PROGRAM USING STATEMENTS , EXPRESSIONS, DECISION MAKING, ITERATIVE STATEMENTS

PROGRAM:

```
#include<stdio.h>
void main()
{
char name[20]= "Abinaya.m";
char address[80]= "Tiruvannamalai";
int date=20;
int month=10;
int year=1990;
int mobile=987456321;
int age=25;
printf("\n====================");
printf("\n NAME: %s",name);
printf("\n ADDRESS:%s", address);
printf("\n DOB:%d:%d:%d", date , month, year);
printf("\n MOBILE NUMBER:%d", mobile);
printf("\n AGE:%d", age);
printf("\n====================");
}
```

OUTPUT :

NAME: ABINAYA.M

ADDRESS:Tiruvannamalai
DOB:20:10:2004

MOBILE NUMBER:987456321

AGE:19

PROGRAM TO GET THE USER DETAILS AND DISPLAY IT

PROGRAM :

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
int main()
{
char name[20];
char address[80];
int date;
int month;
int year;
long int mobile;
char gender[20];
int age;
printf("\n ENTER YOUR NAME:=");
gets(name);
printf("\nENTER YOUR ADDRESS=");
gets(address);
printf("\nENTER YOUR date/month/year=");
scanf("%d/%d/%d",&date,&month,&year);
printf("\n ENTER YOUR AGE=");
scanf("%d",&age);
printf("\n ENTER YOUR GENDER(MALE/FEMALE)=");
scanf("%s",gender);
printf("\nENTER YOUR MOBILE NUMBER=");
scanf("%ld" ,&mobile);
printf("\n====================");
```

```
printf("\n NAME: %s",name);
printf("\n ADDRESS:%s", address);
printf("\n DOB:%d:%d:%d", date , month, year);
printf("\n AGE:%d", age);
printf("\n GENDER:%s", gender);
printf("\n MOBILE NUMBER:%d", mobile);
printf("\n====================");
return 0;
}
```

OUTPUT :

ENTER YOUR NAME:=ABINAYA.M

ENTER YOUR ADDRESS=Tiruvannamalai

ENTER YOUR date/month/year=03/12/2004

ENTER YOUR AGE=19

ENTER YOUR GENDER(MALE/FEMALE)=FEMALE

ENTER YOUR MOBILE NUMBER=987654321

==========

NAME: ABINAYA.M

ADDRESS:Tiruvannamalai.

DOB:3:12:2004

AGE:19

GENDER:FEMALE

MOBILE NUMBER:987654321

==========

PROGRAM TO CHECK WHETHER A GIVEN NUMBER IS ODD OR EVEN

PROGRAM:

```
#include <stdio.h>
int main()
{
int number;
printf("Enter an integer: ");
scanf("%d", &number);
// True if the number is perfectly divisible by 2
if(number % 2 == 0)
printf("%d is even.", number);
else
printf("%d is odd.", number);
return 0;
}
```

OUTPUT:

Enter an integer: -7

-7 is odd.

Enter an integer : 8

8 is even.

PROGRAM TO CHECK WHETHER A GIVEN NUMBER IS ARMSTRONG NUMBER OR NOT?

PROGRAM:

```
#include<stdio.h>
int main()
{
int num,copy_of_num,sum=0,rem;
printf("\nEnter a number:");
scanf("%d",&num);
copy_of_num=num;
while (num != 0)
{
rem = num % 10;
sum = sum + (rem*rem*rem);
num = num / 10;
}
if(copy_of_num == sum)
printf("\n%d is an Armstrong Number",copy_of_num);
else
printf("\n%d is not an Armstrong Number",copy_of_num);
return(0);
}
```

OUTPUT:

Enter a number: 370

370 is an Armstrong Number

PROGRAM: PROGRAM TO PRINT TABLE FOR THE GIVEN NUMBER USING DO WHILE LOOP

