Terna Engineering College
Computer Engineering Department

Program: Sem V

Course: Microprocessor Lab

Faculty: ARATHI BOYANAPALLI

LAB Manual

PART A

(PART A: TO BE REFERRED BY STUDENTS)

Experiment No. 9

A.1 Aim:
Mixed language to separate EVEN and ODD number in the array

A.2 Prerequisite:
Basic knowledge of 8086 instruction set, interrupts and high-level language

A.3 Outcome:
After successful completion of this experiment, students will be able to

1. Use appropriate instructions to program microprocessor to perform various task.
2. Develop the program in assembly/ mixed language for Intel 8086 processor
3. Demonstrate the execution and debugging of assembly/ mixed language program

A.4 Theory:

Mixed-language programming allows you to combine the unique strengths of C++ with your assembly-language routines. There are some cases where you want to achieve things using inline assembly, such as improving speed, reducing memory needs and getting more efficiency. However, an inline assembler is not as powerful as TASM, it does not support macros or directives.

You can write small assembly language routines within your C or C++ code. These routines are compiled using the inline or embedded assembler of the TURBO compiler. However, there are a number of restrictions to the assembly language code you can write if you are using the inline or embedded assembler. These restrictions are such as instruction set and the number of registers can be used.

In c++, every integer will take 2 bytes therefore for accessing any element using from array assembly pointer should increment and decrement by 2.

This is an experiment in which even and odd numbers will be sorted by checking LSB of number. Another way to recognize even or odd numeric is to divide the number and checking reminder.

A.5 Algorithm

I. Accept an array of elements using cin or scanf functions.
II. For every element, check modulus of 2 After bringing the element into accumulator register(DIV instruction can be used) or check LSB of number (use shift instructions).
III. If the remainder is zero then it is even otherwise it odd number put into respective.
IV. Display the respective even and odd arrays.

PART B

(PART B: TO BE COMPLETED BY STUDENTS)

(Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded on the ERP or emailed to the concerned lab in charge faculties at the end of the practical in case the there is no ERP access available)

Roll No.: 50	Name: Amey Thakur
Class: TE-Comps B	Batch: B3
Date of Experiment: 08/09/2020	Date of Submission: 08/09/2020
Grade:	

B.1 Observations and learning:

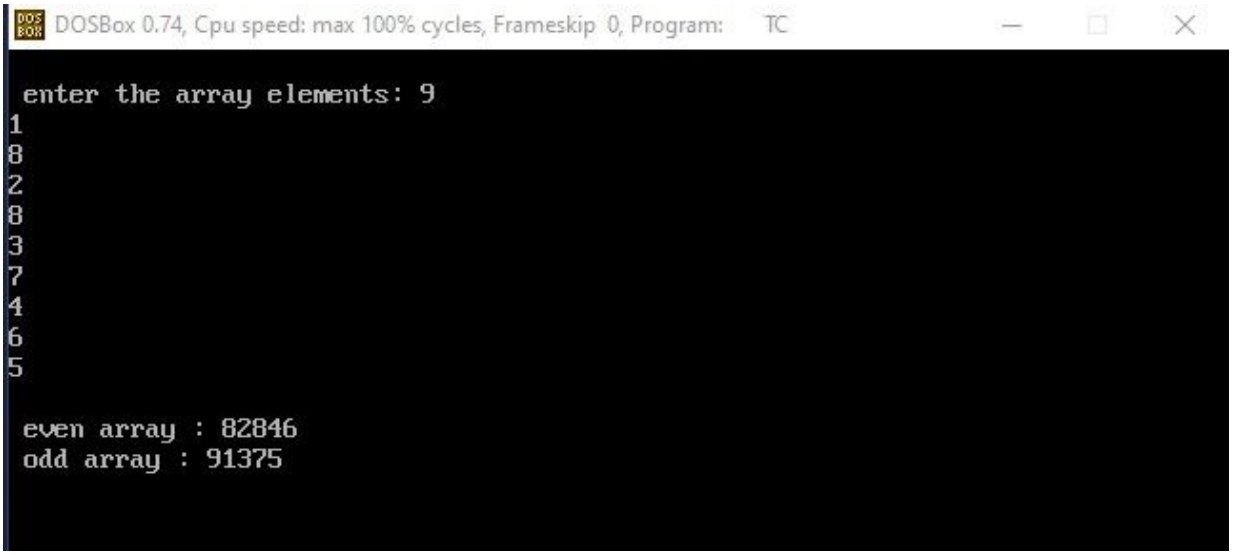
(Software Code written by a student and output of the program)

- **Mixed language to separate EVEN and ODD number in the array**

```
#include<conio.h>
#include<stdio.h>
#include<stdlib.h>
void main()
{
    int arr[10], ev[10], odd[10];
    int no;
    char rem, dv=2;
    int i, j=0, k=0, l1, l2;
    clrscr();
    printf("\n enter the array elements: ");
    for(i=0; i<10; i++)
        scanf("%d",&arr[i]);
    asm lea si, arr
    asm mov cx, 0AH
back: asm mov ax,[si]
    asm mov no, ax
    asm mov bl, dv
    asm div bl
    asm mov rem, ah
    if(rem==1)
    {
     odd[j]=no;
```

```
    j++;
    l1=j;
    }
    else
    {
        ev[k]=no;
        k++;
        l2=k;
    }
    asm inc si
    asm inc si
    asm dec cx
    asm jnz back
    printf("\n even array : ");
    for(i=0; i<l2; i++)
            printf("%d", ev[i]);
    printf("\n odd array : ");
    for(i=0; i<l1; i++)
            printf("%d", odd[i]);
    getch();
}
```

- **Output -**



B.2 Conclusion:

(Students must write the conclusion as per the attainment of individual outcome listed above and learning/observation noted in section B.1)

We have learned about Mixed language which can separate EVEN and ODD number in an array.

B.5 Question of Curiosity

Q1. Write mixed programming language to find the factorial of a number?

Ans:

```
#include<stdio.h>
#include<conio.h>
void main()
{
        int num;
        clrscr();
        printf("\n Enter the number : ");
        scanf("%d",&num);
        asm mov ax,1;
        asm mov bx,1;
        asm mov cx,num;
        come:
        asm mul bx;
        asm inc bx;
        asm dec cx;
        asm jnz come;
        printf("%d",_AX);
        getch();
}
```

Output -

Q2. Write a mixed programming language to check the given number is palindrome or not?

Ans:

```
#include<iostream.h>
#include<conio.h>
void main()
{
int nos[5],x,y;
printf("\n Enter 5 numbers: ");
int i,j;
for(i=0;i<5;i++)
{
cin>>nos[i];
}
```

```
for(i=0,j=4;i<3;i++,j--)
{
x=nos[i];
y=nos[j];
_asm{
mov ax,x
mov bx,y
cmp ax,bx
jnz no
}
}
cout<<"\n Array is Palindrome.";
goto end;
no:
cout<<"\n Array is not palindrome.";
end:
getch();
}
```

- **Output -**

➔ PALINDROME NUMBERS

➔ NON-PALINDROME NUMBERS