

23 July, 19

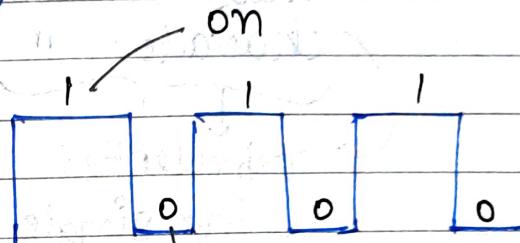
Operating Systems

does memory and resource management.

I O → electrical signals
(Binary bits)

SMPS - captures direct power supply and generates various voltages.

converts electric signals to digital current signal (DC)



- Digital Signals -
- can't travel upto long distances (energy loss)

→ Echogenic Signal = signals used under water (or sound waves)

→ Radio Waves = used for wireless communication (for mobile phones & TV)

Signal → which has freq. & amplitude
(no. of cycles per unit time) ← (Strength of signal)

C language - used for connectivity.

CLASSTIME / Page No.

Date / /

24 July, 19

Constant, variable & Keyword.

c constant

Primary
constant

Secondary
constant.

Integer constant

Real constant

character "

array

pointer

structure

union, etc.

represented

using single

quotes on same side

(' ') (' ')

NOTE:-

- Integer constant must have at least 1 digit
- By default, sign of integer constant is +ve.

Real constants

eg - 2.12

using mantissa &
exponent notation.

mantissa

2 E - 2 / 2 e - 2

↓
exponent.

char (1 byte)

max.size
 $\underline{255}$
 $(2^8 - 1)$

For signed system.
 +127 to -128.

Q How many ways are there to add 2 large nos.
 having more than 30 digits?

25 July, 19

1st program of C language

/ * * /

include <stdio.h>

void main()

{

Statement ;

Terminator

}

printf(); → used for displaying O/P.

Linux

compiler
gcc

editor.

vi editor.

scanf() → fetch I/O from user.
(left to right)

* Precedence order

1st

* / %

2nd

= (assignment)

$\frac{Q}{=}$

$$\begin{aligned} Q &= 2 * 3 / 4 + 4 / 4 + 8 - 2 + 5 / 8 \\ &= 1 + 1 + 8 - 2 + 0 \\ &= \underline{\underline{8}} \end{aligned}$$

LAB Practical → Putty (Windows)

- Username → riyagoel
- Password → 12345.
- New password → 25121997.

↓
(change using \$ passwd)

→ \$ vi hello.c → Enter
Press Insert key
↓

#include <stdio.h>

void main()

{

printf ("Hello"); }

Now press Esc]

? wq

On Terminal]

gcc hello.c (compile)

for execution]

./ a.out

for new directory]

[mkdir assignment]
cd

Q1 → # include <stdio.h>

```
int main () {
    int i;
    printf ("Enter a number");
    scanf ("%d", &i);
    if (i % 2 == 0)
        printf ("no. is even");
    else
        printf ("no. is odd");
}
```

Q2 → # include <stdio.h>

```
int main ()
```

```

int a, b;
printf ("Enter 1st no.");
scanf ("%d", &a);
printf ("Enter 2nd no.");
scanf ("%d", &b);
int temp;
temp = a;
a = b;
b = temp;
printf ("Swapped no.1 = %d\n", a);
printf ("Swapped no.2 = %d\n", b);
}

```

Q4 → #include < stdio.h>
int main()

```

{
    int a, b;
    int res = 1;
    int i;
    printf ("Enter base");
    scanf ("%d", &a);
    printf ("Enter power");
    scanf ("%d", &b);
    for (i = 1; i <= b; i++)
    {
        res *= a;
    }
    printf ("The power is %d", res);
}

```

Q5 → #include <stdio.h>
int main()
{
 int i, num;
 int flag = 0;
 printf ("Enter a no.");
 scanf ("%d", &num);
 for (i = 2; i <= num/2; i++)
 {
 if (num % i == 0)
 {
 flag = 1;
 break;
 }
 }
 if (flag == 1)
 printf ("no. is not prime");
 else
 printf ("no. is prime");
}

