp[2]];
add:=true;
result.rows:=a.rows;
result.columns:=a.columns;
end;
function subtract(a,b:matrix;var
result:matrix):boolean;
var temp:array[1..3] of integer;
begin
for temp[1]:=1 to b.rows do
for temp[2]:=1 to b.columns do
b.matrix[temp[1],temp[2]]:=-1*
b.matrix[temp[1],temp[2]];
subtract:=add(a,b,result)
end;
procedure get(var a:matrix;message:string);
var temp:array[1..4] of integer;
begin
if message='' then message:='Enter The Matrix';
Writeln(message);
Writeln('Enter Rows And Cols');
readln(temp[1],temp[2]);
Writeln('Now Enter Elements ');
for temp[3]:=1 to temp[1] do
for temp[4]:=1 to temp[2] do
readln( a.matrix[temp[3],temp[4]]);
a.rows:=temp[1];a.columns:=temp[2];
end;                                    {Contd.....}
```

Output

```
Welcome To Matix Calculator


Enter Your Choice
-----------------
1.Add
2.Subtract
3.Multiply
4.Quit
-----------------
1
Enter Matrix A
Enter Rows And Cols
1
1
Now Enter Elements
3
Enter Matrix B
Enter Rows And Cols
1
1
Now Enter Elements
4
Result A+B=
7|
```

Matrix Operations…..

Code (Matopr.pas)

```
procedure show(a:matrix;message:string);
var temp:array[1..2] of integer;
begin
if message='' then message:='The Matrix is';
Writeln(message);
for temp[1]:=1 to a.rows do
begin
for temp[2]:=1 to a.columns do
write( a.matrix[temp[1],temp[2]],'|');
writeln;
end;
end;
function product(a,b:matrix;var result:matrix):boolean;
var i,j,k:integer;
begin
product:=false;
if a.columns<>b.rows then exit;
for i:= 1 to a.rows do
for j:= 1 to a.columns do
for k:= 1 to b.columns do
result.matrix[i,k]:=result.matrix[i,k]+(a.matrix[i,j]*b.matrix[i,k]);
product:=true;
result.rows:=a.rows;
result.columns:=b.columns;
end;
function showmenu:integer;
var temp:integer;
begin
writeln('Enter Your Choice');
writeln('-----------------');
writeln('1.Add');
writeln('2.Subtract');
writeln('3.Multiply');
writeln('4.Quit');
writeln('-----------------');
Readln (temp);
if (temp<1) or (temp>4) then
begin
writeln('Invalid Choice');
temp:=showmenu;
end;
showmenu:=temp;
end;
{Cont.....}
```

Matrix Operations.

Code (Matopr.pas)

```
begin
writeln('                Welcome To Matix Calculator              ');
writeln;
writeln;
res:=showmenu;
while res<>4 do
begin
case(res) of
1:       begin
            get(m1,'Enter Matrix A');
            get(m2,'Enter Matrix B');
            if add(m1,m2,m3) then
            show(m3,'Result A+B=') else writeln('Error Processing A+B');
         end;
2:       begin
            get(m1,'Enter Matrix A');
            get(m2,'Enter Matrix B');
            if subtract(m1,m2,m3) then
            show(m3,'Result A-B=') else writeln('Error Processing A-B');
         end;
3:       begin
            get(m1,'Enter Matrix A');
            get(m2,'Enter Matrix B');
            if product(m1,m2,m3) then
            show(m3,'Product AxB=') else writeln('Error Processing AxB');;
         end;
end;
res:=showmenu;
end;
writeln('Bye-Bye');
end.
{Code Concludes}
```

Recursion.... Recursion....Recursion...

Objective

Recursion is a feature available in most of the programming. It is feature where a Function or a Method can call itself with optional parametres. Recursion is made possible with the help of stacks. A stack is a vertical placement of copies of variables of each an every function in the stack. A stack can be LILO(Last in Last Out) or LIFO(Last in First Out). Pascal works on stacks using LIFO method. Here we are using Recursion to evaluate:

1. **Power of a number.**
2. **To evaluate value of base to a power.**
3. **Sin,Cos,Tan of any Degree.**

Code (Power.pas)

