

THE NEOTIA UNIVERSITY

ASSIGNMENT - 01

ASSIGNMENT NAME: Introduction to DOS.

OBJECTIVE: Practice various DOS commands.

COMMANDS : Internal Commands: MD, CD, EDIT, COPY CON, COPY, TYPE, CLS, DIR, DEL, RD, VER, DATE, TIME etc.
External Commands: FORMAT, TREE, DELTREE, DISKCOPY, DISKCOMP, FIND etc.

PROCEDURE:

Switch on the UPS and monitor.

Then switch on the CPU.

After OS is loaded, go to start menu and click.

Go to MS-DOS prompt and click. You will see : C:\WINDOWS>

Type cd\ and then press Enter key. You will see: C :\>

Type your command and see the result.

ASSIGNMENT - 02

ASSIGNMENT NAME: Write a ‘C’ program to convert a temperature given in Degree Centigrade to Degree Fahrenheit & vice versa.

OBJECTIVE: ‘C’ program to convert a temperature from Degree Centigrade to Degree Fahrenheit & vice versa.

ALGORITHM : main()

```
{ float c1, c2, f1, f2;
printf ("\nEnter temperature in Degree Centigrade :");
scanf ("%f", &c1);
f1= (9*c1+160)/5;
printf ("\nResult in Fahrenheit is = %f",f1);
printf ("\nEnter temperature in Fahrenheit :");
scanf ("%f", &f2);
c2= 5*(f2-32)/9;
printf ("\nResult in Centigrade is = %f",c2);
}
```

THE NEOTIA UNIVERSITY

ASSIGNMENT - 03

ASSIGNMENT NAME: Write a ‘C’ program to calculate simple & compound interest.

OBJECTIVE: ‘C’ program to convert simple & compound interest.

ALGORITHM : main()

```
{ float si, ci,p,n,r,a;
printf("\nEnter principle amount,no. of years & rate of interest.");
scanf("%f %f %f",&p,&n,&r);
si=p*n*r/100;
a=p*pow((1+r/100),n);
ci=a-p;
printf("\nSimple Interest is = %f",si);
printf("\nCompound Interest is =%f",ci);
}
```

ASSIGNMENT - 04

ASSIGNMENT NAME: Write a ‘C’ program to find the roots of a quadratic equation.

OBJECTIVE: ‘C’ program to find the roots of a quadratic equation.

ALGORITHM : main()

```
{ float a,b,c,d,r1,r2;
printf("\nEnter co-efficients:");
scanf("%f %f %f",&a,&b,&c);
d=b*b-4*a*c;
if(d>=0)
{
    r1=(-b+sqrt(d))/2*a;
    r2=(-b-sqrt(d))/2*a;
    printf("\nThe real roots are %f and %f",r1,r2);
}
else
    printf("\nThe roots are imaginary");
}
```

THE NEOTIA UNIVERSITY

ASSIGNMENT - 05

ASSIGNMENT NAME: Write a ‘C’ program to determine largest among 3 numbers.

OBJECTIVE: ‘C’ program to determine largest among 3 numbers.

ALGORITHM :main()

```
{ int a,b,c;
    printf("\nEnter three nos.:");
    scanf("%d %d %d",&a,&b,&c);
    if(a>b)
    { if(a>c)
        printf("\n%d is largest", a);
        else
            printf ("\n%d is largest",c);
    }
    else
    { if(b>c)
        printf("\n%d is largest",c);
        else
            printf("\n%d is largest",c);
    }
}
```

ASSIGNMENT - 06

ASSIGNMENT NAME: Write a ‘C’ program to determine whether an year is Leap Year or not.

OBJECTIVE: ‘C’ program to determine whether an year is Leap Year or not.

ALGORITHM :main()

```
{ int year;
    printf("\nEnter any year:");
    scanf("%d",&year);
    if(year%4==0 && year%100!=0 || year%400==0)
        printf("\n%d is Leap Year");
    else
        printf("\n%d is not Leap Year");
}
```

THE NEOTIA UNIVERSITY

ASSIGNMENT - 07

ASSIGNMENT NAME: Write a ‘C’ program to determine whether a number is prime or not.