Q6 → #include <stdio.h>
int main()
{
 int a, b, c;
 printf ("Enter 3 nos.");
 scanf ("%d %d %d", &a, &b, &c);
 if (a >= b && a >= c)
 printf ("Greatest of the 3 nos. is = %d\n", a);
}

if ($b >= a \& b >= c$)
 printf ("Greatest of 3 nos. is = %.d
 b);

if ($c >= a \& c >= b$)
 printf ("Greatest of 3 nos. is = %.d
 c);

}

Q7 → #include <stdio.h>

int main()

{

int n, sum = 0;
 printf ("Enter no. upto which sum
 to be calculated");

scanf ("%d", & n);

int i;

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

{

sum += i;

}

printf ("Sum is = %.d\n", sum);

Q8 →

#include <stdio.h>

int main()

{

int q, r, n1, n2;

printf ("Enter the dividend");

```

scanf ("%d", &n1);
printf ("Enter divisor");
scanf ("%d", &n2);
q = n1/n2;
printf ("Quotient is = %d\n", q);
r = n1 % n2;
printf ("Remainder is = %d\n", r);
}

```

Q9 → #include <stdio.h>
int main()

```

{
    int num, res;
    int i;
    printf ("Enter any number");
    scanf ("%d", &num);
    for (i=1; i<=10; i++)
    {
        res = num * i;
        printf ("%d\n", res);
    }
}

```

Q10 → #include <stdio.h>
int main()

```

{
    int year;
    printf ("Enter year");
    scanf ("%d", &year);
    if (year%100!=0)
    {
        if (year%400==0)
}

```

`printf ("The entered year is a century year
as well as leap year");`

{

`else if (year % 4 == 0)`

`printf ("The entered year is a leap
year but not a century year");`

`else`

`printf ("Not a leap year");`

{

{

Fibonacci Series :-

`#include <stdio.h>`

`int main()`

{

`int f = 0, s = 1, t;`
`int n; int i;`

`printf ("Enter no. upto which
fibonacci series is to be calculated");`

`scanf ("%d", &n);`

`printf ("%d", f);` →
`printf ("%d", s);`

`for (i = 2; i < = n; i++)`

~~scanf ("%d", &t);~~
~~printf ("%d", t);~~

`t = f + s;`
`printf ("%d", t);`
`f = s;`

$s = t;$

{

Q Factorial of a no.

#include <stdio.h>
int main()

{

 int n, fact = 1;
 printf("Enter no.");
 scanf("%d", &n);
 while (n > 0)

{

 fact *= n;

 n--;

{

 printf("Factorial is %d\n", fact);

{

Q Decimal to binary :-

#include <stdio.h>
int main()

{

 int a[50] = {0};
 int num, i, j, k = 0;
 printf("Enter no.");
 scanf("%d", &num);
 i = num;
 while (i > 0)

{

$\text{int } r = \text{?};$
 $a[k+i] = r;$
 $i /= 2;$

}

$\text{for}(j=k; j >= 0; j--)$

}

}

$\text{printf } ("%.d", a[j]);$

}

=

Matrix multiplication

#include <stdio.h>

int main()

{ int i, j, k;

int a[10][10], b[10][10],
c[10][10];

printf ("Enter elements of 1st
matrix");

for (i=0; i<10; i++)

{

for (j=0; j<10; j++)

}

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

$a_{11} \ a_{12}$

printf ("Enter elements of 2nd
matrix");

for (i=0; i<10; i++)

{

```

for( j=0; j<10; j++)
{
    scanf ("%d", &b[i][j]);
}

//printf ("matrix multiplication :-");
for( i=0; i<10; i++)
{
    for( j=0; j<10; j++)
        c[i][j] = 0;
    for( k=0; k<10; k++)
        c[i][j] += a[i][k] *
            b[k][j];
}
printf ("Ans:-");
for( i=0; i<10; i++)
{
    for( j=0; j<10; j++)
        printf ("%d", c[i][j]);
}
    
```

