*A minor project report on*

computer organisation and architecture

*In the partial fulfillment of the requirements for*

*Bachelor of Technology in Computer Science & Engineering*

Submitted

By

R.SAI KIRAN (REG. NO: 171FA04427)

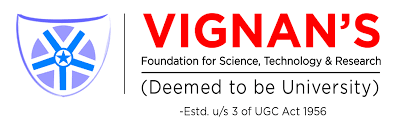
S.DEEPAK (REG. NO: 171FA04428)

P.MOHAN (REG. NO: 171FA04478)

*Under the guidance of*

Mrs.G.Parimala

(Assistant professor)



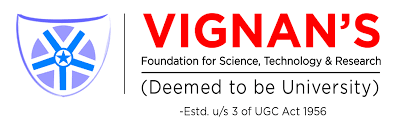
**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**Vignan’s Foundation for Science and Technology Research**

Deemed to be University

**Vadlamudi, Guntur - 522213, INDIA.**

**April 2019**



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEEING**

**CERTIFICATE**

This is to certify that the report entitled “**Assembly Language**” is submitted by **R.saikiran(171FA04427),S.deepak(171FA04428),P.mohan(171FA04478)”**in the partial fulfilment of course work of Assembly Language programming as a minor project, carried outin the department of CSE ,VFSTR deemed to be university.

|  |  |  |
| --- | --- | --- |
| **SIGNATURE** |  | **SIGNATURE** |
| **Dr.VenkatesuluDondeti,** |  | **Mrs.G.Parimala** |
| **HOD**  Professor |  | Assistant Professor  Department CSE |
| Department of CSE |  |  |

Submitted for the External Review held on 02-05-2019

**InternalExaminer External Examiner**

|  |  |
| --- | --- |
| **TABLE OF CONTENT** |  |
| Title | PageNo. |
| **ABSTRACT** | 4 |
| **ASSEMBLY LANGUAGE INTRODUCTION** | 5 |
| **HARDWARE AND SOFTWARE REQUIREMENTS** | 6 |
| **PROBLEM DEFINITION** | 7,8 |
| **SOURCE CODE** | 9-12 |
| **SCREEN SHOTS** | 13,14 |
| **REFERENCES** | 15 |

ABSTRACT

Assembly language is One of the language of computer.It is also called low level language.All programs we do in our daily life like c , c++,java etc.., languages are converted into assemble language by the use of compiler.By the above process we are making computer to understand our language what we are willing to do.ASCII stands for American Standard Code for Information Interchange. Computers can only understand numbers, so an ASCII code is the numerical representation of a character such as 'a' or '@' or an action of some sort. ASCII was developed a long time ago and now the non-printing characters are rarely used for their original purpose. ASCII was actually designed for use with teletypes and so the descriptions are somewhat obscureNotepad.exe creates ASCII text, or in MS Word you can save a file as 'text only'.

**INTRODUCTION**

* A programming language that uses symbolic names to represent operations, registers and memory locations.
* Slightly high level language.
* Better than machine language.
* Assembler translate assembly to machine code.
* Compiler translate high-level programs to machine code
* Either directly, or
* Indirectly via an assembler
* An assembler is a program that converts source code program written in assembly language into object files in machine language.
* Popular assemblers have emerged over the years for the Intel family processors. These include.....
* TASM(turbo assembler)
* NASM(net wide assembler)
* In the early days of programming, all programs were written in assembly language. Now, most programs are written in a high-level language such as c++ or C. Programmers still use assembly language when speed is essential or when they need to perform an operation that isn't possible in a high-level language. Machine code instructions are represented by *mnemonics*
* e.g., MOVE, ADD, SUB

**SOFTWARE REQUIREMENT**

Basic software requirement to develop programs :-

Language – assembly.

Operating system – windows 7 (or) higher .