```
program power;
var i,j:integer;
function pow(base,power:integer):integer;
begin
if power=0 then pow:=1 else pow:=base*pow(base,power-1);
end;
begin
write('Enter Base:  ');readln(i);
write('Enter Power: ');readln(j);
writeln(i,' to the power ',j,' = ',pow(i,j));
readln;
end.
```

Output(Power.exe)

```
Enter Base:  3
Enter Power:  3
3 to the power 3 = 27
```

Code (Factoria.pas)

```
program factorial_;
var i,j,k:integer;
function factorial(x:integer):integer;
begin
if x=1 then factorial:=1 else factorial:=x*factorial(x-1);
end;
begin
write('Enter the Number: ');readln(i);
writeln('Factorial of ',i,' is= ',factorial(i));
readln;
end.
```

Output(Factoria.exe)

```
Enter the Number: 5
Factorial of 5 is= 120
```

Recursion.... Recursion....Recursion.

Objective

Recursion is a feature available in most of the programming. It is feature where a Function or a Method can call itself with optional parametres. Recursion is made possible with the help of stacks. A stack is a vertical placement of copies of variables of each an every function in the stack. A stack can be LILO(Last in Last Out) or LIFO(Last in First Out). Pascal works on stacks using LIFO method. Here we are using Recursion to evaluate:

1. **Power of a number.**
2. **Value of base to a power.**
3. **Sin,Cos,Tan of any angle in Degree.**

Code (Sincosta.pas)

```
program sin_cos_tan;
var power,sign:integer;
var i,j,k,l:real;
function pow(base,power:real):real;
begin
if power=0 then pow:=1 else pow:=base*pow(base,power-1); end;
function factorial(x:integer):real;
begin
if x=0 then factorial:=1 else factorial:=x*factorial(x-1); end;
function sine(x:real):real;
var l:real;
begin
power:=power+2;sign:=-1*sign;
l:= pow(x,power)/factorial(power)*sign;
if power>=29 then sine:=l else
sine:=sine(x)+l;
end;
begin
write('Enter The Degrees To Evaluate : ');readln(i);
j:= pi/180*i;
power:= -1;sign:=-1;
k:=sine(j);
writeln('Sin of ',i:4:5,'= ',k:1:10);
power:= -2;sign:=-1;
l:=sine(j);
writeln('Cos of ',i:4:5,'= ',l:1:10);
writeln('Tan of ',i:4:5,'= ',(k/l):5:10);
readln;
end.
```

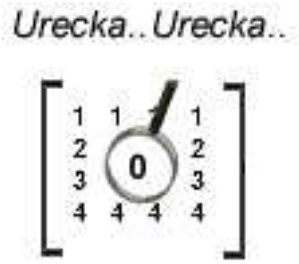
Output(Sincosta.exe)

```
Enter The Degrees To Evaluate :
45
Sin of 45.00000= 0.7071067812
Cos of 45.00000= 0.7071067812
Tan of 45.00000= 1.0000000000
```

Determinant Of 10 x 10 Matrix...

Objective

To find the value of Determinant of a given matrix of order up to 10 x 10. Theoretically the algorithm can be used to calculate the Determinant of an N x N Matrix, where N ∈ (1,∞].



Code (Det.pas)

