Date ___/___/___
Page ___

initialization

→ ar[0] = 10;
ar[1] = 20;

int ar[5] = {10,20,50,40,50}

* write a program to store mark of Five Subject in arry and Find per. using arry.

```
Void main()
{
  Float sub[5], per;
  clrscr();
  printf("Dharmi chander:\n IU1983830007");
  printf("enter the value of marks");
  scanf("%f", &sub[0]);
  printf("Enter 2 sub mark:");
  Scanf("%f", &sub[1]);
  printf("enter 3 sub mark:");
  scanf("%f", &sub[2]);
  Printf("enter 4 sub mark:");
  Scanf["%f", &sub[3]];
  printf["enter 5 sub mark:"];
  Scanf["%f", &sub[4]];
  avg = (sub[0] + sub[1] + sub[2] + sub[3]
    + sub[4]) / 5;
```

```
printf(" avg of 5 sub %.f ", avg);
getch();
}
```

* With for loop.

```
Void main()
{
  float sub[5], avg, i;
  clrscr();
  for(i=0; i<5; i++)
    {
      printf("Enter marks:");
      scanf("%f", &sub[i]);
    }
  avg = (sub[0] + sub[1] + sub[2] + sub[3] +
         sub[4]) / 5;
  printf("avg of 5 sub %.f", avg);
  getch();
}
```

Date ___/___/___
Page ___

Write a program sum of n element using array

```
Void main()
{
  int arr[100], sum, i, n;
  clrscr();
  sum = 0
  Printf("How many no :");
  scanf("%d", &n);

  for(i=0; i<n; i++)
    {
      printf("Enter element");
      scanf("%d", &arr[i]);
      sum = sum + arr[i];
    }
  printf("Sum = %d", sum);
  getch();
}
```

A
ABA
ABCBA
ABCDCBA

```
Void main()
{
    char i, j, k, l;
    int n;
    clrscr();
    printf("Enter value of n");
    scanf("%d", &n);

    for(i='A'; i<'A'+n; i++)
    {
        for(j='A'; j<=i; j++)
        {
            printf("%c", j);
        }
        for(k=i-1; k>='A'; k--)
        {
            printf("%c", k);
        }
        printf("\n");
    }
    getch();
}
```

- A string null terminated array.
- When declaring a character that will hold a string, we need to declare it one char. long longer than we largest string that it will hold.
- Ex:-

to declar an array str that can hold 10 char string.

```
char str[11];
```

* write a program to take input from user then print it.

```
Void main()
{
 char str[100];
 clrscr();
 printf("Enter the String:");
 scanf("%s", str);
 printf("your string is %s", str);
 getch();
}
```

```
Void main()
{
 char str[100];
 clrscr();
 printf("Enter the string:");
 gets(str);
 puts(str);
 get();
}
```

gets(str) => taking string.
puts(str) => printing string.

Date __/__/__
Page

```
* Void main()
  {
    char str[10];
    clrscr();
    ~~str="India";~~
    strcpy(str, "India");
    ~~Print~~ puts(str);
    getch();
  }

* ~~the~~ void main()
  {
    char str[] = {'I', 'n', 'd', 'i', 'a'};
    clrscr();
    puts(str);
    getch()
  }

* Void main()
  {
```

Date / /
Page

String

- Constant is list of char, and close it double quotes.
- we do not add to null, to the end of string function menually, but the compiler automatically.

* String manipulation function:- if u want to use this function u have to put <Stdio.h> library.

1) Strlen(s1):- returns the lenght of s1.

2) Strcpy(s1,s2):- copys s2 in the s1.

3) Strcat(s1,s2):- cat concatnats concatnats s2 onto s1.

4) Strcmp(s1,s2):- returns 0 if s1 and s2 are same, <0 if s1 < s2, >0 if s1 > s2

5) Strchr(s1,ch):- returns a pointer to the first occurs of ch in s1.

6) Strstr(s1,s2):- returns a pointer to the first s2 in s1

Date __/__/__
Page

Write a program to Find lenght of String using strlen Function given by user.

```
#include<stdio.h>

Void main()
{
    Char str[100];
    int l;
    clrscr();

    printf("enter the string:");
    gets(str);

    l = strlen(str);
    Printf("length of %s is %d" str,(l));

    getch();
}
```

Q. write a progreams to find string of length using without using any string function

```
#include<string.h>
  void main()
  {
    char str[100];
    int i;
    clrscr();

    printf("Enter the String ");
    gets(str);
    i=0;

    while(str[i]!='\0')
      i++;

    printf("lenght of %s is %d", str, (i));

    getch();
  }
```

Date __/__/__
Page

2-D ARRAY.