Q Count no. of digits in an integer.

```
#include <stdio.h>
int main()
```

{

```
    int num;
    printf ("Enter no.");
    scanf ("%d", &num);
    int count = 0;
```

```
    while (num > 0)
```

{

```
        num / = 10;
        count++;
        num % 10;
```

}

```
    printf ("no. of digits are %d",
            count);
```

}

Q

Sum of digits of a no.

```
#include <stdio.h>
int main()
```

{

```
    int num; int sum = 0;
    printf ("Enter no.");
    scanf ("%d", &num);
```

```
    while (num > 0)
```

{

```
        int x = num % 10;
```

```

sum += x;
num /= 10;
}
calulation part
printf ("sum of digits : %d\n", sum);
}
    
```

GCD and Lcm of 2 nos.

```

#include <stdio.h>
#include <conio.h>
int gcd (int a, int b)
{
    if (b == 0)
        return a;
    else
        return gcd (b, a % b);
}
    
```

```

int main()
{
    int a, b, lcm;
    printf ("Enter 2 nos: ");
    scanf ("%d %d", &a, &b);
    int result = gcd (a, b);
    printf ("GCD of nos. is : %d\n", result);
}
    
```

// hcf * lcm = product of 2 nos.

```

lcm = (a * b) / result;
printf ("LCM of 2 nos. is : %d\n", lcm);
getch();
return 0;
}
    
```

Q Check if string is palindrome or not.

```
#include < stdio.h>
#include < string.h>
```

```
int main()
```

{

```
char word[50];
int flag = 1;
int i, j; int len;
gets(word);
len = strlen(word);
for (i=0; j=len-1; i<j; i++, j--)
```

{

```
if (word[i] != word[j])
```

{

```
flag = 0;
break;
```

}

}

```
if (flag == 1)
```

```
printf ("String is palindrome");
```

else

```
printf ("String is not palindrome");
```

return 0;

}

Q Reverse the case of a given character.

```
#include <stdio.h>
#include <ctype.h>
int main()
{
    char ch;
    printf("Enter a character");
    scanf("%c", &ch);
    if (isalpha(ch))
    {
        if (islower(ch))
            ch = toupper(ch);
        else if (isupper(ch))
            ch = tolower(ch);
    }
    else
        printf("Entered character is a special character");
    return 0;
}
```

19 Aug. 19

- Q Enter marks of 5 subjects & compute their average w/o using Arrays.
- Q Swap value of 2 variables w/o using temp variable.

```
#include <stdio.h>
int main()
{
    int a, b;
    printf("Enter 2 nos.");
    scanf("%d %d", &a, &b);
    a = a + b;
    b = a - b;
    a = a - b;
    printf("Swapped values : %d %d", a, b);
    return 0;
}
```

- Q WAP for computation of factorial of a no.

```
#include <stdio.h>
int main()
{
    int fact = 1;
    int num;
    printf("Enter no.");
    scanf("%d", &num);
    while(num > 0)
    {
```

```
fact *= num;
num--;
```

}

```
printf("Factorial : %d", fact);
```

Q Enter a Value. Print even & odd nos. upto
that no.

```
#include <stdio.h>
int main()
```

{

```
int i, num;
```

```
printf("Enter no.");
```

```
scanf("%d", &num);
```

```
printf("Even nos. are : ");
for (i=0; i<=num; i+=2)
```