```
Program Determiment_;
type matrix=record
matrix:array[1..10,1..10] of integer;
rows,cols:integer;
end;
{Gobal Declarations}
var i,j:matrix;var k,l:integer;
{Procedure Declarations}
procedure eleminator(matra:matrix;row,col:integer;var
answer:matrix);
var eletemp:matrix;var i,j,ii,jj,k:integer;
begin
ii:=0;
for i:=1 to matra.rows do
if i<>row then begin
ii:=ii+1;
jj:=0;
        for j:=1 to matra.cols do
        if j<>col then begin
        jj:=jj+1;eletemp.matrix[ii,jj]:=matra.matrix[i,j];
        end;
end;
eletemp.rows:=matra.rows-1;
eletemp.cols:=matra.cols-1;
answer:=eletemp;
end;
{Function Declarations}
function pow(base,power:integer):integer;
begin
if power=0 then pow:=1 else
pow:=base*pow(base,power-1);
end;
function determinent(x:matrix;wrow:integer):integer;
var j,k:integer;var l:matrix;
begin
j:=0;
if x.rows<>x.cols then begin
writeln('Invalid Matrix');halt;
end;
```

Output

```
Enter Number of Rows : 4
Enter Number of Coloums : 4
Now Enter The Matrix:-
4
3
2
5
6
8
3
2
5
4
5
7
2
3
4
5

4|3|2|5|
6|8|3|2|
5|4|5|7|
2|3|4|5|
=122
```

Determinant Of 10 x 10 Matrix.

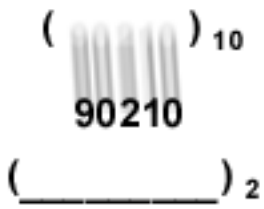
Code (Det.pas)

```
if x.rows=1 then begin
determinent:=x.matrix[1,1];exit;end;
for k:= 1 to x.cols do begin
eleminator(x,wrow,k,l);
j:=j+pow(-1,((k+wrow) mod 2))* x.matrix[wrow,k]* determinent(l,1)
end;
determinent:=j;
end;
{Main Module}
begin
Write('Enter Number of Rows : ');readln(i.rows);
Write('Enter Number of Coloums : ');readln(i.cols);
Writeln('Now Enter The Matrix:-');
for k:=1 to i.rows do
for l:=1 to i.cols do readln(i.matrix[k,l]);
writeln;
for k:=1 to i.rows do begin
for l:=1 to i.cols do write(i.matrix[k,l],'|');
writeln;
end;
writeln('=',determinent(i,1));
readln;
end.
{Code Concludes}
```

Base To Base Conversion-II...

Objective

To extend the program 'Base To Base Conversion-I' and to overcome its limitations.



Code (Dbase2.pas)

```
program decil2Abase;
var i,j,k:integer;
var n:string;
function d2Abase(x,y:integer):string;
var n,o:string;
var k,m:integer;
begin
n:='';m:=0;
while x>0 do begin
k:=x mod y; str(k,n);
o:=n+o;
x:=x div y;
m:=m+1;
end;
delete(o,m+1,length(o)-m);
d2abase:=o;
end;
function A2Dbase(x:string;y:integer):integer;
var n,o:string;
var i,j,k,l:integer;
begin
k:=1;l:=0;
while length(x)>0 do
     begin
     o:=copy(x,length(x) ,1);
     val(o,i,j);
     l:=l+i*k;
     delete(x,length(x),1);
     k:=y*k;
     end;
A2dbase:=l;
end;
function menu:integer;
var k:integer;
begin
writeln;
writeln('----------');writeln('Main Menu');writeln('----------
');
writeln('1.Decimal to any base');writeln('2.Any base to
decimal');
writeln('0.Quit');
writeln('----------');
write('Enter Your Choice ');
readln(k);
if (k>2) or (k<0) then
begin                                          {Contd.....}
```

Output

```
Welcome To Base Calculator

----------
Main Menu
----------
1.Decimal to any base
2.Any base to decimal
0.Quit
----------
Enter Your Choice 1