- C support multi dimension array the simplist form of multi dimensnal array is a dimensitonal array.
- to declare to 2-D integer array D of size 10,20. (int d[10][20];)

- 2D - ARRAY are stored in a rows column matrix. where the left index. Indicats to rows and write the indecents the Column.

arr[3][4] = {1,2,3,4,5,6,7,8,9,10,11,12};

	0	1	2	3
0	1	2	3	4
1	5	6	7	8
2	9	10	11	12

Write a program to initialize 2-D array and display it in matrix form.

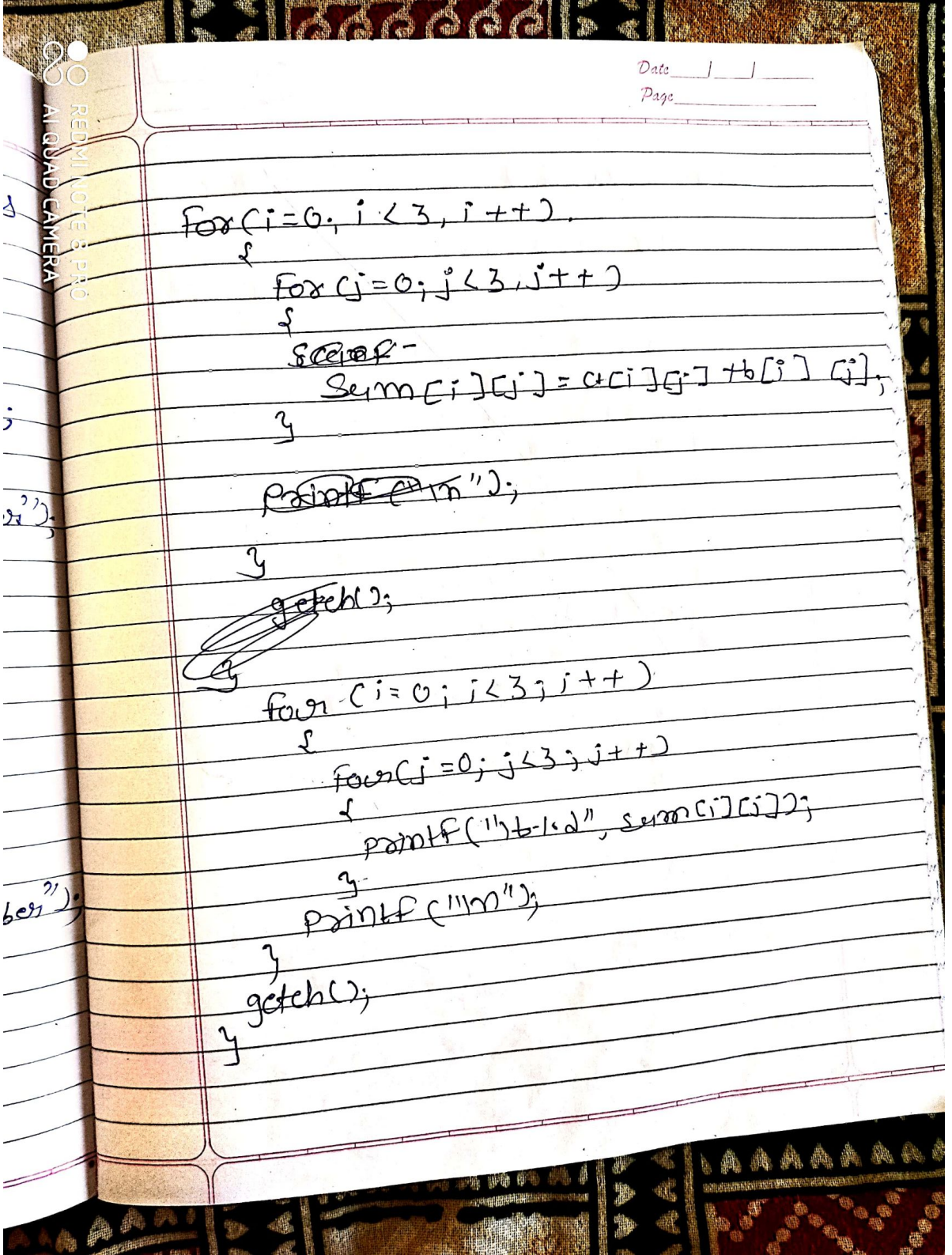
```
Void main() int a[10][10],i,j,r,c;
{clrscr();

Printf("Enter no of rows:");
Scanf("%d",&r);
```

```
printf(" enter no of column");
scanf(" %d", &c);

for (i=0; i<r; i++)
 {
   for (j=0; j<c; j++)
    {
      printf(" enter element");
      scanf("%d", &a[i][j]);
    }
 }

for (i=0; i<r; i++)
 {
   for (j=0; j<c; j++)
   {
   printf(" ~~enter~~ %d", a[i][j]);
   }
   pf("\n");
 }
 getch();
}
```

