**Assignment: 6**

**Exp 1.** A record contains the name of a cricketer, his age, the number of tests matches he has played, and the **average** runs he scored in each test match. Create an array of structures to hold records of 20 such cricketers and then write a program to read these records and arrange them in ascending order by runs, Use the qsort standard library function.

**Coding:**

#include<stdio.h>

struct cricketer{

   char name[50];

   int age;

   int match;

   float avrn;

};

int main() {

   int i,j,n;

   printf("enter number of crickters:");

   scanf("%d",&n);

 struct cricketer c[n],temp1;

   for(i=0;i<n;i++){

       printf("Enter data of cricketer \n");

      printf("Name: ");

      scanf("%s",c[i].name);

      printf("Age: ");

      scanf("%d",&c[i].age);

      printf("Matches: ");

      scanf("%d",&c[i].match);

      printf("Average runs: ");

      scanf("%f",&c[i].avrn);

   }for(i=0;i<n;i++) {

      for(j=i+1;j<n;j++) {

         if(c[i].avrn > c[j].avrn){

            temp1=c[i];

            c[i]=c[j];

            c[j]=temp1;

         }

      }

   }

   printf("Sorted records:\n");

   for(i=0;i<n;i++){

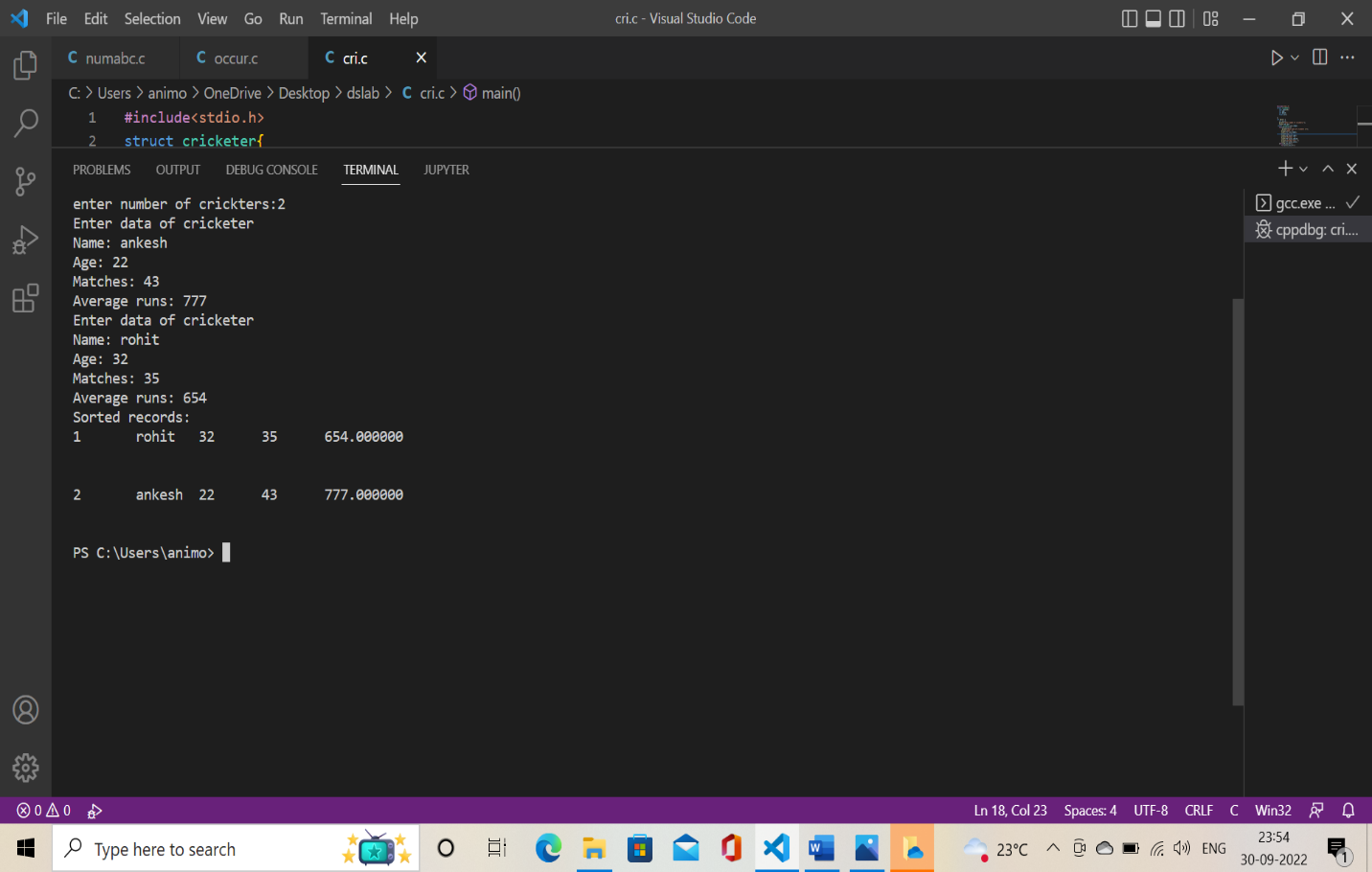
      printf("%d\t%s\t%d\t%d\t%f\n\n\n",i+1,c[i].name,c[i].age,c[i].match,c[i].avrn);