{

```
printf("%d \t", i);
```

}

```
printf("Odd nos. are : ");
for (i=1; i<=num; i+=2)
```

{

```
printf("%d \t", i);
```

}

```
return 0;
```

}

Q Check whether a no. is a perfect sq. or not.

```

#include <stdio.h>
int main()
{
    int num, flag = 0;
    printf("Enter a no: ");
    scanf("%d", &num);
    int i = 1;
    while(i < num)
    {
        if(num == (i * i))
        {
            flag = 1;
            break;
        }
        i++;
    }
    if(flag == 0)
        printf("%d is not a perfect sq. num.");
    else
        printf("%d is a perfect sq. of %d", num, i);
    return 0;
}

```

Q Convert a no. from binary to decimal.

```

#include <iostream.h>
#include <stdio.h>
#include <math.h>
int main()

```

{

```

int num, r, res = 0;
printf("Enter a no. in binary
notation : ");
scanf("%d", &num);
int i = 0;
int j = num;
while (num > 0)

```

{

{

```

r = num % 10;
res += (r * pow(2, i));
num /= 10;
i++;

```

```

printf ("%d in decimal form is : %d", j,
       res);

```

```

getch();
return 0;

```

{

Q WAP to print ASCII table.

```
#include <stdio.h>
```

```
int main()
```

{

```
int i;
```

```
char ch;
```

```
for (i = 0; i < 256; i++)

```

{

```
printf ("%c = %d, ch = %d", ch, ch);
ch = ch + 1;
```

```
return 0;

```

{

26/8/19

CLASSTIME / Page No.
Date / /

* Make a flow chart for do while loop :-

① For Loop

for (initialization, condition, update)

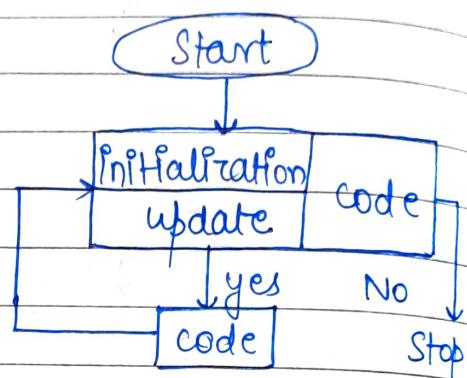
{
 code
 }
}

Q #include <stdio.h>
void main()

{
 int i;

 while (i <= 10);

{
 printf ("%d\n", i);
 i = i + 1;
}



infinite loop

control will not
reach here.

Q #include <stdio.h>
int main()

{

 int i = 1;

 while (i <= 32767)

{

 printf ("%d\n", i);
 i = i + 1;

Not in range,
so will go
into
infinite loop.

Q Calculate overtime pay of 10 employees. Overtime is paid at the rate of 12 RS/ hour worked above 40 hrs.

Q add first 7 terms of $\frac{1}{1!} + \frac{2}{2!} + \frac{3}{3!} + \dots$

```
# include <stdio.h>
```

```
int main()
```

{

```
int fact (int n);
```

```
int num, i;
```

```
printf ("First 7 terms : ");
```

```
for (i = 1; i <= 7; i++)
```

{

```
int x = fact(i);
```

```
float res += i/x;
```

}

```
printf ("%f", res);
```

```
return 0;
```

{

```
int fact (int n)
```

{

```
int fact = 1;
```

```
while (n > 0)
```

{

```
fact *= n;
```

{

```
n--;
```

```
return fact; }
```

(Q1).

#include <stdio.h>

int main()

{

int i, h;

for (i = 1; i <= 10; i++)

printf ("Enter work hrs of

employee no.%d", i);

scanf ("%d", &h);

if (h > 40)

{

 printf ("Overtime pay of the
 employee : %d",
 ((h - 40) * 12));

}

else

 printf ("Employee didn't work
 overtime");

{

return 0;

{

Q

WAP to print :-

A B C D E F G F E D C B A

A B C D E F F E D C B A

A B C D E F D C B A

A B C D F C B A

A B C C B A

A B B A

A A

A

Q WAP to calculate sum of 1st 7 terms

$$\frac{x-1}{x} + \frac{1}{2} \left(\frac{x-1}{x} \right)^2 + \frac{1}{3} \left(\frac{x-1}{x} \right)^3 + \frac{1}{4} \left(\frac{x-1}{x} \right)^4$$

..... where $x > 1/2$

If x is input through keyboard.