Date __/__/__
Page

Write a program to take 2 matrices from user and display sum of two matrics. (3×3).

```
Void main()
{
    int a[3][3], b[3][3], sum[3][3], i, j;
    clrscr();
    printf("Enter elements for first number");
    for(i=0; i<3; i++)
    {
        for(j=0; j<3; j++)
        {
            printf("Enter element");
            scanf("%d", &a[i][j]);
        }
    }
    printf("Enter elements for second number");
    for(i=0; i<3; i++)
    {
        for(j=0; j<3; j++)
        {
            printf("Enter element");
            scanf("%d", &b[i][j]);
        }
    }
```



```
for(i=0; i<3; i++)
{
    for(j=0; j<3; j++)
    {
        scanf-
        sum[i][j] = a[i][j] + b[i][j];
    }

    printf("\n");
}
getch();

for(i=0; i<3; i++)
{
    for(j=0; j<3; j++)
    {
        printf("%d", sum[i][j]);
    }
    printf("\n");
}
getch();
}
```

Date __/__/__
Page

```
Void main()
{
   int a[3][3];

   a[0][0] = 7;
   a[0][1] = 18;
   a[0][2] = 20;
   a[1][0] = 8
   a[1][1] = 9
   a[1][2] = 10
   a[2][0] = a[2][1]
   a[2][2] = 4

   for (i=0; i<3, i++)
   {
      for (j=0; j<3; j++)
      {
         printf("%d", a[i][j]);
      }

   printf("\n");
   }

   getch();
}
```

Date ___/___/___
Page ___

Write a program to Find sum of diagonal element of Matrixe.

```
Void main()
{
  int a[10][10], r, c, i, j, sum=0;
  clrscr();
  Start:
  printf("Enter the number of rows:");
  Scanf("%d", &r);
  printf("Enter the number of column:");
  Scanf("%d", &c);

  if(r != c)
  {
    printf("not a squarel matrix");
    goto start;
  }
  // take matrix

  printf("Enter the elements for matrix:");

  for(i=0; i<3, i++)
  {
    for(j=0; j<3; j++)
    {
      printf("Enter the elemet");
      Scanf("%d", &a[i][j]);
    }
}
```

```
for (i=0; i<r; i++)
{
    for (j=0; j<c; j++)
    {
        if (i==j)
        {
            sum = sum + a[i][j];
        }
    }
}

printf("Sum = %d", &sum);
getch();
}
```

Q. write a prog. To Transpose given matrix;

```
Void main()
{
    int a[10][10], r, c, i, j;
    int t[10][10];
    Start clrscr();
    start:
    printf("Enter the value of r:");
    Scanf("%d", &r);
    printf("Enter the value of c:");
    Scanf("%d", &c);
```

```
if (r != c)
{
    printf("Not a squared matrix");
    goto start;

// take matrix.

printf("Enter the matrix elements:");
    for(i=0; i<3; i++)
    {
        for(j=0; j<3; j++)
        {
            printf("Enter the element:");
            scanf("%d", &a[i][j]);
        }
    }
for(i=0, j<r; i++)
{
    for(j=0; j<c; j++)
    {
        t[j][i] = a[i][j]
    }
}
printf("transposed matrix:");
for(i=0; i<3; i++)
getch();
    for(j=0; j<3; j++)
    {
        printf("Enter the element:");
        scanf("%d", &a[i][j]);
```

☆ Multi dim.

C allows array of more than two D. The general form of multi dimensional array is.

datatype [size 1] [size 2] ... [size n]

Arrays of more than 3D. aren't often use because of amount of memory they require.

For ex:-

A 4D character array with dimension 10, 6, 9, 4, requires.

10 x 6 x 9 x 4 = 2160 bytes.