   }

   return 0;

}

**Output:**



**Exp 2.** Create a structure to specify data of customers in a bank. The data to be stored is Account number, Name, and Balance in the account. Assume a maximum of 200 customers in the bank.

1. Write a function to print the account number and name of each customer with a balance below Rs 100.
2. If a customer requests for withdrawal or deposit, the form contains the fields: Acct no, amount, code (1 for deposit and 0 for withdrawal) WAP to give a message " The balance is insufficient for the specified withdrawal", if on withdrawal the balance falls below Rs 100.

**Coding:**

**Output:**

**Exp 3.** WAP to count the number of occurrences of any two vowels in succession in a line of text. For example, in the following sentence "Please read this application and give me gratuity". such occurrences, ea, ea and ui.

**Coding:**

#include<stdio.h>

#include<strings.h>

#include<string.h>

int main()

{

    int a=0,i,len;

    char str[100];

    printf("Enter a String: ");

    scanf("%[^\n]%\*c",str);

    printf("%s\n",str);

    len=strlen(str);

    for(i=0;i<len;i++)

    {

        if((str[i]=='a'||str[i]=='e'||str[i]=='i'||str[i]=='o'||str[i]=='u'||str[i]=='A'||str[i]=='E'||str[i]=='I'||str[i]=='O'||str[i]=='U')&&(str[i+1]=='a'||str[i+1]=='e'||str[i+1]=='i'||str[i+1]=='o'||str[i+1]=='u'||str[i+1]=='A'||str[i+1]=='E'||str[i+1]=='I'||str[i+1]=='O'||str[i+1]=='U'))

        {

            printf("%c",str[i]);

            printf("%c ",str[i+1]);

            printf(" ");

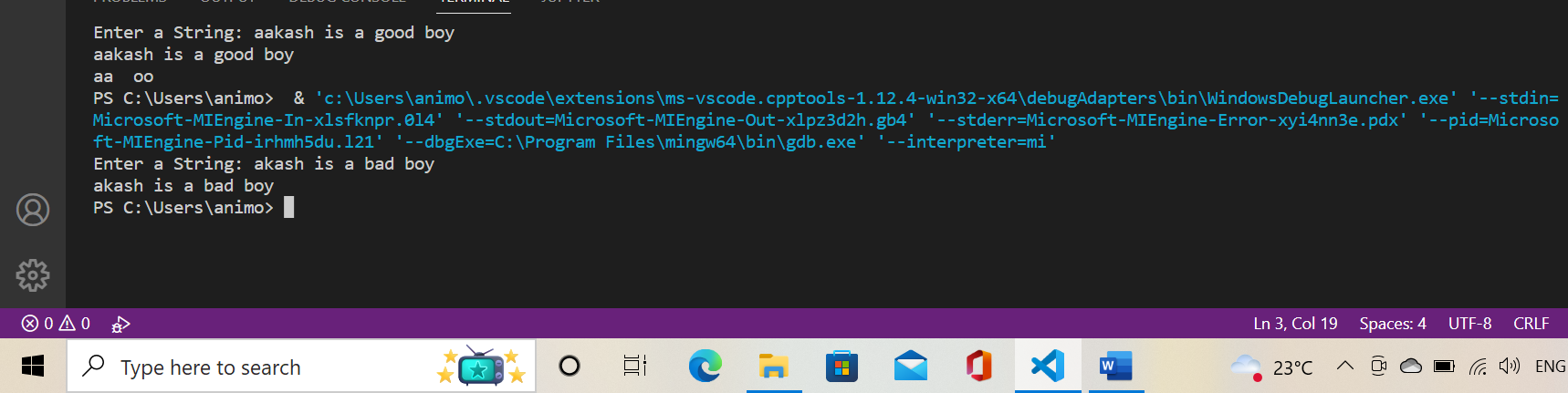
        }

    }

    return 0;

}

**Output:**



**Exp 4.** WAP to receive an integer and printout the number in words. For example, if the number is 5678, it should print Five thousand six hundred and seventy-eight.

