|  |  |
| --- | --- |
| **`EXP NO:** | **PROGRAM USING I/O STATEMENTS AND EXPRESSIONS** |
| **DATE:** |

**AIM:**

To write a C program using I/O statements and expressions.

**ALGORITHM:**

**Step 1:** Start the program.

**Step 2:** Get input from the user.

**Step 3:** Calculate the addition, subtraction, multiplication, division.

**Step 3:** Print the output.

**Step 4:** Stop the program.

**SOURCE CODE:**

#include <stdio.h>

#include <conio.h>

int main()

{

int a,b,c,d,e;

float f;

clrscr();

printf("Input two integers**\n**");

scanf("%d%d", &a, &b);

c = a + b; //addition

d=a-b; //subtraction

e=a\*b; //multiplication

f=a/b; // division

printf("(%d) +(%d) = (%d)**\n**", a, b, c);

printf("(%d) -(%d) = (%d)**\n**", a, b, d);

printf("(%d) \*(%d) = (%d)**\n**", a, b, e);

printf("(%d) + (%d) = (%f)**\n**", a, b, f);

return 0;

}

**OUTPUT:**

Input two integers

10 5

(10) + (5) =15

(10) – (5) =5

(10) \* (5) =50

(10) / (5) =2.000000

**RESULT:**

Thus the C program using I/O statements and expressions was successfully executed and verified.

|  |  |
| --- | --- |
| **EXP NO:** | **PROGRAM USING DECISION-MAKING CONSTRUCTS** |
| **DATE:** |

**AIM:**

To write a C program using decision-making constructs.

**ALGORITHM:**

**Step 1:** Start the program.

**Step 2:** Get input from the user.

**Step 3:** Check the condition and compare the two inputs.

**Step 4:** Ifthe condition is true print the largest number, otherwise print another number.

**Step 5:** Print the output

**Step 6:** Stop the program

**SOURCE CODE:**

#include <stdio.h>

#include <conio.h>

void main()

{

int a, b;

float f;

clrscr();

printf("Input two integers\n");

scanf("%d%d", &a, &b);

if(a>b)

{

printf("%d is larger", a);

}

else

{

printf("%d is larger", b);

}

getch();

}

**OUTPUT:**

Input two integers

5 6

6 is larger

**RESULT:**

Thus the C program using decision making constructs was successfully executed and verified.

|  |  |
| --- | --- |
| **EXP NO:** | **SALARY SLIPS OF EMPLOYEES USING STRUCTURES AND POINTERS** |
| **DATE:** |

**AIM:**

To write a C program to generate salary slip of employees using structures and pointers.

**ALGORITHM:**

**Step 1:** Start the program

**Step 2:** Get the employee details using structure data type

**Step 3:** Check the pay rate of each employee using multiway branch statement

**Step 4:** Calculate basic salary reduce the pf and tax amount

**Step 5:** Print the salary slip of employee

**Step 6:** Stop the program**.**

**SOURCE CODE:**

#include<stdio.h>

#include<dos.h>

struct employee

{

int NO;

     char NAME[10];

    int DESIGN\_CODE;

    int DAYS\_WORKED;

}EMP[12]={ {1,"GANESH",1,25},{2,"MAHESH",1,30},{3,"SURESH",2,28},{4,"KALPESH",2,26},

{5,"RAHUL",2,24},{6,"SUBBU",2,25},{7,"RAKESH",2,23},{8,"ATUL",2,22},

{9,"DHARMESH",3,26},{10,"AJAY",3,26},{11,"ABDUL",3,27},{12,"RASHMI",4,29}

};

void main()

{

int EMPNO;

     void gen\_payslip(int);

   clrscr();

     printf("ENTER THE EMPLOYEE NO TO GENERATE PAYSLIP : ");

     scanf("%d",&EMPNO);

     if(EMPNO>0 && EMPNO<13)

         gen\_payslip(EMPNO);

     else

         printf("\nYOU HAVE ENTERED WRONG EMP NO. !!");

     getch();

}

void gen\_payslip(int EMPNO)

{

struct date D;

char DESG[10];

  float NETPAY,BASIC,PF,PAYRATE,PTAX=200;

  getdate(&D);

  printf("\n\n\t\t\t\tMRK INSTITUTE OF TECHNOLOGY");

  printf("\n\t\t\t\tSALARY MONTH %d %d\n",D.da\_mon,D.da\_year);

  printf("\n\n\tEMP.NO.: %d\t\tEMP.NAME: %s",EMPNO,EMP[EMPNO-1].NAME);

  switch(EMP[EMPNO-1].DESIGN\_CODE)

