**Question 01**

**CODE**

#include<stdio.h>

void matrice\_2by2();

void matrice\_4by4();

void matrice\_8by8();

char max1(int n, char matrix[n][n], int r, int c);

void main(void){

printf("................................................\n");

printf("Roll Number: 23K-3032\t\tName:Shah Hunain\n");

printf("................................................\n\n");

do{

char choice;

printf("\t\tWelcome\nPlease choose from the following:\n");

printf("1. Enter values\n2.Enter roll number\n3. Exit: \n");

char option;

fflush(stdin);

option=getchar();

if(option=='1'){

printf("1) 2-by-2 matrice\n2) 4-by-4 matrice\n3) 8-by-8 matrice\n");

printf("Choice:");

fflush(stdin);

choice=getchar();

printf("\n");

switch(choice){

case '1':

matrice\_2by2();

break;

case '2':

matrice\_4by4();

break;

case '3':

matrice\_8by8();

break;

default:

printf("INVALID INPUT\n");

}

}

if(option=='2'){

int n=4;

char roll\_number[8];

printf("Enter the roll number: ");

fflush(stdin);

gets(roll\_number);

char number[4];

number[0]=roll\_number[4];

number[1]=roll\_number[5];

number[2]=roll\_number[6];

number[3]=roll\_number[7];

char matrix[n][n];

char result[n/2][n/2];

int i, j;

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

for(j=0; j<n; j++){

matrix[i][j]=number[j];

}

}

printf("\n");

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

for(j=0; j<n; j++){

printf("%c\t",matrix[i][j]);

}

printf("\n");

}

for(i=0;i <n/2; i++){

for(j=0; j<n/2; j++){

result[i][j] = max1(n, matrix, i\*2, j\*2);

}

}

printf("\n");

for(i=0;i <n/2; i++){

for(j=0; j<n/2; j++){

printf("%c\t", result[i][j]);

}

printf("\n");

}

}

if(option=='3'){

printf("Exiting....Thank You!\n");

break;

}

}while(1);

}

void matrice\_2by2(){

int i,j;

int matrice[2][2];

printf("Enter the elements of 2-by-2 matrice:\n");

for(i=0; i<2; i++){

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

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

}

}

printf("\n");

int max=matrice[0][1];

for(i=0; i<2; i++){

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

printf("%d\t",matrice[i][j]);

if(max<matrice[i][j]){

max=matrice[i][j];

}

}

printf("\n");

}

printf("\n%d\n",max);

}

void matrice\_4by4(){

int i,j;

int matrice[4][4];

int new1[2][2],new2[2][2],new3[2][2],new4[2][2];

printf("Enter the elements of 4-by-4 matrice:\n");

for(i=0; i<4; i++){

for(j=0; j<4;j++){

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

}

}

for (int i=0; i<2; i++){

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

new1[i][j]=matrice[i][j];

}

}

for (int i=0; i<2; i++){

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

new2[i][j]=matrice[i][j+2];

}

}

for (int i=0; i<2; i++){

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

new3[i][j]=matrice[i+2][j];

}

}

for (int i=0; i<2; i++){

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

new4[i][j]=matrice[i+2][j+2];

}

}

int max1=new1[0][0],max2=new2[0][0],max3=new3[0][0],max4=new4[0][0];

for(i=0; i<2; i++){

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

printf("%d\t",new1[i][j]);

if(max1<new1[i][j]){

max1=new1[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new2[i][j]);

if(max2<new2[i][j]){

max2=new2[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new3[i][j]);

if(max3<new3[i][j]){

max3=new3[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new4[i][j]);

if(max4<new4[i][j]){

max4=new4[i][j];

}

}

printf("\n");

}

printf("\n");

int final\_matrice[2][2]={{max1,max2},{max3,max4}};

printf("The Final Matrice is:\n");

for(i=0; i<2; i++){

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

printf("%d\t",final\_matrice[i][j]);

}

printf("\n");

