

NAME : YANA SRIVASTAVA

SECTION : 23

GROUP : “B”

UID : 20BCS2279

SUBJECT : PROGRAMMING

TOPIC :

A function is provided with zero or more arguments, and it executes the statements on it. Based on the return type, it either returns nothing (void) or something. Develop a program to find greatest of four numbers using function `int max_of_four(int a, int b, int c, int d)` which reads four arguments and returns the greatest of them.

AIM :

Learn how to perform operations using functions in C.

ALGORITHM / FLOWCHART :

- (i) Start the program.
- (ii) Declaration of function `max` which returns an integer
- (iii) Check the greatest among four using ternary operator.
- (iv) Calling the function.
- (v) Declaration of variables in integer datatype.
- (vi) Print the message “Enter any 4 numbers”.

- (vii) Accept the input of numbers from the user.
- (viii) Print the message and the largest number.
- (ix) End the program by returning an integer value.

CODE :

```
//creating a header file
#include<stdio.h>

//function which returns an integer value
int max_of_four(int a, int b, int c, int d)
{
//check the greatest among four using ternary operator
return((a>b&&a>c&&a>d)?( a):(b>c && b>d)?(
b):(c>d)?(c):(d));
}

//calling the function
int main()
{
//declaration of variables in integer datatype
int a,b,c,d;

//print the message
printf("Enter any 4 numbers :\n");

//accept the input of numbers from the user
scanf("%d%d%d%d",&a,&b,&c,&d);
```

```
//print the message and the largest number
printf("Largest out of 4 no is : %d", max_of_four(a,b,c,d));
//return an integer value
return 0;
}
```

ERRORS ENCOUNTERED DURING PROGRAM'S EXECUTION :

No Error.

PROGRAM'S EXPLANATION (In Brief) :

In this program we have to find the greatest of four numbers using function `int max_of_four(int a, int b, int c, int d)` which reads four arguments and returns the greatest of them by using ternary operator.

OUTPUT :

```
Enter any 4 numbers :
10
20
30
40
Largest out of 4 no is : 40
```

TOPIC :

Write a recursive function for computing factorial of a number.
Write main to test its functioning.

AIM :

Learn how to perform operations using functions in C.

FLOWCHARTS / ALGORITHM :

- (i) Start the program.
- (ii) Declaration of function int factorial to return an integer.
- (iii) Declaration of function main to find the factorial of a number.
- (iv) Declaration of variables in integer datatype.
- (v) Print the message
- (vi) Accept the input of number from the user.
- (vii) Check the condition to find factorial of a number using if else statement and print the message according to it.
- (viii) Return 1 in case of number is 0 as factorial of 0 is 1 otherwise call the function itself and print the message according to it.
- (ix) End the program.

CODE :

```
//creating a header file for standard input output operations
#include <stdio.h>

//function to return an integer
```

```
int factorial(int);  
  
//function to find the factorial of a number  
  
int main()  
{  
//declaration of variables in integer datatype  
int n;  
//print the message  
printf("Enter no to find factorial : ");  
//accept the input of number from the user  
scanf("%d", &n);  
  
/*check the condition to find factorial of a number using if else  
statement and print the message according to it*/  
if (n >= 0)  
printf("Factorial of given no %d is : %d", n, factorial(n));  
else  
printf("Not Possible");  
//return an integer value  
return 0;  
}  
  
//function which returns an integer value  
int factorial(int n)
```

```
{  
/*return 1 in case of number is 0 as factorial of 0 is 1 otherwise  
call the function itself and print the message according to it*/  
if(n==0)  
return 1;  
else  
return n*factorial(n-1);  
}
```

ERRORS ENCOUNTERED DURING PROGRAM'S EXECUTION :

No Error.

PROGRAM'S EXPLANATION (In Brief) :

In this program we have to find the factorial of a number using recursive function and if else statement.

OUTPUT :

```
Enter no to find factorial : 10  
Factorial of given no 10 is : 3628800
```

TOPIC :

Write a program to create functions for following

- (i) Input details of employee using input_data() (name, employee id, number of working days, date of joining, initial salary, contact number, designation, department)
- (ii) Calculate monthly salary using calc_salary()
- (iii) Display the monthly salary with deductions if any using display()

AIM :

Learn how to perform operations using functions in C.

FLOWCHARTS / ALGORITHM :

- (i) Start the program.
- (ii) Store the name in string with length 50.
- (iii) Declaration of variables in integer datatype.
- (iv) Store date of joining in string with length 10
- (v) Declaration of variables in integer datatype.
- (vi) Declaration of variables in long long datatype.
- (vii) Store designation in string with length 15
- (viii) Store department in string with length 20
- (ix) Declaration of variables in integer datatype.
- (x) Functions declaration to input data, calculate salary and display
- (xi) Calling the function and return an integer value.

- (xii) Print the message and accept the input from the user.
- (xiii) Call the function to input data, calculate monthly salary and for display.
- (xiv) Calculate the salary by using the formula: $\text{salary} / 30 * \text{no.of work days}$
- (xv) Calculate the monthly salary after deductions using the formula: $\text{monthly salary} - \text{deducted amount}$
- (xvi) End the program.

