

[illegible]

```
#include<stdio.h>
#define Why(a,b,c,d,e,f,g,h,i,j)a##b##d##c
#define Stucked Why(m,a,n,i,s,q,u,a,r,e)
```

[illegible]

3) WRITE A PROGRAM WHICH WILL CONTAIN 4 FUNCTIONS:

```
void A();  
void B();  
void C();  
void None();
```

USER WILL ENTER A CHARACTER FROM KEYBOARD. IF THE CHARACTER ENTERED IS EITHER A,B OR C THE O/P WILL BE "In function A"... else "Invalid choice"

THE PROGRAM MUST NOT CONTAIN IF-ELSE OR SWITCH-CASE.

ANS>>

```
#include<stdio.h>  
#include<string.h>  
void A()  
{  
    printf("In function A\n");  
}  
  
void B()  
{  
    printf("In function B\n");  
}  
  
void C()  
{  
    printf("In function C\n");  
}  
  
void None()  
{  
    printf("Invalid choice\n");  
}  
  
char Call[4][50]={"A","B","C","None"};
```

```
void (*fun_ptr[4])();
```

```
int main()
```

$$\{$$

```
int i=0;
```

```
char a[10];
```

```
fun_ptr[0]=&A;
```

```
fun_ptr[1]=&B;
```

```
fun_ptr[2]=&C;
```

```
fun_ptr[3]=&None;
```

```
printf("\nEnter a character in caps: ");
```

```
gets(a);
```

```
while((strcmp(a,Call[i]))!=0 && i<3)
```

$$\{$$
 $i++;$
$$\}$$

```
fun_ptr[i]();
```

```
return 0;
```

$$\}$$

EXPLANATION:- A simple application of function pointers.

[illegible]

ANS>>

[illegible]

```
??=include <stdio.h>
```

```
int main()
{
    printf("Hello World");
    return 0;
}
```

EXPLANATION:- Trigraphs!!

??= represents #

??/ represents \

??' represents \wedge

??(represents [

```
void flames(int);
```

```

void main()
{
    char a[SIZE],b[SIZE];
    int i,j,n,loa,lob,count=0;
    printf("Enter your name: ");
    gets(a);
    printf("Enter the other name: ");
    gets(b);
    loa=strlen(a);
    lob=strlen(b);
    n=loa+lob;
    for(i=0;i<loa;i++)
    {
        for(j=0;j<lob;j++)
        {
            if (toupper(a[i])==toupper(b[j]))
            {
                a[i]='*';
                b[j]='*';
                count=count+2;
                break;
            }
        }
    }
    n=n-count;
    p=(node)malloc(sizeof(struct linked));
    start->next=p;
    start->c='F';
    p->c='L';
    q=(node)malloc(sizeof(struct linked));
    p->next=q;
    q->c='A';
    p=(node)malloc(sizeof(struct linked));
    q->next=p;
    p->c='M';

```

```
q=(node)malloc(sizeof(struct linked));
p->next=q;
q->c='E';
p=(node)malloc(sizeof(struct linked));
q->next=p;
p->c='S';
p->next=start;
```

```
p=start;
for (j=1;j<6;j++)
{
    for (i=2;i<n;i++)
    {
        q=p->next;
        p=q;
    }
    m=p->next;
    p->next=m->next;
    freenode(m);
    p=p->next;
}
m=p->next;
p=m;
printf("Your relation: %c\n", m->c);
}
```

```
void freenode(node m)
{
    free(m);
}
```

7) WRITE A PROGRAM TO PRINT THE "PASCAL'S TRIANGLE".

THE NUMBER OF LINES TO BE PRINTED WILL BE TAKEN AS INPUT FROM KEYBOARD.

FOR e.g.

IF THE USER INPUTS: 5

O/P WILL BE:-

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

THE LOGIC USE SHOULD NOT BE OF 11^2 .

ANS>>

```
/*program to print pascal tringle of numbers*/
#include<stdio.h>
void main()
{
    int i,j,n,a[100][100];
    printf("Enter nth number of lines to be printed: ");
    scanf("%d", &n);
    for (i=0;i<=n;i++)
    {
        for (j=0;j<=((2*n)+1);j++)
        {
            if (j==n && i==0)
            {
                a[i][j]=1;
            }
            else
            {
                a[i][j]=0;
            }
        }
    }
}
```

```

        }
    }
}
for (i=1;i<=n;i++)
{
    for (j=1;j<(2*n);j++)
    {
        a[i][j]= a[i-1][j-1]+a[i-1][j+1];
    }
}
for (i=0;i<n;i++)
{
    for (j=1;j<(2*n);j++)
    {
        if (a[i][j]!=0)
        {
            printf("%2d",a[i][j]);
        }
        else
        {
            printf(" ");
        }
    }
    printf("\n");
}
}

```

8) WRITE A PROGRAM IN C WITHOUT USING GRAPHICS WHICH WILL DRAW THE GRAPH OF $y=x^2$. TAKE LEFT VERTICAL SIDE OF SCREEN AS POSITIVE x-axis AND

TOP HORIZONTAL SIDE OF SCREEN AS POSITIVE y-axis.

ANS>>

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

```
#include<math.h>
```

```
int f(int x)
```

```
{
```

```
    int y;
```

```
    y=(int)pow(x,2);
```

```
    return y;
```

```
}
```

```
int main()
```

```
{
```

```
    int i,x,j,y;
```

```
    printf("\nEnter value of x: ");
```

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

```
    for (i=0;i<5;i++)
```

```
    {
```

```
        y=f(i);
```

```
        for (j=0;j<y;j++)
```

```
        {
```

```
            printf(" ");
```

```
        }
```

```
        if (j==y)
```

```
        {
```

```
            printf("*\n");
```

```
        }
```

```
    }
```

```
    return 0;
```

```
}
```

9) WRITE A PROGRAM WHICH WILL ACCEPT TWO CHARACTERS OR NUMBERS AND CHECK FOR THEIR EQUALITY.NOTE THAT 0123 & 123 ARE EQUAL.[IGNORE FLOATING PARTS]

ANS>>

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

int main()
{
    char a[50],b[50];
    printf("\nEnter first character: ");
    gets(a);
    printf("\nEnter second character: ");
    gets(b);
    if (isdigit(*a) && isdigit(*b))
    {
        if(atoi(a)==atoi(b))
        {
            printf("\nThe numbers are equal\n");
        }
        else
        {
            printf("\nThe numbers are not equal\n");
        }
    }

    else if (isalpha(*a) && isalpha(*b))
    {
        if(!strcmp(a,b))
        {
            printf("\nStrings are equal\n");
        }
        else
```

```
        {  
            printf("\nThe strings are not equal\n");  
        }  
    }  
    else  
    {  
        printf("\n What you entered do not match with each other's format\n");  
    }  
    printf("\n");  
  
    return 0;  
}
```