}

}

#include <stdio.h>

void matrice\_8by8(){

int i,j;

int matrice[8][8];

int new1[2][2],new2[2][2],new3[2][2],new4[2][2],new5[2][2],new6[2][2],new7[2][2],new8[2][2],new9[2][2],new10[2][2],new11[2][2],new12[2][2],new13[2][2],new14[2][2],new15[2][2],new16[2][2];

printf("Enter the elements of 8-by-8 matrice:\n");

for(i=0; i<8; i++){

for(j=0; j<8;j++){

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

}

}

for (i=0; i<2; i++){

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

new1[i][j]=matrice[i][j];

}

}

for (i=0; i<2; i++){

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

new2[i][j]=matrice[i][j+2];

}

}

for (i=0; i<2; i++){

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

new3[i][j]=matrice[i][j+4];

}

}

for (i=0; i<2; i++){

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

new4[i][j]=matrice[i][j+6];

}

}

for (i=0; i<2; i++){

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

new5[i][j]=matrice[i+2][j];

}

}

for (i=0; i<2; i++){

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

new6[i][j]=matrice[i+2][j+2];

}

}

for (i=0; i<2; i++){

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

new7[i][j]=matrice[i+2][j+4];

}

}

for (i=0; i<2; i++){

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

new8[i][j]=matrice[i+2][j+6];

}

}

for (i=0; i<2; i++){

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

new9[i][j]=matrice[i+4][j];

}

}

for (i=0; i<2; i++){

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

new10[i][j]=matrice[i+4][j+2];

}

}

for (i=0; i<2; i++){

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

new11[i][j]=matrice[i+4][j+4];

}

}

for (i=0; i<2; i++){

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

new12[i][j]=matrice[i+4][j+6];

}

}

for (i=0; i<2; i++){

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

new13[i][j]=matrice[i+6][j];

}

}

for (i=0; i<2; i++){

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

new14[i][j]=matrice[i+6][j+2];

}

}

for (i=0; i<2; i++){

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

new15[i][j]=matrice[i+6][j+4];

}

}

for (i=0; i<2; i++){

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

new16[i][j]=matrice[i+6][j+6];

}

}

int max1=new1[0][0],max2=new2[0][0],max3=new3[0][0],max4=new4[0][0],max5=new5[0][0],max6=new6[0][0],max7=new7[0][0],max8=new8[0][0],max9=new9[0][0],max10=new10[0][0],max11=new11[0][0],max12=new12[0][0],max13=new13[0][0],max14=new14[0][0],max15=new15[0][0],max16=new16[0][0];

for(i=0; i<2; i++){

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

printf("%d\t",new1[i][j]);

if(max1<new1[i][j]){

max1=new1[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new2[i][j]);

if(max2<new2[i][j]){

max2=new2[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new3[i][j]);

if(max3<new3[i][j]){

max3=new3[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new4[i][j]);

if(max4<new4[i][j]){

max4=new4[i][j];

}

}

printf("\n");

}

for(i=0; i<2; i++){

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

printf("%d\t",new5[i][j]);

if(max5<new5[i][j]){

max5=new5[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new6[i][j]);

if(max6<new6[i][j]){

max6=new6[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new7[i][j]);

if(max7<new7[i][j]){

max7=new7[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new8[i][j]);

if(max8<new8[i][j]){

max8=new8[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new9[i][j]);

if(max9<new9[i][j]){

max9=new9[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new10[i][j]);

if(max10<new10[i][j]){

max10=new10[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new11[i][j]);

if(max11<new11[i][j]){

max11=new11[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new12[i][j]);

if(max12<new12[i][j]){

max12=new12[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new13[i][j]);

if(max13<new13[i][j]){

max13=new13[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new14[i][j]);

if(max14<new14[i][j]){

max14=new14[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new15[i][j]);

if(max15<new15[i][j]){

max15=new15[i][j];

}

}

printf("\n");

}

printf("\n");

for(i=0; i<2; i++){

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

printf("%d\t",new16[i][j]);

if(max16<new16[i][j]){

max16=new16[i][j];

}

}

printf("\n");

}

printf("\n");

int final\_matrice[4][4]={{max1,max2,max3,max4},{max5,max6,max7,max8},{max9,max10,max11,max12},{max13,max14,max15,max16}};

printf("The Final Matrice is:\n");

for(i=0; i<4; i++){

for(j=0; j<4; j++){

printf("%d\t",final\_matrice[i][j]);

