

```

#include <stdlib.h>
#include <conio.h>
#include <stdio.h>

int SortArray[100];

/* Rekursive Lösung*/
int rekfak(int x)
{
    if (!x) return(1); /* 0! ist gleich 1... */
    if (x<0) return(-1); /* normalerweise ein Fehler! */
    return (x*rekfak(x-1));
}

int rekfind(int what, int MIN, int MAX)
{
    int m;

    if (MIN > MAX) return(-1); /* Abbruch der Rekursion */

    m =(MIN + MAX) / 2;

    if (what < SortArray[m])
        return(rekfind(what, MIN, m-1));

    if (what > SortArray[m])
        return(rekfind(what, m+1, MAX));

    return(m);
}

/* Iterative Lösung */
int fak(int x)
{
    int hlp;

    if (!x || x==1) return(1); /* Hier brauchen wir nicht zu rechnen */
    if (x<0) return(-1); /* Fehler */

    for (hlp=x; x>0; x--) hlp*=x-1;
    return(hlp);
}

int find(int what, int MIN, int MAX)
{
    int m,p;

    p=-1;
    while (MIN<=MAX && p===-1) /* Abbruch der Schleife */
    {
        m =(MIN + MAX) / 2;
        if (what < SortArray[m])
            MAX=m-1;
        else
            if (what > SortArray[m])
                MIN=m+1;
            else
                p=m;
    }
    return(p);
}

void main()
{
    int i;

    /* Initialisieren */
    for (i=0; i<100; i++) SortArray[i]=i;
    /* Bildschirm löschen */
    clrscr();

    printf("Binäres suchen und Fakultätsberechnung!\n\nIterative Lösung:\n");

```

```

printf("Position von (-1:nicht gefunden):\n"
      "50:%d 0:%d 100:%d 99:%d -2:%d\n\n",
      find(50,0,99),find(0,0,99),find(100,0,99),find(99,0,99),
      find(-2,0,99));
printf("Fakultät von 5!:%d 0!:%d -1!y:%d\n",fak(5),fak(0),fak(-1));
printf("\nRekursive Lösung:\n");
printf("Position von (-1:nicht gefunden):\n"
      "50:%d 0:%d 100:%d 99:%d -2:%d\n\n",
      rekfind(50,0,99),rekfind(0,0,99),rekfind(100,0,99),rekfind(99,0,99),
      rekfind(-2,0,99));
printf("Fakultät von 5!:%d 0!:%d -1!:%d\n",rekfak(5),rekfak(0),rekfak(-1));

```

}

SUB