OBJECTIVE: ‘C’ program to determine whether a number is prime or not.

ALGORITHM : main()

```
{ int num,i;
    printf("\nEnter any number.");
    scanf("%d",&num);
    i=2;
    while(i<=num-1)
    { if(num%i==0)
        { printf("%d is not prime no.",num); break;
        }
        i++;
    }
    if(i==num)
        printf("\n%d is prime no.");
}
```

ASSIGNMENT - 08

ASSIGNMENT NAME: Write a ‘C’ program to determine factorial of a given number.

OBJECTIVE: ‘C’ program to determine factorial of a given number.

ALGORITHM : main()

```
{ int i, fact=1,n;
    printf("\nEnter any no.");
    scanf("%d",&n);
    for(i=1;i<=n;i++)
        fact=fact*i;
    printf("\nFactorial of given no. %d is: %d",n,fact);
}
```

ASSIGNMENT - 09

ASSIGNMENT NAME: Write a ‘C’ program to determine Fibonacci series of n numbers

THE NEOTIA UNIVERSITY

OBJECTIVE: ‘C’ program to determine Fibonacci series of n numbers.

ALGORITHM : main()

```
{  
    int f0=0,f1=1,f2,num,count=0;  
    printf("\nEnter the limit:");  
    scanf("%d",&num);  
    printf("\nThe Fibonacci Series is : \n");  
    if(num<=0)  
        printf("\nWrong Input....!!!");  
    else if(num==1)  
        printf(" %d",f0);  
    else if(num==2)  
        printf(" %d %d",f0,f1);  
    else  
    {  
        count=3;  
        f2=f0+f1;  
        printf(" %d %d %d",f0,f1,f2);  
        while(count<num)  
        {  
            f0=f1;  
            f1=f2;  
            f2=f0+f1;  
            printf(" %d",f2);  
        }  
    }  
}
```

ASSIGNMENT - 10

ASSIGNMENT NAME: Write a ‘C’ program to convert Decimal number to Binary and vice-versa.

OBJECTIVE: ‘C’ program to convert Decimal number to Binary/ Octal/Hexadecimal numbers

ALGORITHM : main()

```
{  
    int i,dnum,bnum,rem;  
    printf("\nEnter any decimal number:");  
    scanf("%d",&dnum);  
    i=0;  
    bnum=0;
```

THE NEOTIA UNIVERSITY

```
while(dnum>0)
{
    rem=dnum%2;
    bnum=bnum+pow(10,i)*rem;
    num/=2;
    i++;
}
printf("\nThe equivalent binary number is = %d",bnum);
printf("\nEnter any binary number:");
scanf("%d",&bnum);
i=0;
dnum=0;
while(bnum>0)
{
    rem=bnum%10;
    dnum=dnum+pow(2,i)*rem;
    num/=10;
    i++;
}
printf("\nThe equivalent binary number is = %d",dnum);
}
```

ASSIGNMENT - 11

ASSIGNMENT NAME: Write a ‘C’ compute following series:
 $s = 1+x/1! + x^2/2! + x^3/3! + \dots\dots$

OBJECTIVE: ‘C’ program to compute the series upto ‘n’ numbers.

ALGORITHM : main()

```
{ int i,n,f=1;
float x,sum=1;
printf("\nEnter how many:");
scanf("%d",&n);
printf("\nEnter value of x:");
scanf("%f",&x);
for(i=1;i<=n;i++)
{
    f=f*i;
    sum+= pow(x,i)/f;
}
printf("\nThe summation of the series is : %f",sum);
}
```

THE NEOTIA UNIVERSITY

ASSIGNMENT - 12

ASSIGNMENT NAME: Write a ‘C’ program to generate the following triangle:

```
1  
0 1  
1 0 1  
0 1 0 1
```

OBJECTIVE: ‘C’ program to generate the triangles upto ‘h’ heights.

ALGORITHM : main()

```
{ int i, j, h,c;  
    printf("\nEnter height:");  
    scanf("%d",&h);  
    for(i=1;i<=h;i++)  
    { for(j=1;j<=i;j++)  
        { c=(i+j-1)%2; printf(" %d",c);  
        }  
        printf("\n");  
    }  
}
```

ASSIGNMENT - 13

ASSIGNMENT NAME: Write a ‘C’ program to compute GCD & LCM of given two numbers.