}

printf("\n");

}

printf("\n");

}

char max1(int n, char matrix[n][n], int r, int c){

int max = 0;

int i, j;

for(i=r; i<r+2; i++){

for(j=c; j<c+2; j++){

if(matrix[i][j] > max){

max = matrix[i][j];

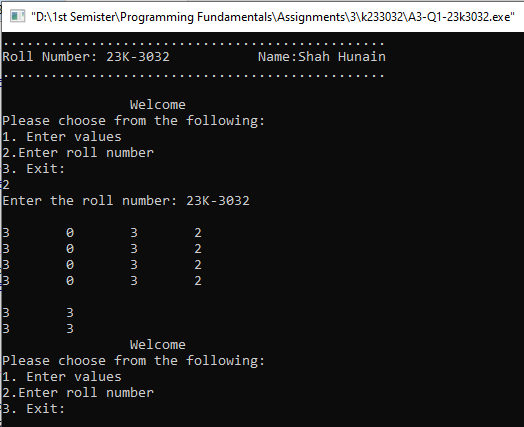
}

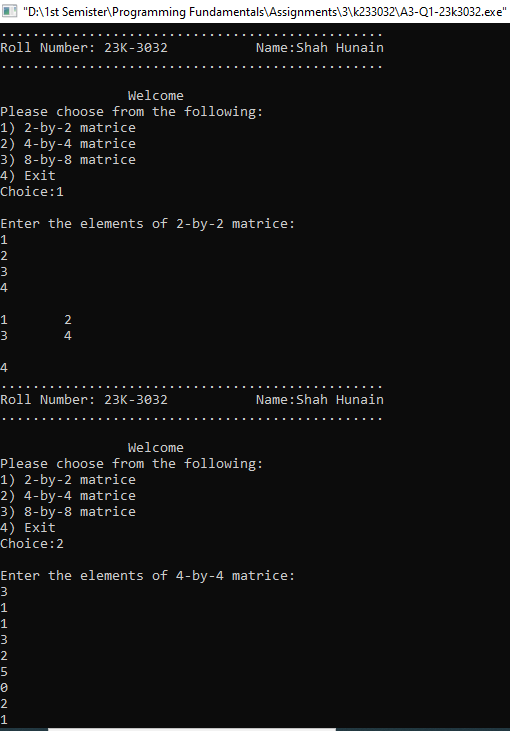
}

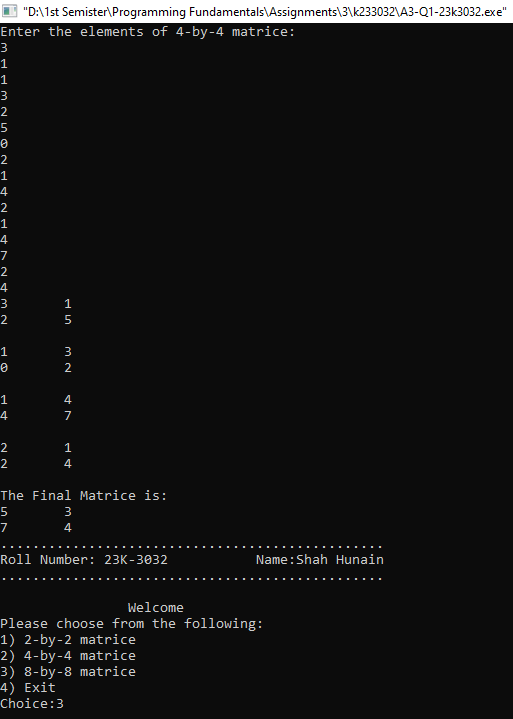
}

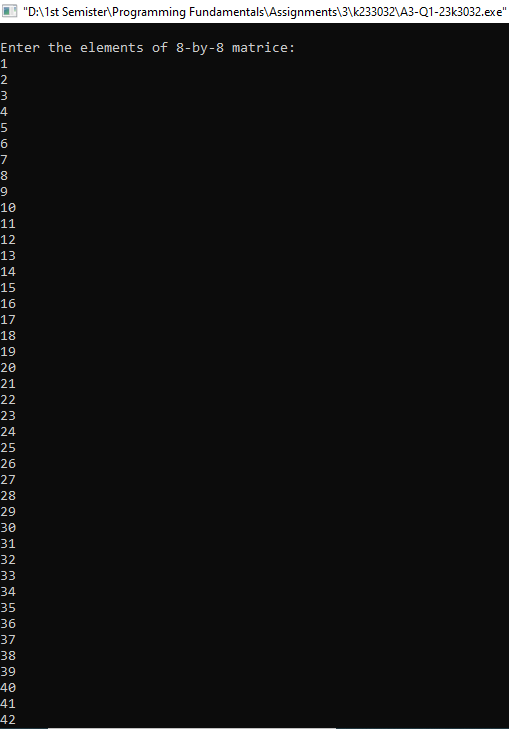
return max;

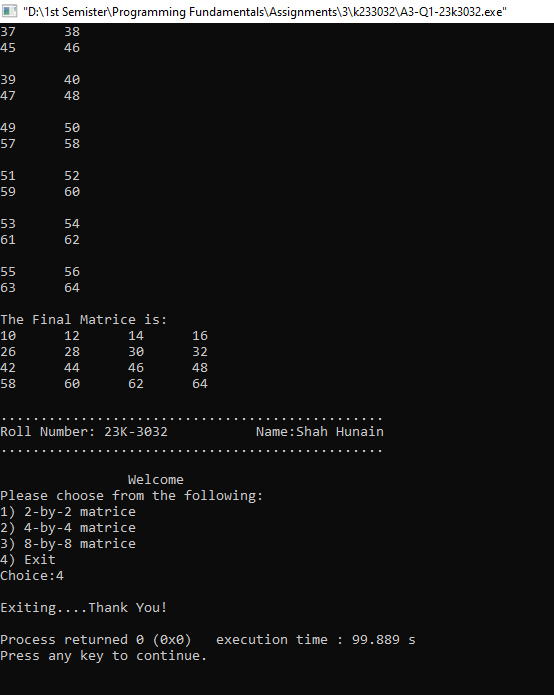
}

**OUTPUT**









**Question 02**

**CODE**

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

#include<time.h>

struct attributes{

char name[20];

char role[10];

int communication;

int teamwork;

int creativity;

};

void main(void){

printf("................................................\n");

printf("Roll Number: 23K-3032\t\tName:Shah Hunain\n");

printf("................................................\n");

struct attributes HR[5], finance[5], Marketing[5], logistics[5];

char r1[5][10]={"Director","Executive","Manager","Employee","Trainee"};

char r2[5][10]={"Director","Executive","Manager","Employee","Trainee"};