If array hold ₹ 2B then total 4320 Bytes would be needed. If array had double (8) bytes. then 17280B would be required.

```
Void main()
{
int a[50], i, max, n;
clrscr()
Printf ("enter total elements:");
Scanf ("%d", &n);
printf ("enter elements:")
for(i=0; i<n; i++)
  {
    printf("enter the elemet%d:", i+1);
    Scanf ("%d", &a[i]);
  }
max = a[0];
for (i=1; i<n; i++)
  {
    if(a[i] > max)
      max = a[i];
  }
Printf(" max elemet is": %d", max);
  getch();
}
```

Date __/__/__
Page ____

✯ Sorting:-

A sorting algoarithom is use to rearranging a given array according to a compar ~~eles~~ opreter of ele

Shorting can be perform in various cases in shorting algo orithem.

- The term storting spelle ar

- Through the ~~state~~ cexy tho ~~deay ses~~ Short the deay

✯ Types of sorting:-

1) bubble short
2) Selection
3) Murge } divied and concer.
4) quick }
5) Inserition
6) bucket
7) radix.
8) Hip

Date __/__/__
Page

Bubble Sort

Ex:-

15 9 13 12 5 4 1 7

1st Pass 9 15 13 12 5 4 1 7

9 13 15 12 5 4 1 7

9 13 12 15 5 4 1 7

9 13 12 5 15 4 1 7

9 13 12 5 4 15 1 7

9 13 12 5 4 1 15 7

9 13 12 5 4 1 7 15

2nd pass 9 13 12 5 4 1 7 15

9 13 12 5 4 1 7 15

9 12 13 5 4 1 7 15

9 12 5 13 4 1 7 15

9 12 5 4 13 1 7 15

9 12 5 4 1 13 7 15

9 12 5 4 1 7 13 15

Date __/__/__
Page

3 pass : 9 12 5 4 1 7 13 15

9 5 12 4 1 7 13 15

9 5 4 12 1 7 13 15

9 5 4 1 12 7 13 15

9 5 4 1 7 12 13 15

4 pass

9 5 4 1 7 12 13 15

5 9 4 1 7 12 13 15

5 4 9 1 7 12 13 15

5 4 1 9 7 12 13 15

5 4 1 7 9 12 13 15

5 pass 5 4 1 7 9 12 13 15

4 5 1 7 9 12 13 15

4 1 5 7 9 12 13 15

6th Post 4 1 5 7 9 12 13 15

1 4 5 7 9 12 13 55

Date __/__/____
Page ____

WAP to merge two one d array.

```
Void main()
{
  int a[50], b[50], m[100], i, n1, n2, j;
  clrscr();
  printf("Enter total elements of array
        1:");
  Scanf("%d", &n1);
  printf("Enter total elements of array
        2:");
  ScanF("%d", &n2);
  printf("Enter Elements for 1st Array");

  for(i=0; i<n1; i++)
   {
     printf("Enter element %d", i+1);
     ScanF("%d", &a[i]);
   }

  for(i=0; i<n2; i++)
   {
     Printf("Enter element %d", i+1);
     ScanF("%d", &b[i]);
   }

  for(i=0; i<n1; i++)
   {
     m[i] = a[i];
   }
```

Date __/__/__
Page

```
for (i = n1, j = 0; i < n1 + n2; i++, j++)
{
    m[i] = b[j];
}

printf ("Merge Array is : \n");
for (i = 0; i < n1 + n2; i++)
{
    printf ("%d", m[i]);
}
getch();
}
```

✱ Storage class in C :-

(1) Auto :-

Variables that are declared inside a FUNCTION are called local variables. are supported to as automatic variables Local variables can be used only by statements that are inside the block in which the variable are declared.

The C language contains the keyword 'auto', which we can use to declare local variable. Auto variables are default so this key word virtually never be used.

Date / /
Page

2) Extern:-

with extern keyword, we can declare a variable without defining it that means when we need to refer a variable that is defined in another part of our program we can declare that variable using extern.

```
Void main()
{
    extern int a, b;
    clrscr();
    printf("%d %d", a, b);
    getch();
}
int a=10; d=11;  definetion.
```

Practical.

☆ Sorting program.

```
Void main()
{
  int a[50], n, i, j, temp;
  clrscr();
  printf("Enter elements:");
  scanf("%d", &n);
```

```
printf("enter element:");
for(i=0; i<n; i++)
{
    scanf("%d", &a[i]);
}

for(i=0; i<n-1; i++)
{
    for(j=0; i<n-i-j; j++)
    {
        if(a[j+1]<a[j])
        {
            temp = a[j];
            a[j] = a[j+1];
            a[j+1] = temp;
        }
    }
}

printf("Sorted array is:");
for(i=0; i<n; i++)
{
    printf("%d", a[i]);
}
getch();
}
```

Date __/__/__
Page

3) Static.