**Coding:**

#include<stdio.h>

#include<string.h>

void numberInUnits(int num)

{

    switch(num)

    {

        case 1:

            printf("One ");

            break;

        case 2:

            printf("Two ");

            break;

        case 3:

            printf("Three ");

            break;

        case 4:

            printf("Four ");

            break;

        case 5:

            printf("Five ");

            break;

        case 6:

            printf("Six ");

            break;

        case 7:

            printf("Seven ");

            break;

        case 8:

            printf("Eight ");

            break;

        case 9:

            printf("Nine ");

            break;

    }

}

void numberInTens(int num)

{

    int tensDigit = num/10;

    int unitsDigit = num%10;

    if(tensDigit == 1)

    {

        switch(unitsDigit)

        {

            case 0:

                printf("Ten ");

                break;

            case 1:

                printf("Eleven ");

                break;

            case 2:

                printf("Twelve ");

                break;

            case 3:

                printf("Thirteen ");

                break;

            case 4:

                printf("Fourteen ");

                break;

            case 5:

                printf("Fifteen ");

                break;

            case 6:

                printf("Sixteen ");

                break;

            case 7:

                printf("Seventeen ");

                break;

            case 8:

                printf("Eighteen ");

                break;

            case 9:

                printf("Ninteen ");

                break;

        }

    }

    else

    {

        switch(tensDigit)

        {

            case 2:

                printf("Twenty ");

                break;

            case 3:

                printf("Thirty ");

                break;

            case 4:

                printf("Forty ");

                break;

            case 5:

                printf("Fifty ");

                break;

            case 6:

                printf("Sixty ");

                break;

            case 7:

                printf("Seventy ");

                break;

            case 8:

                printf("Eighty ");

                break;

            case 9:

                printf("Ninety ");

                break;

        }

        numberInUnits(unitsDigit);

    }

}

void numberInHundreds(int num)

{

    if(num/100)

    {

        numberInUnits(num/100);

        if(num%100) printf("hundred and ");

        else printf("hundred ");

        numberInTens(num%100);

    }

    else numberInTens(num%100);

}

void numberInThousands(int num)

{

    if(num/1000)

    {

        numberInUnits(num/1000);

        printf("thousand ");

        numberInHundreds(num%1000);

    }

    else numberInTens(num%1000);

}

void numberInTenThousands(int num)

{

    if(num/1000)

    {

        numberInTens(num/1000);

        printf("thousand ");

        numberInHundreds(num%1000);

    }

    else numberInHundreds(num%1000);

}

void numberInHundredThousands(int num)

{

    if(num/100000)

    {

        numberInHundreds(num/1000);

        printf("thousand ");

        numberInHundreds(num%1000);

    }

    else numberInTenThousands(num%10000);

}

void numberInMillions(int num)

{

    if(num/1000000)

    {

        numberInUnits(num/1000000);

        printf("million ");

        numberInHundredThousands(num%1000000);

    }

    else numberInHundredThousands(num%1000000);

}

void numberInTenMillions(int num)

{

    if(num/10000000)

    {

        numberInTens(num/1000000);

        printf("million ");

        numberInHundredThousands(num%1000000);

    }

    else numberInMillions(num%10000000);

}

void numberInHundredMillions(int num)

{

    if(num/100000000)

    {

        numberInHundreds(num/1000000);

        printf("million ");

        numberInHundredThousands(num%1000000);

    }

    else numberInTenMillions(num%100000000);

}

void numberInBillions(int num)

{

    if(num/1000000000)

    {

        numberInUnits(num/1000000000);

        printf("billion ");

        numberInHundredMillions(num%1000000000);

    }

    else numberInHundredMillions(num%1000000000);

}

void main()

{

    int i, j, n, temp, num[10];

    printf("Enter any number: ");

    scanf("%d", &n);

    temp = n;

    while(temp)

    {

        temp /= 10;

        i++;

    }

    switch(i)

        {

            case 1:

                numberInUnits(n);

                break;

            case 2:

                numberInTens(n);

                break;

            case 3:

                numberInHundreds(n);

                break;

            case 4:

                numberInThousands(n);

                break;

            case 5:

                numberInTenThousands(n);

                break;

            case 6:

                numberInHundredThousands(n);

                break;

            case 7:

                numberInMillions(n);

                break;

            case 8:

                numberInTenMillions(n);

                break;

            case 9:

                numberInHundredMillions(n);

                break;

            case 10:

                numberInBillions(n);

                break;

        }

}

**Output:**