char r3[5][10]={"Director","Executive","Manager","Employee","Trainee"};

char r4[5][10]={"Director","Executive","Manager","Employee","Trainee"};

char names[20][10] = {"Roman","Reigns","Seth","Freakin","Rollins","Dean","Ambrose","Kevin","Owens","Randy","Brock","Lesnar","Dominik","Sami","Drew","Mcintyre","Finn","Balor","CM","Punk"};

char zero[]={'\0'};

srand(time(NULL));

int i,n;

for(i=0;i<5;i++){

n=rand() % 20;

if(strcmp(names[n],zero)!=0){

strcpy(HR[i].name,names[n]);

strcpy(names[n],zero);

}

}

for(i=0; i<5; i++){

n=rand() % 20;

if(strcmp(names[n],zero)!=0){

strcpy(finance[i].name,names[n]);

strcpy(names[n],zero);

}

}

for(i=0; i<5; i++){

n=rand() % 20;

if(strcmp(names[n],zero)!=0){

strcpy(Marketing[i].name,names[n]);

strcpy(names[n],zero);

}

}

for(i=0; i<5; i++){

n=rand() % 20;

if(strcmp(names[n],zero)!=0){

strcpy(logistics[i].name,names[n]);

strcpy(names[n],zero);

}

}

for(i=0; i<5; i++){

n=rand() % 5;

if(strcmp(r1[n],zero)==1){

strcpy(HR[i].role,r1[n]);

strcpy(r1[n],zero);