#include <stdio.h>

int main()

{

int i, j, x;

printf("Enter value of x");

scanf("%d", &x);

if ($x > 1/2$)

float res = 1;

{

or

for (i=1; i<=7; i++)

{

j = i;

while (j > 0)

{

res = $\left(\frac{x-1}{x} \right) * res.$

}

for (i=1; i<=7; i++)

{

float res = $(x-1)/x;$

sum += $\frac{1}{i} * pow(res, i);$

j--;

}

sum += $\frac{1}{i} * res;$

else printf("Value of x is less
than 0.5");

return 0;

}

2 Sept, 19

Arrays

- ① Linear & similar elements.
- ② Continuous memory allocation

`int a[10];` → provides info. to compiler
 ↘ size that an array a
 data type will be used.
 of elements

```
#include <stdio.h>
void main()
{
```

```
    int a[50]; // defined
    int i, j;
    for(i=0; i<50; i++)
    {
```

```
        printf("enter marks");
        scanf("%d", &a[i]);
    }
```

```
    int sum=0;
    for(i=0; i<50; i++)
        sum = sum + a[i];
    }
```

```
    avg = sum/50;
    printf("%d", avg);
}
```

→ We can also write `int a[] = {2, 3, 1, 5, ...}`

int a[30];
for(i=0; i<35; i++)

it won't give out of bound error.

int a[30]; sizeof(int) = 2 bytes
 sizeof array = 60 bytes.

continuous memory locations.

* Matrix (2-D) :-

two dimensions row
 column.

int a[10][3];

int a[10][3] = { { 1, 2, 3 }, { 4, 5, 6 } ... }

row 1

{ 2 D arrays are arrays of arrays }

OR

int a[10][3] = { 1, 2, 3, 4, 5, ... }

NOTE: It is mandatory to provide column no.
Row no. is optional

int a[][3];

Q1

Compute mean & standard deviation.

6, 12, 8, 5, 3, 2, 18, 30.

$$\sqrt{\frac{1}{n} \sum (x_i - \bar{x})^2} \rightarrow \text{Standard deviation}$$

\bar{x} = mean.

```
#include <Stdio.h>
```

```
int main()
```

```
{
```

```
int *a;
```

```
int n;
```

```
printf("Enter no. of elements");
```

```
scanf ("%d", &n);
```

```
a = new int [n];
```

```
printf("Enter elements");
```

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

```
{
```

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

```
}
```

```
float mean, sum = 0;
```

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

```
{
```

```
sum + a[i];
```

```
}
```

```
mean = (sum)/n;
```

```
printf("Mean is : %f", mean);
```

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

```
{
```

$p = \text{pow}(a[i] - \text{mean}, 2);$
 } float res + = p;
 }
 res = res/n;
 printf ("Standard deviation is: %f",
 sqrt(res));
 return 0;
}

Q Pick largest element from a 5x5 matrix

```

#include <stdio.h>
int main()
{
    int a[5][5];
    printf ("n Enter elements:");
    for (i=0; i<5; i++)
    {
        for (j=0; j<5; j++)
            scanf ("%d", &a[i][j]);
    }
}

```

large = a[0][0];
 for (i=0; i<5; i++)
 {
 for (j=0; j<5; j++)
 if (a[i][j] > large)

```
large = a[i][j];  
    } } }
```

```
printf ("The largest element is : %d",  
       large);
```

```
return 0;
```

Q Write a C program to shift an array
circularly left by 2 positions.