Enter the Number: 9021
Enter target Base: 2
9021 in base 2 is= 10001100111101

----------
Main Menu
----------
1.Decimal to any base
2.Any base to decimal
0.Quit
----------
Enter Your Choice 2
Enter the Number: 10001100111101
Enter its Base: 2
10001100111101 in base 10 is= 9021
```

Base To Base Conversion-II.

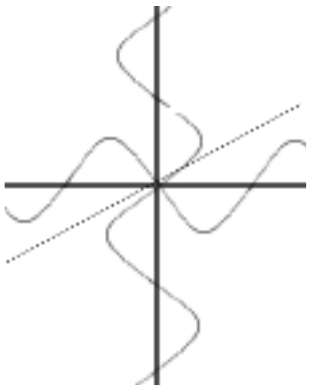
Code (Dbase2.pas)

```
writeln('Invalid Choice..');
k:=menu;
end;
menu:=k;
end;
{Main Module}
begin
k:=0;
Writeln('          Welcome To Base Calculator');
repeat
case k of
1:begin
write('Enter the Number: ');readln(i);
Write('Enter target Base: ');readln(j);
writeln(i,' in base ',j,' is= ',d2abase(i,j));
end;
2:begin
write('Enter the Number: ');readln(n);
Write('Enter its Base: ');readln(j);
writeln(n,' in base 10 is= ',a2dbase(n,j));
end;
end;
k:=menu;
until k=0;
writeln('Bye-Bye');
end.
```

Drawing Graphs Of A Function.

Objective

To draw the graph of a pre defined function which gives x in terms of y. Ex: x=4y.



Code (Graph.pas)

```
var x,y,xx,yy:real;
var temp:array[1..4] of integer;
begin
for temp[2]:=0 to 24 do
begin
yy:=-temp[2];
y:=yy+12;
x:=sin(y);
xx:=x+40;
if temp[2]<>12 then
for temp[1]:=0 to 75 do
begin
if temp[1]=39 then write('|') else
if temp[1]=trunc(xx) then write('*') else write(' ');
end
else
for temp[1]:=0 to 75 do write('-');
writeln;
end;
readln;
end.
```

*Output (x=3*sin y)*

```
    |*
    | *
    | *
    |
  * |
  * |
    |*
    | *
    | *
------------
  * |
  * |
    |
    | *
    | *
    |*
   *|
```

*Output (x=0.5*y*y)*

```
   |
   |      *
   |    *
   |  *
   |*
---------------
   |*
   |  *
   |    *
   |      *
```

File I/O Operations...

Objective

To create a menu driven program which enables users to perform operations like File Copy, File Create/Overwrite, File append, File merge etc.

Code (Fileoprs.pas)

```
program fileoprs;
{Global Declarations}
var k:integer;
procedure fileshow;
var j:string;var i:char;var k:text;
var l:longint;
begin
l:=1;
Write('Enter The File Name : ');
readln(j);assign (k,j);reset(k);
writeln('File Contents of ',j,':-');
while not eof(k) do begin
read(k,i);write(i);
if eoln(k) then l:=l+1;
if l>18 then  begin
l:=1;
writeln;
write ('Press Enter To Continue...');read(i);
end;
end;
writeln;
write('Press Enter To Conclude...');read(i);
end;

procedure copycon;
var j:string;var i:char;var k:text;
begin
Write('Enter The File Name : ');
readln(j);assign (k,j);rewrite(k);
Writeln('Now Enter the Text :- ');
repeat
     repeat
     read(i);
     write(k,i);
     until  eoln;
until  eof;
close (k);
end;
procedure fileCopy;
var j:array[1..2] of string;var i:char;
var k,l :file of char;var m:longint;
begin
m:=1;                                          {Contd.....}
```

Output

```
Welcome To File Operations Demo
Main Menu

_____________________
1. Create/Overwrite File
2. View Contents Of A File
3. Append To A File
4. Copy A File
5. Merge Files
0. Quit

_____________________
Enter Your Choice:1
Enter The File Name : test.txt
Now Enter the Text :-
Hello there To alll^Z

Main Menu

_____________________
1. Create/Overwrite File
2. View Contents Of A File
3. Append To A File
4. Copy A File
5. Merge Files
0. Quit