```
#include<stdio.h>
int main()
{
int i=1,number=0;
printf("Enter a number: ");
scanf("%d",&number);
do{
printf("%d \n",(number*i));
i++;
}while(i<=10);
return 0;
}
```

OUTPUT:

Enter a number:

5

5

10

15

20

25

30

35

40

45

50

PROGRAM: PROGRAM TO FIND WHETHER THE GIVEN YEAR IS LEAP

```
#include <stdio.h>
#include <conio.h>
void main()
{
int year;
printf("Enter a year :\n");
scanf("%d", &year);
if ((year % 400) == 0)
printf("%d is a leap year \n",year);
else
if ((year % 100) != 0 && (year % 4) == 0)
printf("%d is a leap year \n",year);
else
printf("%d is not a leap year \n",year);
}
```

OUTPUT:

Enter a year:

2000

2000 is a leap year

Enter a year:

1900

1900 is not a leap year

PROGRAM TO DISPLAY BIGGEST OF THREE NUMBERS

PROGRAM:

```
#include <stdio.h>
void main()
{
int A,B,C;
printf("Enter 3 integer number \n");
scanf("%d",&A);
scanf("%d",&B);
scanf("%d",&C);
if(A>B){
if(A>C){
printf(" %d is the Greatest Number \n",A);
}
else{
printf("%d is the greatest Number \n",C);
}
}
else{
if(B>C){
printf("%d is the greatest Number \n",B );
}
else{
printf("%d is the greatest Number \n", C);
}
}
}
```

OUTPUT:

Enter three numbers: -4.5

3.9

5.6

5.60 is the largest number.

PROGRAM TO CHECK WHETHER THE ENTERED CHARACTER IS VOWEL OR NOT(USE SWITCH CASE)

PROGRAM:

```
#include<stdio.h>
#include<conio.h>
int main()
{
char ch;
printf("Enter a character: ");
scanf("%c",&ch);
//condition to check character is alphabet or not
if((ch>='A' && ch<='Z') || (ch>='a' && ch<='z'))
{
switch(ch)
{
case 'A':
case 'E':
case 'I':
case 'O':
case 'U':
case 'a':
case 'e':
case 'i':
case 'o':
case 'u':
printf("%c is a VOWEL.\n",ch);
break;
default:
printf("%c is a CONSONANT.\n",ch);
}
```

```
}
else
{
printf("%c is not an alphabet.\n",ch);
}
return 0;
}
```

OUTPUT:

Enter a character E

E is a vowel

Enter a character

X

X is a consonant Enter a character

+

+ is not an alphabet

PROGRAM: PROGRAM TO FIND FACTORIAL OF A GIVEN NUMBER

```
#include<stdio.h>
int main()
{
int n, i; longfactorial = 1;
printf("Enter an integer: ");
scanf("%d",&n);
// show error if the user enters a negative integer
if (n < 0)
printf("Error! Factorial of a negative number doesn't exist.");
else
{
for(i=1; i<=n; ++i)
{
factorial *= i;
// factorial = factorial*i;
}
printf("Factorial of %d = %lu", n, factorial);
}
return 0;
}
```

OUTPUT:

Enter an integer: 10

Factorial of 10 = 3628800

PROGRAM TO FIND OUT THE AVERAGE OF 4 INTEGERS

PROGRAM:

```
#include<stdio.h>
void main()
{
inti,n,sum=0,nu[100];
float avg;
clrscr();
printf("\nEnter the numbers\n");
for(i=0;i<3;i++)
{
scanf("%d",&nu[i]);
sum = sum + nu[i];
}
avg = (float)sum/n;
printf("\nAverage is :
%.2f\n",n,avg); getch();
}
```

OUTPUT:

Enter the numbers 32

45

54

22

Average is 38.25

PROGRAM TO PERFORM THE CALCULATOR OPERATIONS, NAMELY, ADDITION, SUBTRACTION, MULTIPLICATION, DIVISION AND SQUARE OF ANUMBER

PROGRAM:

```
#include<stdio.h>
// functions declaration
int add(int n1, int n2);
int subtract(int n1, int n2);
int multiply(int n1, int n2);
int divide(int n1, int n2);
int square(int n1);
// main function
int main()
{
int num1, num2;
printf("Enter two numbers: ");
scanf("%d %d", &num1, &num2);
printf("%d + %d = %d\n", num1, num2, add(num1, num2));
printf("%d - %d = %d\n", num1, num2, subtract(num1, num2));
printf("%d * %d = %d\n", num1, num2, multiply(num1, num2));
printf("%d / %d = %d\n", num1, num2, divide(num1, num2));
printf("%d^%d=%d\n",num1,num1,square( num1));
return 0;
}
// function to add two integer numbers
int add(int n1, int n2)
{
int result;
result = n1 + n2;
return result;
}
```

```
// function to subtract two integer numbers
int subtract(int n1, int n2)
{
int result;
result = n1 - n2;
return result;
}
// function to multiply two integer numbers
int multiply(int n1, int n2)
{
int result;
result = n1 * n2;
return result;
}
// function to divide two integer numbers
int divide(int n1, int n2)
{
int result;
result = n1 / n2;
return result;
}
// function to find square of a number
int square(int n1)
{
int result;
result = n1*n1;
return result;
}
```

OUTPUT:

Enter two numbers:2

3

2+3=5

2-3=-1

2*3=6

2/3=0

2^2=4

PROGRAM TO PERFORM SWAPPING USING FUNCTIONS

PROGRAM:

```
#include<stdio.h>
void main()
{
void swap(int,int);
inta,b,r;
clrscr();
printf("enter value for a&b: ");
scanf("%d%d",&a,&b);
swap(a,b);
getch();
}
void swap(inta,int b)
int temp;
temp=a;
a=b;
b=temp;
printf("after swapping the value for a & b is : %d %d",a,b);
}
```

OUTPUT:

Enter values for a&b:23

32

After swapping values of a&b is : 32 23

PROGRAM TO DISPLAY ALL PRIME NUMBERS BETWEEN TWO INTERVALS USING FUNCTIONS

PROGRAM:

```
#include <stdio.h>
/* Function declarations */
int isPrime(int num);
void printPrimes(int lowerLimit, int upperLimit);
int main()
{
int lowerLimit, upperLimit;
printf("Enter the lower and upper limit to list primes: ");
scanf("%d%d", &lowerLimit, &upperLimit);
/* Call function to print all primes between the given range*/
printPrimes(lowerLimit, upperLimit);
return 0;
}
/* Print all prime numbers between lower limit and upper limit*/
void printPrimes(int lowerLimit, int upperLimit)
{
printf("All prime number between %d to %d are: ", lowerLimit, upperLimit);
while(lowerLimit <= upperLimit)
{
/* Print if current number is prime*/
if(isPrime(lowerLimit))
{
printf("%d, ", lowerLimit);
}
lowerLimit++;
}
}
```

```
/*Check whether a number is prime or not*/
/*Returns 1 if the number is prime otherwise 0*/
int isPrime(int num)
{
int i;
for(i=2; i<=num/2; i++)
{
/* If the number is divisible by any number*/
/*other than 1 and self then it is not prime*/
if(num % i == 0)
{
return 0;
}
}
return 1;
}
```

OUTPUT:

Enter lower and upper limit to list :0

50

All prime number between 0 to 50 are 5791113151719212325272931333353739414345474 9

PROGRAM TO DISPLAY ARRAY ELEMENTS USING 2D ARRAYS

PROGRAM:

```
#include<stdio.h>
int main(){
/* 2D array declaration*/
int disp[2][3];
/*Counter variables for the loop*/
int i, j;
for(i=0; i<2; i++) {
for(j=0;j<3;j++) {
printf("Enter value for disp[%d][%d]:", i, j);
scanf("%d", &disp[i][j]);
}
}
//Displaying array elements
printf("Two Dimensional array elements:\n");
for(i=0; i<2; i++) {
for(j=0;j<3;j++) {
printf("%d ", disp[i][j]);
if(j==2){
printf("\n");
}
}
}
return 0;
}
```

OUTPUT:

Enter value for disp[0][0]:22

Enter value for disp[0][1]:33

Enter value for disp[0][2]:44

Enter value for disp[1][0]:55

Enter value for disp[1][1]:66

Enter value for disp[1][2]:7

PROGRAM TO GET THE LARGEST ELEMENT OF AN ARRAY USING FUNCTION

PROGRAM:

```
#include <stdio.h>
#include <conio.h>
max(int  [],int);
void main()
{
int a[]={10,5,45,12,19};
int n=5,m;
clrscr();
m=max(a,n);
printf("\nmaximum number is %d",m);
getch();
}
max(int x[],int k)
{
int t,i;
t=x[0];
for(i=1;i<k;i++)
{
if(x[i]>t)
t=x[i];
}
return(t);
}
```

OUTPUT:

Maximum number is 45

PROGRAM TO STORE STUDENT INFORMATION IN STRUCTURE AND DISPLAY IT

PROGRAM:

```
#include<stdio.h>
struct student
{
int roll_no, mark1, mark2, mark3, total;
float average;
char name[10],grade;
};
void struct_funct_student(struct student stu);
int main()
{
struct student stud;
printf("\nRoll No.=");
scanf("%d",&stud.roll_no);
printf("Name=");
scanf("%s",stud.name);
printf("Mark1=");
scanf("%d",&stud.mark1);
printf("Mark2=");
scanf("%d",&stud.mark2);
printf("Mark3=");
scanf("%d",&stud.mark3);
struct_funct_student(stud);
return 0;
}
void struct_funct_student( struct student stu)
{
stu.total = stu.mark1 + stu.mark2 + stu.mark3;
stu.average = stu.total / 3;
```

```
if(stu.average >= 90)
stu.grade='S';
else if(stu.average >= 80)
stu.grade='A';
else if(stu.average >= 70)
stu.grade='B';
else if(stu.average >= 60)
stu.grade='C';
else if(stu.average >= 50)
stu.grade='D';
else
stu.grade='F';
printf("\n ROLL NO. \t NAME \t TOTAL \t AVG \t GRADE \n");
printf("%d \t %s \t %d \t %f \t %c", stu.roll_no,stu.name,stu.total,stu.average,stu.grade);
}
```

OUTPUT:

Roll no:02

Name:ARUNKARTHIK

Mark1:88

Mark2:99

Mark3:88

Roll no	Name	Total	Avg	Grade
02	ARUNKARTHIK	275	91.000000	S

PROGRAM TO READ THE STUDENT DATA AND CALCULATE THETOTAL MARKS

PROGRAM:

```
#include<stdio.h>
struct student
{
int sub1;
int sub2;
int sub3;
int sub4;
int sub5;
};
void main()
{
struct student s[10];
int i,total=0;
clrscr();
for(i=0;i<=9;i++)
{
printf("\nEnter Marks in Five Subjects = ");
scanf("%d%d%d",& s[i].sub1,&s[i].sub2,&s[i].sub3,&s[i].sub4,&s[i].sub5);
total=s[i].sub1+s[i].sub2+s[i].sub3+s[i].sub4+s[i].sub5;
printf("\nTotal marks of s[%d] Student= %d",i,total);
}
getch();
}
```

OUTPUT:

```
Enter marks in 5 subjects :88
77
66
88
99
Total mark of s[0] student :418
Enter marks in 5 subjects :77
77
66
88
99
Total mark of s[1] student :408
Enter marks in 5 subjects :66
77
66
88
99
Total mark of s[2] student :398
Enter marks in 5 subjects :88
77
66
88
99
Total mark of s[3] student :418
Enter marks in 5 subjects :88
77
66
88
99
```

Total mark of s[4] student :418

Enter marks in 5 subjects :50

77

66

88

99

Total mark of s[5] student :380

Enter marks in 5 subjects :88

77

66

88

99

Total mark of s[6] student :418

Enter marks in 5 subjects :88

77

66

88

99

Total mark of s[7] student :418

Enter marks in 5 subjects :88

77

66

88

99

Total mark of s[8] student :418

PROGRAM TO GENERATE SALARY SLIP OF EMPLOYEES USING STRUCTURES AND POINTERS

PROGRAM:

```
#include<stdio.h>
#include<conio.h>
#include <stdlib.h>
struct emp
{
int empno ;
char name[10];
int bpay, allow, ded, npay ;
struct emp *next;
} ;
void main()
{
int i,n=0;
EX:No:3c PROGRAM TO GENERATE SALARY SLIP OF EMPLOYEES USING
STRUCTURES AND POINTERS
char answer;
int more_data = 1;
struct emp *current_ptr, *head_ptr;
clrscr() ;
head_ptr = (struct emp *)malloc (sizeof(struct emp));
current_ptr = head_ptr;
while (more_data)
{
printf("\nEnter the employee number : ") ;
scanf("%d", & current_ptr->empno) ;
printf("\nEnter the name : ") ;
scanf("%s",& current_ptr->name) ;
```

```
printf("\nEnter the basic pay, allowances & deductions : ");
scanf("%d %d %d", & current_ptr ->bpay, & current_ptr ->allow,& current_ptr ->ded) ;
e[i].npay = e[i].bpay + e[i].allow - e[i].ded ; n++;
printf("Would you like to add another employee? (y/n): ");
scanf("%s", answer);
if (answer!= 'Y')
{
current_ptr->next = (struct eme *) NULL;
more_data = 0;
}
else
{
current_ptr->next = (struct emp *) malloc
(sizeof(struct emp));
current_ptr = current_ptr->next;
}
}
printf("\nEmp. No. Name \t Bpay \t Allow \t Ded \t Npay\n\n") ;
current_ptr = head_ptr;
for(i = 0 ; i < n ; i++)
{
printf("%d \t %s \t %d \t %d \t %d \t %d \n", current_ptr->empno,current_ptr->name,
current_ptr->bpay, current_ptr->allow, current_ptr->ded,current_ptr->npay) ;
current_ptr=current_ptr->next;
}
getch() ;
}
```

OUTPUT :

YOUR ANNUAL PAYMENT CALCULATOR

Enter your id number:567980

Enter your name :ARUNKARTHIK

Enter your basic pay:850000

Enter your allowance:500000

Enter your detection:350000

Your annual income is1000000

CREATIONOFTELEPHONEDIRECTORY

PROGRAM:

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
typedef  struct  Phonebook_Contacts
{
char FirstName[20];
char LastName[20];
char PhoneNumber[20];
}
 phone;
void AddEntry(phone * );
 void DeleteEntry(phone * );
 void PrintEntry(phone * );
void SearchForNumber(phone * );
 int counter = 0;
char FileName[256];
FILE *pRead;
FILE *pWrite;
void main (void)
{
phone *phonebook;
phonebook = (phone*) malloc(sizeof(phone)*100); int iSelection = 0;
if (phonebook == NULL)
printf("Out of Memory. The program will now exit");
 else
{
do
```

```
{
printf("\n\t\t\tPhonebook Menu");
printf("\n\n\t(1)\tAdd Friend");
printf("\n\t(2)\tDelete Friend");
printf("\n\t(3)\tDisplay Phonebook Entries");
printf("\n\t(4)\tSearch for Phone Number");
printf("\n\t(5)\tExit Phonebook");
printf("\n\nWhat would you like to do? ");
scanf("%d", &iSelection);
if (iSelection == 1)
{
AddEntry(phonebook);
}
if (iSelection == 2)
{
DeleteEntry(phonebook);
}
if (iSelection == 3)
{
PrintEntry(phonebook);
}
if (iSelection == 4)
{
SearchForNumber(phonebook);
}
if (iSelection == 5)
{
printf("\nYou have chosen to exit the Phonebook.\n");
exit(1);
}
}while (iSelection <= 4);
```

```
}
}
void AddEntry (phone * phonebook)
{
pWrite = fopen("phonebook_contacts.dat", "a");