- Variable declare as static are permenent variable within their own Function and file they are not known outside their function or file, but they maintain their values between calles. When you apply static keywords to local variable, the compiler create permenent storage for it.
- Static local variable are very imp. of creation of Fun. bcoz several types of Funcation must perisove value between calls.

47 Register Variables:-

The register storeg specify apply only Two int, char, or pointer types. register specifier request the compiler to keep the value of variable in the register of the CPU. rather them in memory. where normal variable are stored. - this mean that operation on register variable could occer much fester then a normal variable becoz register variable actually held in the CPU. and did not requier a memory cycles to determin or modify its value.

acording need of user. this Function are calle User defined Functions.

- In the syntax return-type specify the type of data that the Function returns a Function may return any type of data and accept and data. the peramiter list is comma seprated list of variable name and their associated types. the peramter recive peremeter values of argument when the funcation is called as.
- A Function can be without peramiter in which case the peramter list can be empty.

Date / /
Page

Ex:1 // Function definition.

```
Void my_intro()
{
    Printf (" My name is Dhwani ");
    Printf("\n IU1983830007 ");
}
Void main()
{
    Clrscr();
    my_intro(); // Calling the Function.
    getch();
}
```

★ Function Prototype.

A Function prototype is the declaration of Function that specify the Function name, parameters and return type. it don't Contain Fun. body.

A function prototype gives information to the complier that Functions may be use in program.

```
clrscr();
void my_intro()
getch();
}
void my intro()
{
printf(" Dharm Chandel \n");
printf(" IU1983830007");
}
```

Date / /
Page

* with parameter with return value.

```
int add_numbers (int no1, int no2)
{
    int sum;
    sum = no1 + no2;
    return sum;
}

void main()
{
    int ans;
    clrscr();
    ans = add_numbers(20,30)
                    // calling the fun.
    printf("ans = %d", ans);
}

Void main()
{
    int ans;
    printf("Enter the no1");
    Scanf("%d", no1");
    Printf("Enter the no2");
    Scanf("%d" no2");
    ans = add_numbers(no1, no2)
    getch();
}
```

(U.D.F)

Date ___/___/___
Page ___

Functions

★ Advantages of Functions. (user defined)

1. By using Func. we can avoid re-coding of same code or logic in the program.
2. We can call Functions any number of time in the program.
3. We can understand large program easy when it is divided into Functions.
4. We can reuse the same code.

★ Write a Function to find maximum of two numbers. Also call this Fun. through main. Function will take two no as argument and return max no.

```
int max_number (int no1, int no2); // Function Prototype.

Void main()
{
  int a,b, max;
  Clrscr();
  Printf ("Enter the value of a:");
  Scanf ("%d", &a);
  Printf ("Enter the value of b:");
  Scanf ("%d", &b);
```

```
max = max_number(a,b); // function calling
printf("max of %d & %d is %d", a, b, max);
getch();
}

int max_number(int no1, int no2)
{
   int max;
   if(no1 > no2)
      max = no1;
   else
      max = no2;
   return max;
}
```

★ with per. No return value.

```
Void max_number(int no1, int no2);

Void main()
{
   int a, b;
   clrscr();
   printf("Enter the value of a:");
   printfscanf("%d", &a);
   printf("Enter the value of b:");
   Scanf("%d", &b);
```

```
}

Void max_number(int no1, int no2)
{
    int max;
    if (no1 > no2)
      max = no1;
    else
    max = no2;
    printf("max of %d & %d is %d.",
            no1, no2, max);
}
```

write a function to find area of rectangle and calling through main function (with, with).

```
{
    int a, b, ans;
    clrscr();
    printf("Enter the value of a:");
    scanf("%d", &a);
    printf("Enter the value of b:");
    scanf("%d", &b);
    ans = rec_fn(a, b);
    printf("ans of %d %d %d is %d",
                a, b, ans);
    getch();
}

int rec_fn(int a, int b);

{
    int ans;
    ans = a * b;
    return ans;
}
```

Date / /
Page

4. (with parameter no return value)

```
void rec_A(int, int);

void main()
{
int a,b;
clrscr();

printf(" enter a:");
scanf("%d", &a);
printf(" enter b:");
scanf("%d", &b);

rec_A(a,b);
getch();
}

void rec_A( int a, int b);
{
    int ans;
    ans = a*b;
    printf(" ans is %d & %d is %d",
                a, b, ans);
}
```