  {

case 1: PAYRATE=400;

printf("\tDESIGNATION: CLERK");

       break;

case 2: PAYRATE=300;

      printf("\tDESIGNATION: DRIVER");

       break;

case 3: PAYRATE=250;

       printf("\tDESIGNATION: HELPER");

       break;

case 4: PAYRATE=350;

       printf("\tDESIGNATION: COMP.OPTR");

       break;

}

BASIC=PAYRATE\*EMP[EMPNO-1].DAYS\_WORKED;

  PF=BASIC/10;

  printf("\n\n\tDAYS WORKED: %d",EMP[EMPNO-1].DAYS\_WORKED);

printf("\t\tPAY RATE: %.0f\t\tGEN.DATE: %d/%d/%d", PAYRATE, D.da\_day, D.da\_mon, D.da\_year);

  printf("\n\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

  printf("\n\n\tEARNINGS\tAMOUNT(RS.)\t\tDEDUCTIONS\tAMOUNT(RS.)");

  printf("\n\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

  printf("\n\n\tBASIC PAY\t%.0f\t\t\tP.F.\t\t%.0f",BASIC,PF);

  printf("\n\n\t\t\t\t\t\tPROF.TAX\t%.0f",PTAX);

  printf("\n\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

  printf("\n\n\tGROSS EARN.\t%.0f\t\t\tTOTAL DEDUCT.\t%.0f",BASIC,PF+PTAX);

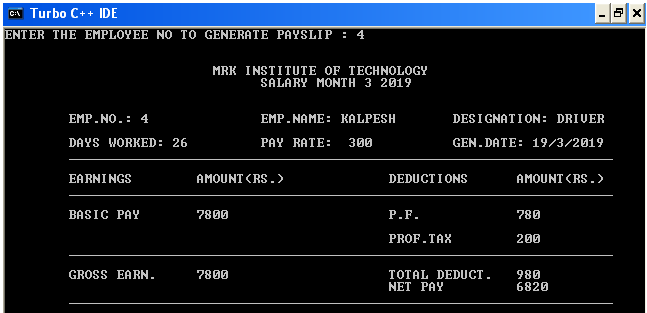
  NETPAY=BASIC-(PF+PTAX);

  printf("\n\t\t\t\t\t\tNET PAY\t\t%.0f",NETPAY);

  printf("\n\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

**OUTPUT:**

****

**RESULT:**

Thus the C program to generate salary slip of employees using structures and pointers was successfully executed and verified.

|  |  |
| --- | --- |
| **EXP NO:** | **INTERNAL MARKS OF STUDENTS USING ARRAY AND FUNCTIONS** |
| **DATE:** |

**AIM:**

To write a C Program to Compute internal marks of students for five different subjects using array and functions.

**ALGORITHM**

**Step 1:** Start

**Step 2:** Declare variables

**Step 3:** Read the number of students .

**Step 4:** Read the student mark details

**Step 5:** Calculate internal mark by i=total of three test marks / 3 for each subject per student.

**Step 6:** Display the output of the calculations for all the students.

**Step 7:** Stop

**SOURCE CODE:**

#include<stdio.h>

#include<conio.h>

struct stud

{

char name[20];

long int rollno;

int marks[50];

int i[5];

}students[10];

void calcinternal(int);

int main()

{

int a,b,j,n,total;

clrscr();

printf("How many students : \n");

scanf("%d",&n);

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

{

printf("\n\nEnter the details of %d student : ", a+1);

printf("\n\nEnter student %d Name : ", a);

scanf("%s", students[a].name);

printf("\n\nEnter student %d Roll Number : ", a);

scanf("%ld", &students[a].rollno);

total=0;

for(b=0;b<=4;++b)

{

for(j=0;j<=2;++j)

{

printf("\n\nEnter the test %d mark of subject-%d : ",j+1, b+1);

scanf("%d", &students[a].marks[b,j]);

}

}

}

calcinternal(n);

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

{

printf("\n\n\t\t\t\tMark Sheet\n");

printf("\nName of Student : %s", students[a].name);

printf("\t\t\t\t Roll No : %ld", students[a].rollno);

printf("\n------------------------------------------------------------------------");

for(b=0;b<5;b++)

{

printf("\n\n\t Subject %d internal \t\t :\t %d", b+1, students[a].i[b]);

}

printf("\n\n------------------------------------------------------------------------\n");

getch();

}

return(0);

}

void calcinternal(int n)

{

int a,b,j,total;

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

{

for(b=0;b<5;b++)

{

total=0;

for(j=0;j<=2;++j)

{

total += students[a].marks[b,j];

}

students[a].i[b]=total/3;

}

}

}

**OUTPUT:**

How many students : 1

Enter the details of 1 student :