_____________________
Enter Your Choice: 2
Enter The File Name : test.txt
File Contents of test.txt:-
Hello there To alll
Press Enter To Conclude...
```

File I/O Operations..

Code (Fileoprs.pas)

```
Write('Enter Source File Name : ');
readln(j[1]);Write('Enter Target File Name : ');
readln(j[2]);
assign (k,j[1]);reset(k);
assign (l,j[2]);rewrite(l);
write('Copying');
while not eof(k) do
begin
if m mod (filesize(k) div 9+1)=0 then write('.');
read(k,i);
write(l,i);
m:=m+1;
end;
writeln('Done');
close (k);
close(l);
end;
procedure filemerge;
var j:array[1..3] of string;var i:char;
var k,l,n:file of char;var m:longint;
begin
m:=1;
Write('Enter Source File Name 1 : ');readln(j[1]);
Write('Enter Source File Name 2 : ');readln(j[2]);
Write('Enter Target File Name : ');readln(j[3]);
assign (k,j[1]);reset(k);
assign (n,j[2]);reset(n);
assign (l,j[3]);rewrite(l);
write('Merging');
while not eof(k) do begin
if m mod (filesize(k) div 9+1)=0 then write('.');
read(k,i);write(l,i);m:=m+1;
end;
while not eof(n) do begin
if m mod (filesize(n) div 9+1)=0 then write('.');
read(n,i);write(l,i);m:=m+1;
end;
writeln('Done');close (k);close(l);close (n);
end;
procedure fileappend;
var j:string;var i:char;var k:text;
begin
Write('Enter The File Name : ');
readln(j);assign (k,j);reset(k);
writeln('File Contents of ',j,':-');
while not eof(k) do begin
read(k,i);write(i);
end;
append(k);
writeln;
Writeln('Now Enter the Text :- ');
```

{Contd.....}

File I/O Operations.

Code (Fileoprs.pas)

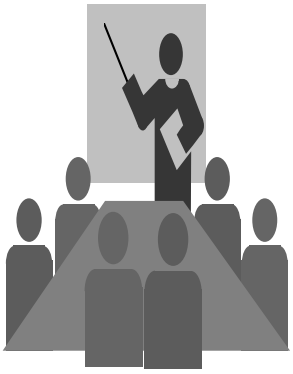
```
repeat
     repeat
     read(i);
     write(k,i);
     until  eoln;
until  eof;
close (k);
writeln;
writeln('Done');
end;

function menu:integer;
var k:integer;
begin
writeln;
writeln;
writeln('Main Menu');
writeln('_____________________');
writeln('1. Create/Overwrite File');writeln('2. View Contents Of A File');
writeln('3. Append To A File');writeln('4. Copy A File');
writeln('5. Merge Files');writeln('0. Quit');
writeln('_____________________');
write('Enter Your Choice: ');readln(k);
if (k<0) or (k>5) then begin
writeln('Invalid Input');
k:=menu;
end;
menu:=k;
end;

begin
writeln;
writeln('            Welcome To File Operations Demo');
writeln;
k:=0;
repeat
case k of
1:copycon;
2:fileshow;
3:fileappend;
4:filecopy;
5:filemerge;
end;
k:=menu;
until k=0;
writeln;
writeln('Bye-Bye');
end.
{Code Concludes}
```

Student Record Management System...

Objective

To create a Menu Driven Record Management Program which enables a users to perform following operations:-

1. Add New Student Record.
2. Update Any Student Record.
3. Delete Any Student Record.
4. View Students Record as:
   - View All Student Record.
   - Search for Record by a Name String.
   - Search for Record by a Class Name.
   - Search for Record by a Address String.
   - Search for Record by a Phone no. String.
   - Search for Record by a Name String.
   - Search for Record by a Scno Number.



Code (Stufdat.pas)