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
// Input array
```

```
for (i=0; i<2; i++)  
{
```

```
    temp = a[0];
```

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

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

```
}
```

```
a[n-1] = temp;
```

```
}
```

```
// Output array
```

25 Sept, 19

CLASSTIME / Page No.
Date / /

→ Concatenate 2 strings :-

#include <stdio.h>
int main()

{

char str1[12] = "Hello";
char str2[12] = "World";
int l1 = strlen(str1);
int l2 = strlen(str2);
char str3[l1 + l2];

use while

instead

of for.

{ for (int i = 0; str1[i] != '\0'; i++)
{ str3[k++] = str1[i];
}

for (int i = 0; str2[i] != '\0'; i++)

{

str3[k++] = str2[i];

{

printf("%s", str3);

{

↳ X

i = 0;

while (str3[i] != '\0')

{

printf("%c", str3[i]);

{

Q Find freq. of characters.

```
#include <stdio.h>
int main()
{
    char ch; int c=0;
    char str[100];
    scanf ("%s", &str);
    printf ("\n Enter chr. to be searched");
    scanf ("%c", &ch);

    int i=0;
    while (str[i] != '\0')
    {
        if (str[i] == ch)
            c++;
    }
    printf ("\n Count : %d", c);
    return 0;
}
```

Q Find no. of vowels, consonants, digits & white spaces.

Q Reverse a string using Recursion
Find length of string
Concatenate 2 strings.

Q C Program to copy a string.
~~Reverse~~ all characters in a string except Remove alphabets

Q Sort elements lexicographically.

```

(A1) #include <stdio.h>
      int main()
    {
        char str[100];
        gets(str);
        for(i=0; str[i]!='\0'; i++)
        {
            if(str[i] == 'a' || str[i] == 'e' || str[i] == 'i' || str[i] == 'o' || str[i] == 'u')
                c1++;
            else if(str[i] >= '0' && str[i] <= '9')
                c_dig++;
            else if(str[i] == ' ')
                c_space++;
            else
                c_con++;
        }
        printf("%d\n", c1);
        printf("%d\n", c_dig);
        printf("%d\n", c_space);
        printf("%d\n", c_con);
    }
    return 0;
}

```

A3

Find length of string

```
#include <stdio.h>
int main()
{
    int i=0, c=0;
    char str[100];
    gets(str);
    while (str[i] != '\0')
    {
        c++;
        i++;
    }
    printf("Length = %d", c);
    return 0;
}
```

A5 C program to copy a string

```
#include <stdio.h>
int main()
{
    char str [100];
    gets(str);
    int l = strlen(str);
    char dup dup[l];
    int i = 0;
    while (str[i] != '\0')
    {
        dup[i] = str[i];
        i++;
    }
}
```

```
// printf ("%s", dup);
```

```
int i = 0;
```

```
while (dup[i] != '\0')
```

```
{
```

```
printf ("%c", dup[i]);
```

```
i++;
```

```
}
```

```
return 0;
```

```
}
```

A7) Sort elements lexicographically

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
char str[100];
```

```
gets(str);
```

```
int i, j, temp;
```

```
int l = strlen(str);
```

```
for (i = 0; i < l - 1; i++)
```

```
{
```

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

```
{
```

```
if (strcmp(str[j], str[j + 1]) > 0)
```

```
temp = str[j];
```

```
str[j] = str[j + 1];
```

```
str[j + 1] = temp;
```

```
}
```

```
}
```

```
}
```

```

 $i = 0;$ 
while (str[i] != '10')
{
    printf ("%c", str[i]);
    i++;
}
return 0;
}.

```

A8) Remove all char except alphabet

```

#include <stdio.h>
int main()
{
    char str[100];
    gets(str);
    int k = 0;
    char dup[100];
    int i = 0;
    while (str[i] != '10')
    {
        if (str[i] >= 'a' && str[i] <= 'z')
        {
            dup[k++] = '0';
        }
        else
        {
            dup[k++] = str[i];
        }
        i++;
    }
    for (i = 0; i < k; i++)
    {
        printf ("%c", dup[i]);
    }
    return 0;
}.

```