OBJECTIVE: ‘C’ program to compute GCD & LCM of given two numbers.

ALGORITHM : main()

```
{ int a,b,x,rem,l;  
    printf("\nEnter two positive integers :");  
    scanf("%d%d",&a,&b);  
    x=a*b;  
    if(a>b)  
    { rem=a%b;  
        while(rem!=0)  
        { a=b;  
            b=rem;  
            rem=a%b;  
        }  
        printf("\nGCD=%d",b);  
        l=x/b;  
        printf("\nLCM=%d",l);  
    }
```

THE NEOTIA UNIVERSITY

```
    }
else
{
    rem=b%a;
    while(rem!=0)
    {
        b=a;
        a=rem;
        rem=b%a;
    }
    printf("\nGCD=%d",a);
    l=x/a;
    printf("\nLCM=%d",l);
}
}
```

ASSIGNMENT - 14

ASSIGNMENT NAME: Write a ‘C’ program to compute Bubble sort.

OBJECTIVE: ‘C’ program to compute Bubble sort.

ALGORITHM : main()

```
{ int arr[max],n,i,j,temp;
printf("\n\t\tBUBBLE SORT");
printf("\n\t\t-----");
printf("\nEnter no. of elements to sort:");
scanf("%d",&n);
for(i=0;i<n;i++)
{
    printf("\nEnter value in arr[%d]:",i+1);
    scanf("%d",&arr[i]);
}
for(i=0;i<n-1;i++)
    for(j=0;j<n-1-i;j++)
        if(arr[j]>arr[j+1])
        {
            temp=arr[j+1];
            arr[j+1]=arr[j];
            arr[j]=temp;
        }
printf("\nThe sorted list is:\n");
for(i=0;i<n;i++)
    printf("%d\t",arr[i]);
getch();
}
```

THE NEOTIA UNIVERSITY

ASSIGNMENT - 15

ASSIGNMENT NAME: Write a ‘C’ program to compute Linear & Binary search.

OBJECTIVE: ‘C’ program to compute Linear & Binary search.

ALGORITHM : Linear Search:

```
main()
{ int arr[max],num,i;
  printf("\n\tInput values in the array.");
  for(i=0;i<max;i++)
  { printf("\n\tEnter element in arr[%d]:",i);
    scanf("%d",&arr[i]);
  }
  printf("\n\tThe elements are:\n");
  for(i=0;i<max;i++)
    printf("\t%d",arr[i]);
  printf("\n\tEnter element to search:");
  scanf("%d",&num);
  for(i=0;i<max;i++)
  { if(arr[i]==num)
    { printf("\n\t%d found in %dth position.",num,i);
      getch();
      exit(0);
    }
  }
  printf("\n\t%d not found.",num);
}
```

Binary Search:

```
main()
{ int arr[max],num,i,j,ch,lb,ub,mid,f,temp;
  printf("\n\tInput values in the array.");
  for(i=0;i<max;i++)
  {printf("\n\tEnter element in arr[%d]:",i);
   scanf("%d",&arr[i]);
  }
  for(i=0;i<max;i++)
  { for(j=0;j<max-i-1;j++)
    {
      { if(arr[j]>arr[j+1])
        { temp=arr[j];
          arr[j]=arr[j+1];
          arr[j+1]=temp;
        }
    }
  }
```

THE NEOTIA UNIVERSITY

```
}

printf("\n\tThe elements are:\n");
for(i=0;i<max;i++)
    printf("\t%d",arr[i]);
printf("\n\tEnter element to search:");
scanf("%d",&num);
lb=0;ub=max-1;mid=0,f=-1;
while(lb<=ub)
{ mid=(lb+ub)/2;
    if(arr[mid]<num)
        lb=mid+1;
    else
        if(arr[mid]>num)
            ub=mid-1;
        else
            {printf("\n\t%d is found in %dth position.",num,mid);
             getch();
             exit(0);
            }
    }
printf("\n\t%d is not found.",num);
}
```

ASSIGNMENT - 16

ASSIGNMENT NAME: Write a ‘C’ program to compute Matrix addition & multiplication.