```
const
masterfile:string='d:\tp\bin\source\students\master.stu';
const
tempfile1:string='d:\tp\bin\source\students\temp1.stu';
const
tempfile2:string='d:\tp\bin\source\students\temp2.stu';
type students=record
scno:integer;
name:string[30];
phone:string[10];
adress:string[40];
class:integer;
end;
{Global Declarations}
var i:students;var k:integer;var j:char;
{Function Declarations}
function is_there(scno:integer):boolean;
var l:file of students;
var k:students;
begin
is_there:=false;
assign(l,masterfile);
reset(l);
while not eof(l) do begin
read(l,k);
if k.scno=scno then is_there:=true;
end;
end;                                    {Contd.....}
```

Output

```
Welcome To Intellective Academy

Main Menu
_____________________
1. Add New Student Record
2. Update Stuedent Record
3. Delete Student Record
4. View Student Records
0. Quit
_____________________
Enter Your Choice:1
Student Scno:        : 9
Enter Student name     : Ronak Jain
Enter Class          : 10
Enter Address        : 57 Hello Nagar
Enter Phone Number     : 90210
Add More (Y/N)n
Merging.................Done
Copying........Done

Main Menu
_____________________
1. Add New Student Record
2. Update Stuedent Record
3. Delete Student Record
4. View Student Records
0. Quit
_____________________
Enter Your Choice:0
Bye-Bye
```

Student Record Management System...

Code (Stufdat.pas)

```
function last_scno:integer;
var i:integer;var l:file of students;
var k:students;
begin
i:=0;
assign(l,masterfile);
reset(l);
while not eof(l) do begin
read(l,k);
i:=k.scno;
end;
last_scno:=i;
end;
procedure fileCopy(j1,j2:string);
var i:char;var k,l :file of char;var m:longint;
begin
m:=1;
assign (k,j1);reset(k);
assign (l,j2);rewrite(l);
write('Copying');
while not eof(k) do
begin
if m mod (filesize(k) div 9+1)=0 then write('.');
read(k,i);
write(l,i);
m:=m+1;
end;
writeln('Done');
close (k);
close(l);
end;
procedure filemerge(j1,j2,j3:string);
var i:char;var k,l,n:file of char;var m:longint;
begin
m:=1;
assign (k,j1);reset(k);
assign (n,j2);reset(n);
assign (l,j3);rewrite(l);
write('Merging');
while not eof(k) do begin
if m mod (filesize(k) div 9+1)=0 then write('.');
read(k,i);write(l,i);m:=m+1;
end;
while not eof(n) do begin
if m mod (filesize(n) div 9+1)=0 then write('.');
read(n,i);write(l,i);m:=m+1;
end;
writeln('Done');close (k);close(l);close (n);
end;
```

{Contd.....}

Student Record Management System...

Code (Stufdat.pas)

```
procedure addnew;
var n:integer;var j:char;
var l:file of students;var m:students;
begin
assign(l,tempfile1);
rewrite(l);
j:='y';
n:=last_scno;
while upcase(j)='Y' do
with m do
begin
n:=n+1;
scno:=n;
writeln('Student Scno:          : ',scno);
write  ('Enter Student name      : ');read(name);
write  ('Enter Class             : ');readln(class);
write  ('Enter Address           : ');readln(adress);
write  ('Enter Phone Number      : ');readln(phone);
write  (l,m);
write('Add More (Y/N)');readln(j);
end;
filemerge(masterfile,tempfile1,tempfile2);
filecopy(tempfile2,masterfile);
end;
function fileupdate(i :students):boolean;
var k,j:file of students;var m,l:integer;var n:students;
begin
fileupdate:=false;
assign(k,masterfile);assign(j,tempfile1);
reset(k);rewrite(j);
while not eof(k) do begin
read(k,n);
if n.scno=i.scno then begin
write(j,i);fileupdate:=true; end
else write(j,n);
end;
filecopy(tempfile1,masterfile);
end;
function filedelete(i :students):boolean;
var k,j:file of students;var m,l:integer;var n:students;
begin
filedelete:=false;
assign(k,masterfile);assign(j,tempfile1);
reset(k);rewrite(j);
while not eof(k) do begin
read(k,n);
if n.scno<>i.scno then begin
write(j,n);
end
```

{Contd.....}

Student Record Management System...

Code (Stufdat.pas)