}

}

for(i=0; i<5; i++){

n=rand() % 5;

if(strcmp(r2[n],zero)==1){

strcpy(finance[i].role,r2[n]);

strcpy(r2[n],zero);

}

}

for(i=0; i<5; i++){

n=rand() % 5;

if(strcmp(r3[n],zero)==1){

strcpy(Marketing[i].role,r3[n]);

strcpy(r3[n],zero);

}

}

for(i=0; i<5; i++){

n=rand() % 5;

if(strcmp(r4[n],zero)==1){

strcpy(logistics[i].role,r4[n]);

strcpy(r4[n],zero);

}

}

for(i=0; i<5; i++){

HR[i].communication=rand() % 100 +1;

HR[i].teamwork=rand() % 100+1;

HR[i].creativity=rand() % 100+1;

}

for(i=0; i<5; i++){

finance[i].communication=rand() % 100 +1;

finance[i].teamwork=rand() % 100 +1;

finance[i].creativity=rand() % 100 +1;

}

for(i=0; i<5; i++){

Marketing[i].communication=rand() % 100 +1;

Marketing[i].teamwork=rand() % 100 +1;

Marketing[i].creativity=rand() % 100 +1;

}

for(i=0; i<5; i++){

logistics[i].communication=rand() % 100 +1;

logistics[i].teamwork=rand() % 100 +1;

logistics[i].creativity=rand() % 100 +1;

}

int sums[4]={0}, s1=0;

for(i=0; i<5; i++){

sums[0]=sums[0]+HR[i].communication;

sums[0]=sums[0]+HR[i].teamwork;

sums[0]=sums[0]+HR[i].creativity;

}

for(i=0; i<5; i++){

sums[1]=sums[1]+finance[i].communication;

sums[1]=sums[1]+finance[i].teamwork;

sums[1]=sums[1]+finance[i].creativity;

}

for(i=0; i<5; i++){

sums[2]=sums[2]+Marketing[i].communication;

sums[2]=sums[2]+Marketing[i].teamwork;

sums[2]=sums[2]+Marketing[i].creativity;

}

for(i=0; i<5; i++){

sums[3]=sums[3]+logistics[i].communication;

sums[3]=sums[3]+logistics[i].teamwork;

sums[3]=sums[3]+logistics[i].creativity;

}

for(i=0; i<4; i++){

if(sums[i]>s1){

s1=sums[i];

}

}

if(s1==sums[0]){

printf("The Best Department is HR :\n");

for(i=0; i<5; i++){

printf("\nName : %s", HR[i].name);

printf("\nRole : %s", HR[i].role);

printf("\nCommunication : %d", HR[i].communication);

printf("\nCreativity : %d", HR[i].creativity);

printf("\nTeamwork : %d", HR[i].teamwork);

printf("\n");

}

}

else if(s1==sums[1]){

printf("\nThe Best Department is Finance :\n");

for(i=0; i<5; i++){

printf("\nName : %s", finance[i].name);

printf("\nRole : %s", finance[i].role);

printf("\nCommunication : %d", finance[i].communication);

printf("\nCreativity : %d", finance[i].creativity);

printf("\nTeamwork : %d", finance[i].teamwork);

printf("\n");

}

}

else if(s1==sums[2]){

printf("\nThe Best Department is Marketing :\n");

for(i=0; i<5; i++){

printf("\nName : %s", Marketing[i].name);

printf("\nRole : %s", Marketing[i].role);

printf("\nCommunication : %d", Marketing[i].communication);

printf("\nCreativity : %d", Marketing[i].creativity);

printf("\nTeamwork : %d", Marketing[i].teamwork);

printf("\n");

}

}

else if(s1==sums[3]){

printf("\nThe Best Department is Logistics :\n");

for(i=0; i<5; i++){

printf("\nName : %s", logistics[i].name);

printf("\nRole : %s", logistics[i].role);

printf("\nCommunication : %d", logistics[i].communication);

printf("\nCreativity : %d", logistics[i].creativity);

printf("\nTeamwork : %d", logistics[i].teamwork);