OBJECTIVE: ‘C’ program to compute Matrix addition & multiplication.

ALGORITHM : Matrix Addition:

```
main()
{
    int a[3][3],b[3][3],c[3][3], i, j;
    printf ("\nEnter elements in first matrix:\n");
    for(i=0;i<3;i++)
        for(j=0;j<3;j++)
            {
                printf ("\nEnter element in a[%d][%d]:",i,j);
                scanf ("%d",&a[i][j]);
            }
    printf ("\nEnter elements in second matrix:\n");
    for(i=0;i<3;i++)
        for(j=0;j<3;j++)
            {
                printf ("\nEnter element in b[%d][%d]:",i,j);
                scanf ("%d",&b[i][j]);
            }
    for(i=0;i<3;i++)
        for(j=0;j<3;j++)
            c[i][j] = a[i][j] + b[i][j];
    printf ("\nMatrix addition result is:\n");
    for(i=0;i<3;i++)
        for(j=0;j<3;j++)
            printf ("%d ",c[i][j]);
}
```

THE NEOTIA UNIVERSITY

```
        }
    for(i=0;i<3;i++)
        for(j=0;j<3;j++)
            c[i][j]=a[i][j]+b[i][j];
    printf ("\nMatrix after addition :\n");
    for(i=0;i<3;i++)
    { for(j=0;j<3;j++)
        printf("%t%od", c[i][j]);
        printf("\n");
    }
Matrix Multiplication:
main()
{int a[3][3],b[3][3],c[3][3], i, j,k;
printf ("\nEnter elements in first matrix:\n");
for(i=0;i<3;i++)
    for(j=0;j<3;j++)
    { printf("\nEnter element in a[%od][%od]:",i,j);
        scanf("%od",&a[i][j]);
    }
printf ("\nEnter elements in second matrix:\n");
for(i=0;i<3;i++)
    for(j=0;j<3;j++)
    { printf("\nEnter element in b[%od][%od]:",i,j);
        scanf("%od",&b[i][j]);
    }
for(i=1;i<=n;i++)
{ for(j=1;j<=n;j++)
    {c[i][j]=0;
        for(k=1;k<=n;k++)
            c[i][j]+=a[i][k]*b[k][j];
    }
}
for(i=1;i<=n;i++)
{for(j=1;j<=n;j++)
    printf("%d\t",c[i][j]);
    printf("\n");
}
```

ASSIGNMENT – 17

ASSIGNMENT NAME: Write a C program to compute Fibonacci series using user defined function.

THE NEOTIA UNIVERSITY

OBJECTIVE: ‘C’ program to compute Fibonacci series using user defined function.

ALGORITHM:

```
int fibo(int n)
{
    if(n==0 || n==1)
        return n;
    else
        return(fibo(n-1)+fibo(n-2));
}
```

ASSIGNMENT - 18

ASSIGNMENT NAME: Write a ‘C’ program to compute Factorial of a given no. using recursion.

OBJECTIVE: ‘C’ program to compute Factorial of a given no. & Fibonacci series of a given no.

ALGORITHM :

```
int fact(int n)
{
    if(n==0)
        return 1;
    else
        return(n*fact(n-1));
}
```

ASSIGNMENT - 19

ASSIGNMENT NAME: Write a ‘C’ program to count no. of vowels, digits, consonants, spaces, words of a given string.

OBJECTIVE: ‘C’ program to count no. of vowels, digits, consonants, spaces, words of a given string.

