

PSPC PRACTICALS - 2

Program to find greatest and smallest number in an array

```
#include<stdio.h>

int main()
{
    int arr[50],i,n,great,small;
    printf("\nEnter the number of elements : ");
    scanf("%d",&n);
    printf("\nInput the array elements : ");
    for(i=0;i<n;++i)
        scanf("%d",&arr[i]);

    great=small=arr[0];

    for(i=1;i<n;++i)
    {
        if(arr[i]>great)
            great=arr[i];

        if(arr[i]<small)
            small=arr[i];
    }

    printf("The smallest element is %d\n",small);
    printf("The largest element is %d\n",great);

    return 0;
}
```

```
Enter the number of elements : 5

Input the array elements : 11
33
55
99
100
The smallest element is 11
The largest element is 100

...Program finished with exit code 0
Press ENTER to exit console.
```

Program to swap two numbers using call by value and call by reference

Call by value:

```
#include <stdio.h>
void call(int a,int b);
int main()
{
    int a,b;
    printf("Enter the two numbers:\n");
    scanf("%d%d",&a,&b);
    printf("The values of a is %d and b is %d \n",a,b);
```

```
    call(a,b);

    return 0;
}
void call(int a ,int b)
{
    int num;
    num=a;
    a=b;
    b=num;
    printf("The swapped values are a=%d and b=%d",a,b);
}
```

```
Enter the two numbers:
10
20
The values of a is 10 and b is 20
The swapped values are a=20 and b=10

...Program finished with exit code 0
Press ENTER to exit console.
```

Call by reference:

```

#include <stdio.h>
void call(int *a,int *b);
int main()
{
    int a,b;
    printf("Enter the two numbers:\n");
    scanf("%d%d",&a,&b);
    printf("The values of a is %d and b is %d \n",a,b);
    call(&a,&b);

    return 0;
}
void call(int *a ,int *b)
{
    int *num;
    num=a;
    a=b;
    b=num;
    printf("The swapped values are a=%d and b=%d",*a,*b);
}

```

```

Enter the two numbers:
11
22
The values of a is 11 and b is 22
The swapped values are a=22 and b=11

...Program finished with exit code 0
Press ENTER to exit console.

```

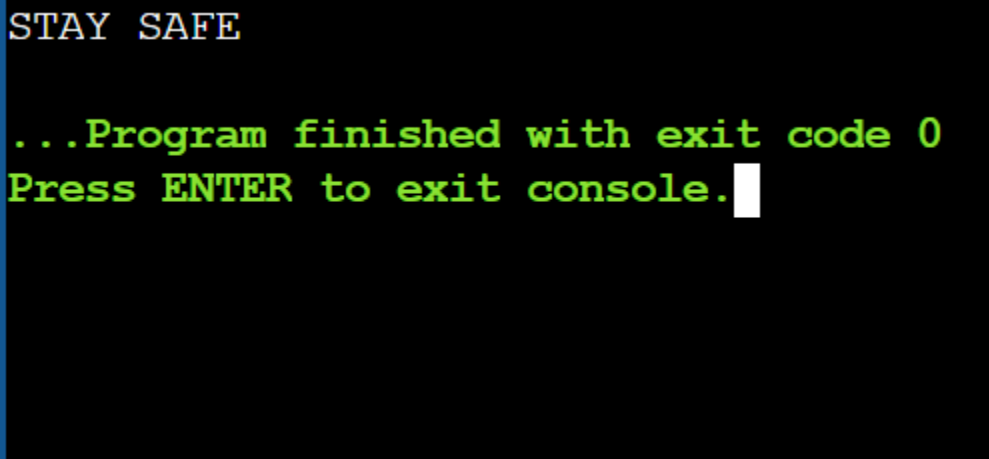
FUNCTIONS

Function without arguments and no return type

```
#include<stdio.h>

void main()
{
    void display();
    display();
    getch();
}

void display()
{
    printf("STAY SAFE");
}
```



STAY SAFE

...Program finished with exit code 0
Press ENTER to exit console.