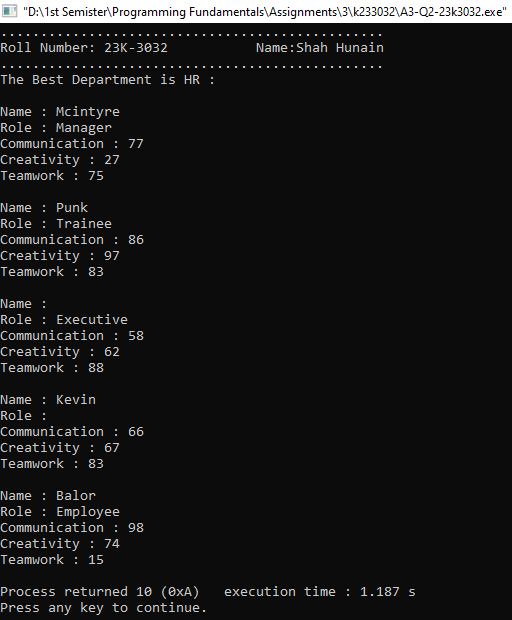
printf("\n");

}

}

}

**OUTPUT**



**Question 03**

**CODE**

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

void combine(int n);

struct personal{

int id;

char name[40];

};

struct department{

int id;

int salary;

};

void main(void){

printf("................................................\n");

printf("Roll Number: 23K-3032\t\tName:Shah Hunain\n");

printf("................................................\n\n");

int i,j,n;

i=j=n=0;

printf("Please enter the number of employees: ");

scanf("%d",&n);

struct personal emp1[n];

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

printf("Enter the id:");

scanf("%d",&emp1[i].id);

fflush(stdin);

printf("Enter the name:");

fflush(stdin);

gets(emp1[i].name);

}

FILE \*personal\_file=fopen("Personal.txt","w");

if(personal\_file==NULL){

printf("Error opening the file\n");

exit(0);

}

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

fprintf(personal\_file,"%d %s\n",emp1[i].id,emp1[i].name);

}

fclose(personal\_file);

struct department emp2[n];

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

printf("Enter id:");

scanf("%d", &emp2[i].id);

printf("Enter salary:");

scanf("%d", &emp2[i].salary);

}

FILE \*department\_file=fopen("Department.txt","w");

if(department\_file==NULL){

printf("Error opening the file\n");

exit(0);

}

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

fprintf(department\_file,"%d %d\n",emp2[i].id,emp2[i].salary);

}

fclose(department\_file);

combine(n);

}

void combine(int n){

int i,j;

struct personal data1[n];

struct department data2[n];

FILE \*personal\_file=fopen("Personal.txt","r");

if(personal\_file==NULL){

printf("Error opening the file\n");

exit(0);

}

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

fscanf(personal\_file,"%d %s\n",&data1[i].id,&data1[i].name);

}

fclose(personal\_file);

FILE \*department\_file=fopen("Department.txt","r");

if(department\_file==NULL){

printf("Error opening the file\n");

exit(0);

}

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

fscanf(department\_file,"%d %d\n",&data2[i].id,&data2[i].salary);

}

fclose(department\_file);

FILE \*merged\_file=fopen("Combine.txt", "w");

if(merged\_file==NULL){

printf("Error opening the file\n");

exit(0);

}

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

for (j=0; j<n; j++){

if (data1[i].id==data2[j].id){

fprintf(merged\_file,"%d %s %d\n",data1[i].id,data1[i].name,data2[j].salary);

}

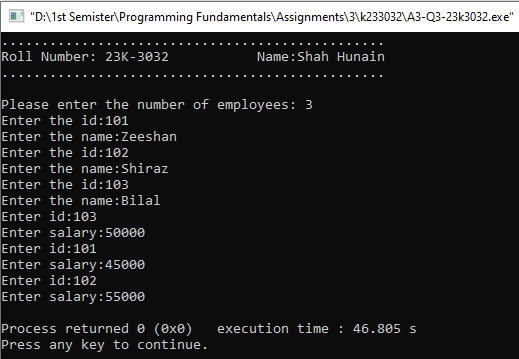
}

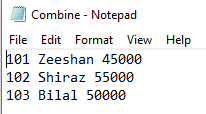
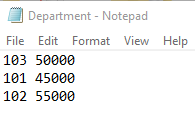
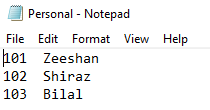
}

fclose(merged\_file);