ALGORITHM :

```
main()
{char str[100];
 int i,v=0,d=0,c=0,s=0,w=1;
```

THE NEOTIA UNIVERSITY

```
printf("\nEnter any string:");
fflush(stdin); gets(str);
for(i=0;str[i]!='\0';i++)
{if(isalpha(str[i]))
 {switch(tolower(str[i]))
  { case 'a':
  case 'e':
  case 'i':
  case 'o':
  case 'u': v++;break;
  default : c++;
  }
 }
 if(str[i]==' ' || str[i]!='\0')
 { s++;w++; }
 if(isdigit(str[i]))
 d++;
}
printf("\nThe no. of vowels is = %d",v);
printf("\nThe no. of consonents is = %d",c);
printf("\nThe no. of digits is = %d",d);
printf("\nThe no. of spaces is = %d",s);
printf("\nThe no. of words is = %d",w);
}
```

ASSIGNMENT – 20

ASSIGNMENT NAME: Write a ‘C’ program using the functions: strlen, strcpy, strcat & strcmp.

OBJECTIVE: ‘C’ program using the functions : strlen, strcpy, strcat & strcmp.

ALGORITHM :

```
main()
{char *s1,*s2,*s3,*s4,*s5;
 int l1,l2;
 printf("\nEnter 1st string::");gets(s1);
 printf("\nEnter 2nd string::");gets(s2);
 printf("\nEnter 3rd string::");gets(s3);
 l1=xstrlen(s1);l2=xstrlen(s2);
 printf("\nLength of 1st string is = %d",l1);
 printf("\nLength of 1st string is = %d",l2);
 strcpy(s4,s1);
 printf("\nCopy of 1st string is = ");puts(s4);
```

THE NEOTIA UNIVERSITY

```
xstrcpy(s5,s2);
printf("\nCopy of 2nd string is = ");puts(s5);
xstrcat(s1,s2);
printf("\nConcatenated 1st & 2nd string is = ");puts(s1);
if(xstrcmp(s1,s3)==0)
    printf("\nConcatenated(1st & 2nd) & 3rd Strings are
           equal.");
else
    printf("\nConcatenated(1st & 2nd) & 3rd Strings are
           equal.");
}
int xstrlen(char *s)
{int i;
 for(i=0;s[i]!='\0';i++);return i;
}
void xstrcpy(char *s2,char *s1)
{int i,j,l;
 l=xstrlen(s1);
 for(i=0,j=0;i<l;i++)
    s2[j++]=s1[i];
 s2[j]='\0';
}
void xstrcat(char *s1,char *s2)
{int i,j,l;
 l=xstrlen(s1);
 for(i=0;s2[i]!='\0';i++)
    s1[l++]=s2[i];
 s1[l]='\0';
}
int xstrcmp(char *s1,char *s2)
{int i,j,f=0;
 for(i=0;s1[i]!='\0';i++)
 {for(j=0;s2[j]!='\0';j++)
  {if(s1[i]!=s2[j])
   {f=-1;return f;}
  }
 }
 return f;
}
```

ASSIGNMENT – 21

ASSIGNMENT NAME: Write a ‘C’ program to determine whether a given string is palindrome or not.

THE NEOTIA UNIVERSITY

OBJECTIVE: ‘C’ program to determine whether a given string is Palindrome or not.

ALGORITHM :

```
main()
{
    char str[30];
    int l,i,j,f=1;
    printf("\nEnter any string:");
    gets(str);
    l=strlen(str);
    for(i=l-1,j=0;i>=0;i--,j++)
    {
        if(str[i]!=str[j])
        {
            f=0; break;
        }
        else
        {
            f=1;
            continue;
        }
    }
    if(f==0)
        printf("\nString is not palindrome.");
    else
        printf("\nString is palindrome.");
}
```

ASSIGNMENT – 22

ASSIGNMENT NAME: Write a ‘C’ program to store the name, roll no. & marks in 3 subjects of students using structure. Print the names of all students with average greater than 60%.

OBJECTIVE: ‘C’ program to store the name, roll no. & marks in 3 subjects of students using structure. Print the names of all students with average greater than 60%.