Enter student 1 Name : S.Suresh

Enter student 1 Roll Number : 536435

Enter the test 1mark of subject-1 : 46 42

Enter the test 2 mark of subject-1 : 56

Enter the test 3 mark of subject-1 : 76

Enter the test 1 mark of subject-2 : 85

Enter the test 2mark of subject-2 : 75

Enter the test 3mark of subject-2 : 75

Enter the test 1mark of subject-3 : 66

Enter the test 2 mark of subject-3 : 86

Enter the test 3 mark of subject-3 : 70

Enter the test 1 mark of subject-4 : 25

Enter the test 2mark of subject-4 : 35

Enter the test 3mark of subject-4 : 61

Enter the test 1 mark of subject-5 : 45

Enter the test 2mark of subject-5 : 75

Enter the test 3mark of subject-5 : 60

Mark Sheet

Name of Student : S.Suresh Roll No : 536435

------------------------------------------------------------------------

subject 1 internal : 59

subject 2 internal : 78

subject 3 internal : 74

subject 4 internal : 40

subject 5 internal : 60

------------------------------------------------------------------------

**]**

**RESULT:**

Thus the C program to compute internal marks of students for five different subjects using structures and functions was successfully executed and verified.

|  |  |
| --- | --- |
| **EXP NO:** | **BANKING APPLICATION USING SEQUENTIAL ACCESS FILE** |
| **DATE:** |

**AIM:**

To write a C program to count the number of account holders whose balance is less than the minimum balance using sequential access file.

**ALGORITHM:**

**Step 1:** Start the program

**Step 2:** Enter the account holder details

**Step 3:** Then perform display, depositor, withdraw, number of account holder whose balance is less than

minimum balance, delete operation using multiway branch statement.

**Step 4:** Print the output

**Step 5:** Stop the program.

**SOURCE CODE:**

#include <stdio.h>  
#include <stdlib.h>  
#include <conio.h>  
#include <string.h>  
#define MINBAL 500  
struct Bank\_Account  
{

char no[10];  
char name[20];  
char balance[15];

};  
struct Bank\_Account acc;  
void main()  
{

long int pos1,pos2,pos;  
FILE \*fp;  
char \*ano,\*amt;  
char choice;  
int type,flag=0;  
float bal;  
do  
{

clrscr();  
      fflush(stdin);  
      printf("1. Add a New Account Holder\n");  
      printf("2. Display\n");  
      printf("3. Deposit or Withdraw\n");  
      printf("4. Number of Account Holder Whose Balance is less than the Minimum

Balance\n");  
      printf("5. Delete All\n");  
      printf("6. Stop\n");  
    printf("Enter your choice : ");  
      choice=getchar();  
      switch(choice)

{  
      case '1' :  
   fflush(stdin);  
   fp=fopen("acc.dat","a");  
   printf("\nEnter the Account Number : ");  
   gets(acc.no);  
   printf("\nEnter the Account Holder Name : ");  
   gets(acc.name);  
   printf("\nEnter the Initial Amount to deposit : ");  
   gets(acc.balance);  
   fseek(fp,0,2);  
   fwrite(&acc,sizeof(acc),1,fp);  
   fclose(fp);  
   break;  
        case '2' :  
   fp=fopen("acc.dat","r");  
   if(fp==NULL)  
     printf("\nFile is Empty");  
   else  
   {  
   printf("\nA/c Number\tA/c Holder Name   Balance\n");  
   while(fread(&acc,sizeof(acc),1,fp)==1)  
       printf("%-10s\t\t%-20s\t%s\n",acc.no,acc.name,acc.balance);  
   fclose(fp);  
   }  
   break;  
        case '3' :  
   fflush(stdin);  
   flag=0;  
   fp=fopen("acc.dat","r+");  
   printf("\nEnter the Account Number : ");  
   gets(ano);  
   for(pos1=ftell(fp);fread(&acc,sizeof(acc),1,fp)==1;pos1=ftell(fp))  
   {  
     if(strcmp(acc.no,ano)==0)  
     {  
       printf("\nEnter the Type 1 for deposit  & 2 for withdraw : ");  
       scanf("%d",&type);  
       printf("\nYour Current Balance is : %s",acc.balance);  
       printf("\nEnter the Amount to transact : ");  
       fflush(stdin);  
       gets(amt);  
       if(type==1)  
 bal = atof(acc.balance) + atof(amt);  
       else  
       {  
 bal = atof(acc.balance) - atof(amt);  
 if(bal<0)  
 {  
   printf("\nRs.%s Not available in your

A/c\n",amt);  
   flag=2;  
   break;