if ( pWrite == NULL )
{
perror("The following error occurred ");
 exit(EXIT_FAILURE);
}
else
{
counter++;
realloc(phonebook, sizeof(phone));
 printf("\nFirst Name: ");
scanf("%s", phonebook[counter-1].FirstName);
printf("Last Name: ");
scanf("%s", phonebook[counter-1].LastName);
printf("Phone Number (XXX-XXX-XXXX): ");
scanf("%s", phonebook[counter-1].PhoneNumber);
printf("\n\tFriend successfully added to Phonebook\n");
fprintf(pWrite, "%s\t%s\t%s\n", phonebook[counter-1].FirstName,phonebook [counter-1].LastName,
phonebook[counter-1].PhoneNumber);
fclose(pWrite);
}
}
void DeleteEntry (phone * phonebook)
{
int x = 0; int i = 0;
char deleteFirstName[20];
```

```
 char deleteLastName[20];
printf("\nFirst name: ");
scanf("%s", deleteFirstName);
printf("Last name: ");
scanf("%s", deleteLastName);
for (x = 0; x < counter; x++)
{
if (strcmp(deleteFirstName, phonebook[x].FirstName) ==0)
{
if (strcmp(deleteLastName, phonebook[x].LastName) == 0)
{
for ( i = x; i < counter - 1; i++ )
{
strcpy(phonebook[i].FirstName,phonebook[i+1].FirstName);
strcpy(phonebook[i].LastName,phonebook[i+1].LastName);
strcpy(phonebook[i].PhoneNumber,phonebook[i+1].PhoneNumber);
}
printf("Record deleted from the phonebook.\n\n");
--counter; return;
}
}
}
printf("That contact was not found, please try again.");
}
void PrintEntry (phone * phonebook)
{
int x = 0;
printf("\nPhonebook Entries:\n\n ");
pRead = fopen("phonebook_contacts.dat", "r");
 if ( pRead == NULL)
{
```

```
perror("The following error occurred: ");
 exit(EXIT_FAILURE);
}
else
{
for( x = 0; x < counter; x++)
{
printf("\n(%d)\n", x+1); printf("Name: %s %s\n",
phonebook[x].FirstName, phonebook[x].LastName);
printf("Number: %s\n", phonebook[x].PhoneNumber);
}
}
fclose(pRead);
}
void SearchForNumber (phone * phonebook)
{
int x = 0;
char TempFirstName[20];
char TempLastName[20];
printf("\nPlease type the name of the friend you wish to find a number for.");
 printf("\n\nFirst Name: ");
scanf("%s", TempFirstName);
printf("Last Name: ");
scanf("%s", TempLastName);
for (x = 0; x < counter; x++)
{
if (strcmp(TempFirstName, phonebook[x].FirstName) == 0)
{
if (strcmp(TempLastName, phonebook[x].LastName) == 0)
{
```

```
printf("\n%s %s's phone number is %s\n",phonebook[x].FirstName,phonebook[x].LastName,
phonebook[x].PhoneNumber);

}

}

}

}
```

OUTPUT:

What would you like to do? 1

First name : ARUN

Last name :KARTHIK

Phone number :812-251-8082

Friend successfully added to phone book

What would you like to do ?2

First name :ARUN

Last name :KARTHIK

Phone number:812-251-8082

Recoed removed from phonebook

What would you like to do?3

Phonebook entries(1)

Name : ARUNKARTHIK

Phone Number:812-251-8082

What would you like to do?4

Please type the name of the friend you wish to find number for

First name :ARUN

Last name :KARTHIK

ARUNKARTHIK phonenumber:812-251-8082

What would you like to do?5

You choosed exit the phonebook