CODE :

```
//creating a header file for standard input output operations
#include <stdio.h>

//name in string with length 50
char name[50];

//declaration of variables in integer datatype
int emp_id;
int num_work_days;

//date of joining in string with length 10
char doj[10];

//declaration of variables in integer datatype
int salary;

//declaration of variables in long long datatype
```



```
long long contact_num;

//designation in string with length 15
char designation[15];

//department in string with length 20
char deptt[20];

//declaration of variables in integer datatype
int mon_salary;

/*Functions declaration to input data, calculate salary and
display*/

void input_data();

void calc_salary();

void display();

//calling the function

int main()

{

input_data();

calc_salary();

display();

//return an integer value
```

```
return 0;

}

//function to input the data

void input_data()

{

//print the message

printf("Enter name of the Employee:");

//accept the input of name from the user

scanf("%s", name);

//print the message

printf("Enter emp_id of the Employee:");

//accept the input of employee id from the user

scanf("%d", &emp_id);

//print the message

printf("Enter number of working days:");

//accept the input of number of working days

scanf("%d", &num_work_days);

//print the message

printf("Enter date of joining:");
```

```
//accept the input of date of joining from the user
scanf("%s", doj);

//print the message
printf("Enter salary:");

//accept the input of salary from the user
scanf("%d", &salary);

//print the message
printf("Enter contact_num:");

//accept the input of contact number from the user
scanf("%lld", &contact_num);

//print the message
printf("Enter designation:");

//accept the input of designation from the user
scanf("%s", designation);

//print the message
printf("Enter deptt:");

//accept the input of departement from the user
scanf("%s", deptt);

//print the message and name
```

```
printf("\nName is:%s",name);

//print the message and employee id

printf("\nEmployee ID is:%d", emp_id);

//print the message and number of working days

printf("\nNumber of working days are:%d", num_work_days);

//print the message and date of joining

printf("\nDate of joining is:%s", doj);

//print the message and salary

printf("\nSalary is:%d", salary);

//print the message and contact number

printf("\ncontact_num is:%lld", contact_num);

//print the message and designation

printf("\ndesignation is:%s", designation);

//print the message and departement

printf("\ndeptt is:%s", deptt);

}

//function to calculate the salary

void calc_salary()
```

```
{  
  
/*calculate the salary by using the formula: salary / 30* no.of  
work days*/  
  
mon_salary=salary/30*num_work_days;  
  
}  
  
//function to display the data  
  
void display()  
  
{  
  
//declaration of variables in integer datatype  
  
int deduct;  
  
//print the message and name and monthly salary  
  
printf("\n\nMonthly salary of %s is:%d",name,mon_salary);  
  
//print the message  
  
printf("\n\nEnter the deductions of salary if any(in Rs):");  
  
//accept the input from the user  
  
scanf("%d",&deduct);  
  
/*calculate the monthly salary after deductions using the  
formula: monthly salary - deducted amount*/  
  
mon_salary= mon_salary - deduct;
```

```
//print the message and name and monthly salary after deduction  
printf("\nMonthly salary of %s is:%d",name,mon_salary);  
}
```

ERRORS ENCOUNTERED DURING PROGRAM'S EXECUTION :

No Error.

PROGRAM'S EXPLANATION (In Brief) :

In this program we have to input the details of employee using input_data() (name, employee id, number of working days, date of joining, initial salary, contact number, designation, department) and then Calculate monthly salary using calc_salary() and Display the monthly salary with deductions if any using display().

OUTPUT :

```
Enter name of the Employee:RAJ
Enter emp_id of the Employee:3607
Enter number of working days:20
Enter date of joining:04april2020
Enter salary:6000
Enter contact_num:8579034126
Enter designation:EP
Enter deptt:CSE

Name is:RAJ
Employee ID is:3607
Number of working days are:20
Date of joining is:04april2020
Salary is:6000
contact_num is:8579034126
designation is:EP
deptt is:CSE

Monthly salary of RAJ is:4000

Enter the deductions of salary if any(in Rs):40
Monthly salary of RAJ is:3960
```

TOPIC :

Store age of all students of your class in an array. Pass this array as an argument and find average height of the class and return it to calling function.

AIM :

Learn how to perform operations using functions in C.

FLOWCHART / ALGORITHM :

- (i) Start the program.
- (ii) Declaration of function to return an integer value.
- (iii) Declaration of variables in float and integer data type.
- (iv) Loop to enter the number of students.
- (v) Print the number of students and accept the input of height and age from the user.
- (vi) Calculate the average of height : $\text{sum} \backslash \text{no. of students}$,
- (vii) End the program.

CODE :

```
#include<stdio.h>

float avg_ht(int age[],int n);

int main()

{

int age[30],n,i;

float ah;

printf("Enter no. of students in the class:");

scanf("%d",&n);

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

{

printf("\nEnter age of student %d:",i+1);

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

}

ah=avg_ht(age,n);

printf("Average height of class is:%f",ah);

return 0;

}

float avg_ht(int age[],int n)

{
```



```
int h[n],i;
float avg,sum=0.0;
for(i=0;i<n;i++)
{
if(age[i]==16)
h[i]=168;
else if(age[i]==17)
h[i]=170;
else if(age[i]==18)
h[i]=172;
else if(age[i]==19)
h[i]=175;
sum=sum+h[i];
avg=sum/n;
}
return avg;
}
```

ERRORS ENCOUNTERED DURING PROGRAM'S EXECUTION :

No Error.

PROGRAM'S EXPLANATION :

In this program we have to Store age of all students of your class in an array. Pass this array as an argument and find average height of the class and return it to calling function.

OUTPUT :

```
Enter no. of students in the class:3
Enter age of student 1:160
Enter age of student 2:165
Enter age of student 3:171
Average height of class is:-27009470.000000

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