```
else
filedelete:=true;
end;
filecopy(tempfile1,masterfile);
end;

procedure showscno(x:students);
var i:file of students;j:students;
begin
if not is_there(x.scno) then begin
writeln('Record Not Found');
exit;
end;

assign(i,masterfile);reset(i);

while not eof(i) do begin
read(i,j);
if j.scno=x.scno then with j do
begin
writeln ('Scno Number: ',scno);
writeln ('Name       : ',name);
writeln ('Class      : ',Class);
writeln ('Address    : ',adress);
writeln ('Phone      : ',Phone);
readln;
end;
end;
end;

procedure showname(x:students);
var i:file of students;var j:students;var k:integer;
begin
k:=-1;
assign(i,masterfile);reset(i);
while not eof(i) do begin
read(i,j);
if pos(x.name,j.name)>0 then with j do
begin
k:=k+1;
writeln ('Scno Number: ',scno);
writeln ('Name       : ',name);
writeln ('Class      : ',Class);
writeln ('Address    : ',adress);
writeln ('Phone      : ',Phone);
writeln;
if k>=4 then begin
k:=0;
write('Press Enter To Continue');
readln;
end;                                                    {Contd....}
```

Student Record Management System...

Code (Stufdat.pas)

```
end;
end;
   if k>-1 then begin
      write('Press Enter To Conclude');
      readln;
      end
      else
      writeln('Record Not Found');
end;

procedure showclass(x:students);
var i:file of students;var j:students;var k:integer;
begin
k:=-1;
assign(i,masterfile);reset(i);
while not eof(i) do begin
read(i,j);
if x.class=j.class then with j do
begin
k:=k+1;
writeln ('Scno Number: ',scno);
writeln ('Name       : ',name);
writeln ('Class      : ',Class);
writeln ('Address    : ',adress);
writeln ('Phone      : ',Phone);
writeln;
if k>=4 then begin
k:=0;
write('Press Enter To Continue');
readln;
end;
end;
end;
if k>-1 then begin
write('Press Enter To Conclude');
readln;
end
else
writeln('Record Not Found');
end;

procedure showadress(x:students);
var i:file of students;var j:students;var k:integer;
begin
k:=-1;
assign(i,masterfile);reset(i);
while not eof(i) do begin
read(i,j);
if pos(x.adress,j.adress)>0 then with j do
```

{Contd....}

Student Record Management System...

Code (Stufdat.pas)

```
begin
k:=k+1;
writeln ('Scno Number: ',scno);
writeln ('Name       : ',name);
writeln ('Class      : ',Class);
writeln ('Address    : ',adress);
writeln ('Phone      : ',Phone);
writeln;
if k>=4 then begin
k:=0;
write('Press Enter To Continue');
readln;
end;
end;
end;
if k>-1 then begin
write('Press Enter To Conclude');
readln
end
else
writeln('Record Not Found');
end;

procedure showphone(x:students);
var i:file of students;var j:students;var k:integer;
begin
k:=-1;
assign(i,masterfile);reset(i);
while not eof(i) do begin
read(i,j);
if pos(x.phone,j.phone)>0 then with j do
begin
k:=k+1;
writeln ('Scno Number: ',scno);
writeln ('Name       : ',name);
writeln ('Class      : ',Class);
writeln ('Address    : ',adress);
writeln ('Phone      : ',Phone);
writeln;
if k>=4 then begin
k:=0;
write('Press Enter To Continue');
readln;
end;
end;
end;
if k>=0 then begin
write('Press Enter To Conclude');
readln;
end
else
writeln('Record Not Found');
end;
```

{Contd.....}

Student Record Management System...

Code (Stufdat.pas)