}

}  
       flag++;  
       break;  
     }

}  
   if(flag==1)  
   {  
       pos2=ftell(fp);  
       pos = pos2-pos1;  
       fseek(fp,-pos,1);  
       sprintf(amt,"%.2f",bal);  
       strcpy(acc.balance,amt);  
       fwrite(&acc,sizeof(acc),1,fp);  
   }  
   else if(flag==0)  
     printf("\nA/c Number Not exits... Check it again");  
   fclose(fp);  
   break;

case '4':  
   fp=fopen("acc.dat","r");  
   flag=0;  
   while(fread(&acc,sizeof(acc),1,fp)==1)  
   {  
     bal = atof(acc.balance);  
     if(bal<MINBAL)  
     flag++;  
   }  
   printf("\nThe Number of Account Holder whose Balance less than the

Minimum Balance : %d",flag);  
   fclose(fp);  
   break;

case '5':  
   remove("acc.dat");  
   break;

case '6':  
   fclose(fp);  
   exit(0);

}  
    printf("\nPress any key to continue....");  
    getch();

} while (choice!='6');

}

**OUTPUT:**

1. Add a New Account Holder

2. Display

3. Deposit or Withdraw

4. Number of Account Holder Whose Balance is less than the Minimum Balance

5. Delete All

6. Stop

Enter your choice : 1

Enter the Account Number : 565656

Enter the Account Holder Name : sathya

Enter the Initial Amount to deposit : 1000

Press any key to continue....

Enter your choice : 2

A/c Number A/c Holder Name Balance

565656 sathya 1000

Press any key to continue....

**RESULT:**

Thus the C program to count the number of account holders whose balance is less than the minimum balance using sequential access file was successfully executed and verified.

|  |  |
| --- | --- |
| **EXP NO:** | **REAL TIME APPLICATION IN RAILWAY RESERVATION SYSTEM** |
| **DATE:** |

**AIM:**

To write a C program for railway reservation system with the modules namely booking, availability checking, and cancellation.

**ALGORITHM:**

**Step 1:** Start the program

**Step 2:** Enter the customer detail

**Step 3:** Perform booking the railway ticket, then check the availability of seats then cancel ticket using multiway branch statement

**Step 4:** Stop the program.

**SOURCE CODE:**

#include<stdio.h>

#include<conio.h>

int first=5,second=5,third=5;

struct node

{

int ticketno;

int phoneno;

char name[100];

char address[100];

}s;

void booking()

{

printf("Enter your details");

printf("\nName:");

scanf("%s",s.name);

printf("\nphonenumber:");

scanf("%d",&s.phoneno);

printf("\naddress:");

scanf("%s",s.address);

printf("\nticketnumber only 1-10:");

scanf("%d",&s.ticketno);

}

void availability()

{

int c;

printf("Availability cheking");

printf("\n1.First class\n2.Second class\n3.Third class\n");

printf("Enter the option");

scanf("%d",&c);

switch(c)

{

case 1:

if(first>0)

{

printf("Seat Available\n");

first--;

}

else

{

printf("Seat not Available");

}

break;

case 2:

if(second>0)

{

printf("Seat Available\n");

second--;

}

else

{

printf("Seat not Available");

}

break;

case 3:

I f(third>0)

{

printf("Seat Available\n");

third--;

}

else

{

printf("Seat not Available");

}

break;

default:

break;

}

}

void cancel()

{

int c;

printf("Cancel\n");

printf("Which class you want to cancel");

printf("\n1.First class\n2.Second class\n3.Third class\n");

printf("Enter the option");

scanf("%d",&c);

switch(c)

{

case 1:

first++;

break;

case 2:

second++;

break;

case 3:

third++;

break;

default:

break;

}

printf("\nTicket is canceled");

}

void main()

{

int n;

clrscr();

printf("Welcome to railway ticket reservation\n");

while(1)

{

printf("\n1.Booking\n2.Availability cheking\n3.Cancel\n4.Exit\nEnter your option:");

scanf("%d",&n);

switch(n)

{

case 1:

booking();

break;

case 2:

availability();

break;

case 3:

cancel();

break;

case 4:

exit(0);

break;

default:

break;

}

getch();

}

}

**OUTPUT:**

Welcome to railway ticket reservation

1.Booking

2.Availability cheking

3.Cancel

Enter your option:1

Enter your details

Name: akshu

Phonenumber: 620430

Address:4/148 near railway junction, thanjavur.614714

Ticket number only 1-10:   5

Welcome to railway ticket reservation

1.Booking

2.Availability cheking

3.Cancel

Enter your option:2

Availability cheking

1.First class

2.Second class

3.Third class

Enter the option1

Seat available

Welcome to railway ticket reservation

1.Booking

2.Availability cheking

3.Cancel

Enter your option:3

Cancel

Which class you want to cancel

1.First class

2.Second class

3.Third class

Enter the option1

Ticket is canceled

**RESULT:**

Thus the C program for railway reservation system with the modules namely booking, availability checking, and cancellation was successfully executed and verified.