ALGORITHM :

```
struct student
{
    char name[30];
    int roll,marks[3],avg;
}
main()
{
    struct student s[100];
    int i,j,n,sum;
    printf("\nEnter how many student(s) u want to store:");
    scanf("%d",&n);
    for(i=0;i<n;i++)
    {
        printf("\n\t\tData of %dth student",i);
        // Add code to read name, roll, and marks for student i
    }
}
```

THE NEOTIA UNIVERSITY

```
printf("\n\t\t=====\n");
printf("\nEnter name:::",i);
fflush(stdin);
gets(s[i].name);
printf("\nEnter roll:::");
scanf("%d",&s[i].roll);
for(j=0;j<3;j++)
{ printf("\nEnter marks[%d] :::",j+1);
  scanf("%d",&s[i].marks[j]);
}
printf("\nThe Student who have got average greater
than 60% are:::\n"); s[i].avg=0;
for(i=0;i<n;i++)
{ sum=0;
  for(j=0;j<3;j++)
  { sum=sum+s[i].marks[j];
    s[i].avg=sum/3; }
  if(s[i].avg>60)
    puts(s[i].name);
}
}
```

ASSIGNMENT – 23

ASSIGNMENT NAME: Write a ‘C’ program to copy a disk file into another disk file using command line arguments.

OBJECTIVE: ‘C’ program to copy a disk file into another disk file using command line arguments.

ALGORITHM :

```
main(int argc,char * argv[])
{
  FILE *fs,*ft;
  char ch;
  if(argc!=3)
  {
    printf("\nImproper number of arguments");exit(0);
  }
  fs=fopen(argv[1],"r");
  if(fs==NULL)
  {
    printf("\nCannot open source file.");exit(0);
  }
```

THE NEOTIA UNIVERSITY

```
ft=fopen(argv[2],"w");
if(ft==NULL)
{
    printf("\nCannot open source file.");
    fclose(fs);exit(0);
}
while(1)
{
    ch=fgetc(fs);
    if(ch==EOF) break;
    fputc(ch,ft);
}
fclose(fs);fclose(ft);
}
```

ASSIGNMENT – 24

ASSIGNMENT NAME: Write a ‘C’ program to count the number of lines, words and characters in a given File.

OBJECTIVE: ‘C’ program to count the number of lines, words and characters in a given File.

ALGORITHM :

```
main()
{
    FILE *fp;
    int nol=0,now=0,noc=0;char ch;
    fp=fopen("ass23.cpp","r");
    if(fp==NULL)
    {printf("\nCannot open file");exit(0);}
    while(1)
    {
        ch=fgetc(fp);
        if(ch==EOF) break;
        if(ch=='\n') nol++;
        else if(ch==' ') now++;
        else noc++;
    }
    fclose(fp);
    printf("\nThe no.of line = %d",nol);
    printf("\nThe no.of word = %d",now);
    printf("\nThe no.of character = %d",noc);
}
```

THE NEOTIA UNIVERSITY

ASSIGNMENT – 25

ASSIGNMENT NAME: Write a ‘C’ program to display the frequency of characters in a given file.

OBJECTIVE: ‘C’ program to display the frequency of characters in a given file.

ALGORITHM :

```
main()
{
    int f[256],i;
    char c,sfile[20];
    FILE *fp;
    printf("\nEnter a file name:");
    scanf("%s",sfile);
    if((fp=fopen(sfile,"r"))==NULL)
    {
        printf("\nCan not open file %s",sfile);
        exit(0);
    }
    /* initializing count */
    for(i=0;i<256;i++)
        f[i]=0;
    while((c=getchar(fp))!=EOF)
    {
        ++f[c]; /* counter corresponding current
                   ASCII character is incremented */
    }
    /* printing result */
    for(i=0;i<256;i++)
    {
        if(f[i]>0)
            printf("\n%c occurs %d times.",i,f[i]);
    }
}
```

ASSIGNMENT – 26

ASSIGNMENT NAME: Write a ‘C’ program to swap of two variables using call by reference.

OBJECTIVE: ‘C’ program to swap of two variables using call by reference.

THE NEOTIA UNIVERSITY

ALGORITHM :

```
main()
{ int a,b;
  printf("\nEnter two integer variables:");
  scanf("%d %d",&a,&b);
  printf("\nThe values of two variables before swap:\n");
  printf(" %d %d",a,b);
  swap(&a,&b);
  printf("\nThe values of two variables after swap:\n");
  printf(" %d %d",a,b);
}
void swap(int *x,int *y)
{ int t;
  t=*x;*x=*y;*y=t;
}
```