```
procedure showall;
var i:file of students;var j:students;var k:integer;
begin
k:=-1;
assign(i,masterfile);reset(i);
while not eof(i) do with j do
begin
read(i,j);
k:=k+1;
writeln ('Scno Number: ',scno);
writeln ('Name       : ',name);
writeln ('Class      : ',Class);
writeln ('Address    : ',adress);
writeln ('Phone      : ',Phone);
writeln;
if k>=4 then begin
k:=0;
write('Press Enter To Continue');
readln;
end
end;
if k>=0 then begin
write('Press Enter To Conclude');
readln
end
else
writeln('Record Not Found');
end;

function Viewmenu:integer;
var k:integer;
begin
writeln;
writeln;
writeln('View Menu');
writeln('_____');
writeln('1. View All Student Records');writeln('2. View Student Record By Scno');
writeln('3. View Student Records By Name');writeln('4. View Student Records By Class');
writeln('5. View Student Records By Phone');writeln('6. View Student Records By Address');
writeln('0. Back');
writeln('_____');
write('Enter Your Choice: ');readln(k);
if  (k<0) or (k>6)   then
begin
writeln('Invalid Input');
k:=viewmenu;
end;
viewmenu:=k;
end;
```

{Contd.....}

Student Record Management System...

Code (Stufdat.pas)

```
function mainmenu:integer;
var k:integer;
begin
writeln;
writeln;
writeln('Main Menu');
writeln('_____');
writeln('1. Add New Student Record');writeln('2. Update Stuedent Record');
writeln('3. Delete Student Record');writeln('4. View Student Records');
writeln('0. Quit');
writeln('_____');
write('Enter Your Choice: ');readln(k);
if (k<0) or (k>4) then begin
writeln('Invalid Input');
k:=mainmenu;
end;
if k=4 then k:=k+viewmenu;
mainmenu:=k;
end;

procedure get(n:integer;var ans:students);
begin
if n=0 then begin
write('Enter Scno        :');readln(ans.scno);
end;
if n=1 then begin
write('Enter Name        :');readln(ans.name);
end;
if n=2 then begin
write('Enter Class       :');readln(ans.Class);
end;
if n=3 then begin
write('Enter Address     :');readln(ans.Adress);
end;
if n=4 then begin
write('Enter Phone Number:');readln(ans.Phone);
end;
writeln;
end;
{Main Module}
begin
k:=0;
Writeln;
Writeln('          Welcome To Intellective Academy');
Writeln;
repeat
case k of
1: addnew;
2: begin{update get(0,i);if end;}
```

{Contd.....}

Student Record Management System...

Code (Stufdat.pas)

```
if is_there(i.scno) then begin
get(1,i);get(2,i);get(3,i);get(4,i);
write('Update This Record (Y/N) : ');readln(j);
if upcase(j)='Y' then begin
fileupdate(i);
writeln('Record  Updated');
end;
end
else
writeln('Scno Number Not Found');
end;
3: begin {delete}
get(0,i);
if is_there(i.scno) then begin
showscno(i);
write('Delete This Record (Y/N) : ');readln(j);
if upcase(j)='Y' then begin
filedelete(i);
writeln('Record Deleted');
end;
end
else
writeln('Scno Number Not Found');
end;
5:showall;
6:begin
get(0,i);showscno(i);
end;
7:begin
get(1,i);showname(i);
end;
8:begin
get(2,i);showclass(i);
end;
9:begin
get(4,i);showphone(i);
end;
10:begin
get(3,i);showadress(i);
end;
end;
k:=mainmenu;
until k=0;
writeln;
writeln('Bye-Bye');
end.
{Code Concludes}
```