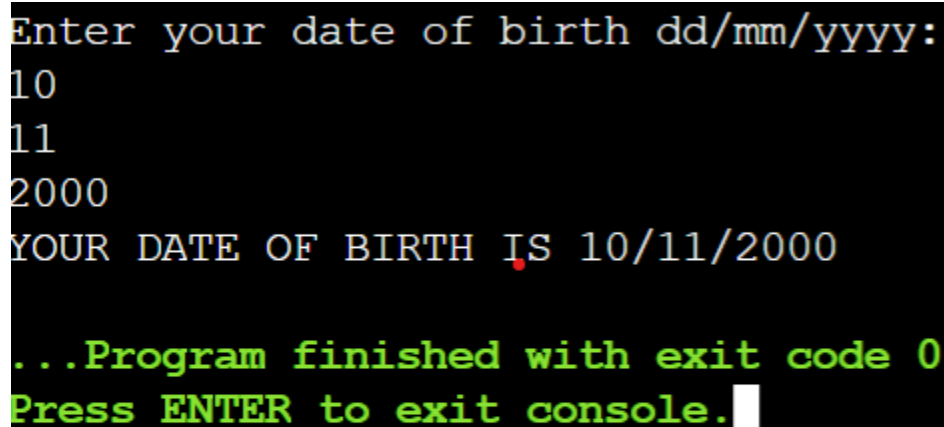
Function with arguments and without return value

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

```

int dob(int,int,int);
int d,m,y;
printf("Enter your date of birth dd/mm/yyyy: \n");
scanf("%d%d%d",&d,&m,&y);
dob(d,m,y);
getch();
}
dob(int a,int b,int c)
{
    printf("YOUR DATE OF BIRTH IS %d/%d/%d",a,b,c);
}

```



The screenshot shows a terminal window with a black background. The text is displayed in a monospaced font. The first line is a prompt "Enter your date of birth dd/mm/yyyy:". The user has entered "10", "11", and "2000" on separate lines. The next line shows the output "YOUR DATE OF BIRTH IS 10/11/2000". Below this, there is a green message "...Program finished with exit code 0" and another green message "Press ENTER to exit console." followed by a white cursor block.

Functions with arguments and with return values

```

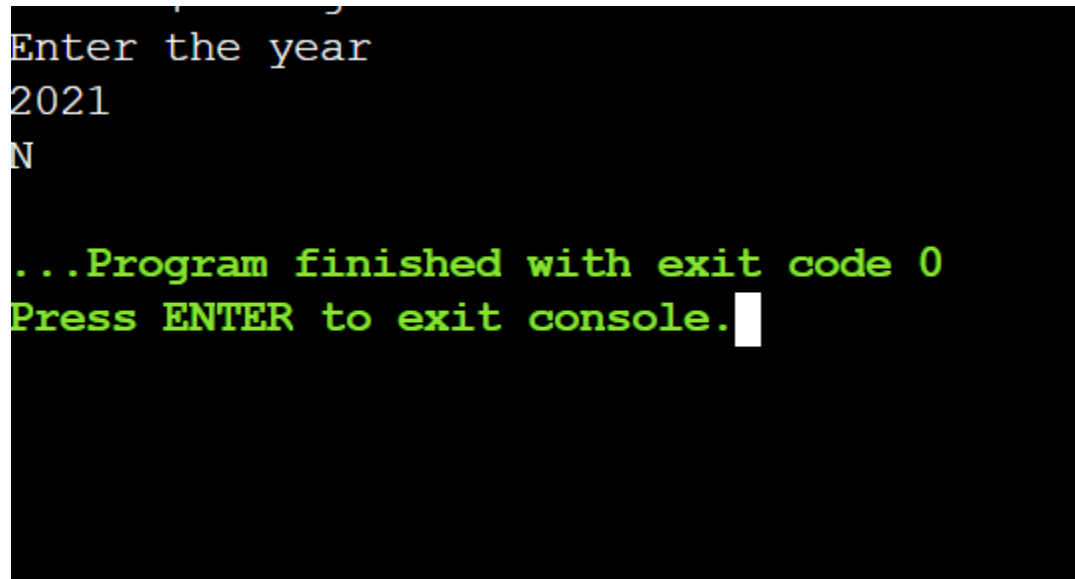
#include<stdio.h>
char leapyear(int);
int main()
{
    int year;
    char check;
    printf("Enter the year\n");
    scanf("%d",&year);
    check=leapyear(year);
}

```

```

    printf("%c",check);
    getch();
}
char leapyear(int givenyear)
{
    if(givenyear%4==0)
        return('S');
    else
        return('N');
}

```



```

Enter the year
2021
N

...Program finished with exit code 0
Press ENTER to exit console.

```

Function without arguments but with return values

```

#include<stdio.h>
int cal(int);

int main()
{
    int s,area;
    printf("Enter the side of a square:\n");
    scanf("%d",&s);

```

```
    area=cal(s);  
    printf("The area of the square is %d",area);  
    return 0;  
}  
int cal(int s)  
{  
    int area;  
    area=(s*s);  
    return area;  
}
```

Enter the side of a square:

11

The area of the square is 121

...Program finished with exit code 0

Press ENTER to exit console.