}

**OUTPUT**





**Question 04**

**CODE**

#include<stdio.h>

#include<string.h>

struct worker{

int worker\_id;

char first\_name[20];

char last\_name[20];

int salary;

char department[20];

struct joining\_date{

int year;

int month;

int date;

int hour;

int minute;

int second;

}j\_d;

};

struct bonus{

int worker\_ref\_id;

int bonus\_amount;

struct bonus\_date{

int year;

int month;

int date;

int hour;

int minute;

int second;

}b\_d;

};

struct title{

int worker\_ref\_id;

char worker\_title[20];

struct affected\_from{

int year;

int month;

int date;

int hour;

int minute;

int second;

}a\_f;

};

void main(void){

struct worker w\_var[8];

struct bonus b\_var[5];

struct title t\_var[8];

printf("................................................\n");

printf("Roll Number: 23K-3032\t\tName:Shah Hunain\n");

printf("................................................\n\n");

int i;

for(i=0; i<5; i++){

printf("Enter the worker id: ");

scanf("%d",&w\_var[i].worker\_id);

printf("\nEnter the first name: ");

fflush(stdin);

gets(w\_var[i].first\_name);

printf("\nEnter the last name: ");

fflush(stdin);

gets(w\_var[i].last\_name);

printf("\nEnter the salary: ");

scanf("%d",&w\_var[i].salary);

printf("\nEnter the year of joining: ");

scanf("%d",&w\_var[i].j\_d.year);

printf("\nEnter the month of joining: ");

scanf("%d",&w\_var[i].j\_d.month);

printf("\nEnter the date of joining: ");

scanf("%d",&w\_var[i].j\_d.date);

printf("\nEnter the department: ");

fflush(stdin);

gets(w\_var[i].department);

printf("\n");

}

int max=0,sum=0;

for (i=0; i<5; i++) {

if(strcmp(w\_var[i].department,"HR")==0){

sum=sum+w\_var[i].salary;

if (max<w\_var[i].salary){

max=w\_var[i].salary;

printf("\n%d\n",w\_var[i].worker\_id);

puts(w\_var[i].first\_name);

puts(w\_var[i].last\_name);

printf("%d\n",w\_var[i].salary);

printf("%d-%d-%d\n",w\_var[i].j\_d.year,w\_var[i].j\_d.month,w\_var[i].j\_d.date);

puts(w\_var[i].department);

}

}

}

printf("Total salary in 'HR' department is: %d\n",sum);

max=sum=0;

for (i=0; i<5; i++) {

if(strcmp(w\_var[i].department,"Admin")==0){

sum=sum+w\_var[i].salary;

if (max<w\_var[i].salary){

max=w\_var[i].salary;

printf("\n%d\n",w\_var[i].worker\_id);

puts(w\_var[i].first\_name);

puts(w\_var[i].last\_name);

printf("%d\n",w\_var[i].salary);

printf("%d-%d-%d\n",w\_var[i].j\_d.year,w\_var[i].j\_d.month,w\_var[i].j\_d.date);

puts(w\_var[i].department);

}

}

}

printf("Total salary in 'Admin' department is: %d\n",sum);

max=sum=0;

for (i=0; i<5; i++){

if (strcmp(w\_var[i].department,"Account")==0){

sum=sum+w\_var[i].salary;

if (max<w\_var[i].salary){

max=w\_var[i].salary;

max=w\_var[i].salary;

printf("\n%d\n",w\_var[i].worker\_id);

puts(w\_var[i].first\_name);

puts(w\_var[i].last\_name);

printf("%d\n",w\_var[i].salary);

printf("%d-%d-%d\n",w\_var[i].j\_d.year,w\_var[i].j\_d.month,w\_var[i].j\_d.date);

puts(w\_var[i].department);

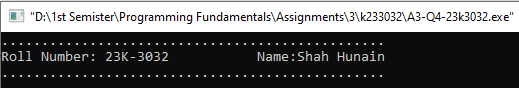
}

}

}

printf("Total salary in 'Account' department is: %d\n",sum);

}

**OUTPUT**