Assembler – TASM

**HARDWARE REQUIREMENT**

Hard disk: 10 GB space

RAM: 256 MB

Processor: 32-bit processor intel

Minimum 256 MB Virtual Memory

1. **Program that will copy a string STRING\_1 into another string STRING\_2 using the instruction MOVSB and REP.**

PROBLEM DEFINITION :

* For this program we have read one string using a variable .
* **movsb** moves only a single byte from the source string to the destination
* To move the entire string, first initialize **cx** to the number *N* of bytes in the source string and execute **rep movsb**
* The **rep** prefix causes **movsb** to be executed *N* times
* After each **movsb**, **cx** is decremented until it becomes 0

**2.Program to print Lower Case Letters from A to Z**

PROGRAM DEFINITION :

* For developing this program we should know about ascii values and should use while loop
* The while loop is a if statement fallowed by the body of loop fallowed by unconditional jump to top of the loop.
* While loop is used to check the condition first if the condition is true then loop will be executed else loop will not be.
* cmp :- cmp instruction is used to compare destination operand to source operand.
* Note :-non-destructive subtraction of source from destination (destination operand is not changed)
* Syntax:- cmp destination,source
* jae :- jae is a jump based unsigned comparison , jae is nothing but jump if above or equal
* Jae is used when the condition is used to check variable is >= some other variable.
* Ex : if a>= b
* Jmp :- jmp is used to perform unconditional unconditional jump.
* Such an instruction transfers the flow of execution by changing the instruction pointer register.
* Syntax:- jmp position
* Inc :- it is a key word for incrementing the value.
* Syntax:-inc operand
* These are some that are used in the program

**3. to convert lower case into upper case and vise versa**

PROGRAM DEFINITION :

* This is a program to print lower case letters to upper case letters and also upper case letters to lower case letters in assembly language.
* In this program, the actual code of conversion of string to upper case (or) lower case is present in main() function. An array of char type is declared which will store the entered string by the user.
* Then, for loop is used to convert the string into upper case (or) lower case string and if block is used to check that if characters are in lower case (or) upper case then, convert them in upper case (or) lower case by subtracting 32 from their ASCII value.

SOURCE CODE :

.model small

.stack 100h

.data

str1 db "copy a string$"

msg db 'the copied string is: $'

str2 db 14 dup(?)

.code

main proc

mov ax,@data

mov ds,ax

mov es,ax

mov si,offset str1

mov di,offset str2

mov cx,14

cld

rep movsb

lea dx,msg

mov ah,9

int 21h

mov dl,offset str2

mov ah,9

int 21h

mov ah,4ch

int 21h

main endp

end main

SOURCE CODE :

.MODEL SMALL

.STACK 100H

.DATA

PROMPT DB 'The lower Case Letters from a to z are : $'

.CODE

MAIN PROC

MOV AX, @DATA

MOV DS, AX

LEA DX, PROMPT

MOV AH, 9

INT 21H

MOV CX, 26

MOV AH, 2

MOV DL, 97

@WHILE\_LOOP:

CMP CX, 0

JE @END\_LOOP

INT 21H

INC DL

DEC CX

JMP @WHILE\_LOOP

@END\_LOOP:

MOV AH, 4CH

INT 21H

MAIN ENDP

END MAIN

Source code:

DIS MACRO STR

MOV AH,09H

LEA DX,STR

INT 21H

ENDM

DATA SEGMENT

MSG1 DB "ENTER YOUR STRING : $"

MSG2 DB "CONVERTED STRING IS : $"

STR1 DB 20 DUP('$')

LINE DB 10,13,'$'

DATA ENDS

CODE SEGMENT

ASSUME DS:DATA,CS:CODE

START:

MOV AX,DATA

MOV DS,AX

DIS MSG1

MOV AH,0AH

LEA DX,STR1

INT 21H

DIS LINE

MOV CH,00

MOV CL,BYTE PTR[STR1+1]

LEA SI,STR1+2

L1: MOV AH,BYTE PTR[SI]

CMP AH,'A'

JL L4

CMP AH,'Z'

JG L2

ADD BYTE PTR[SI],32

JMP L3

L2:CMP AH,'a'

JL L4

CMP AH,'z'

JG L4

SUB BYTE PTR[SI],32

L3:INC SI

LOOP L1

DIS MSG2

DIS STR1+2

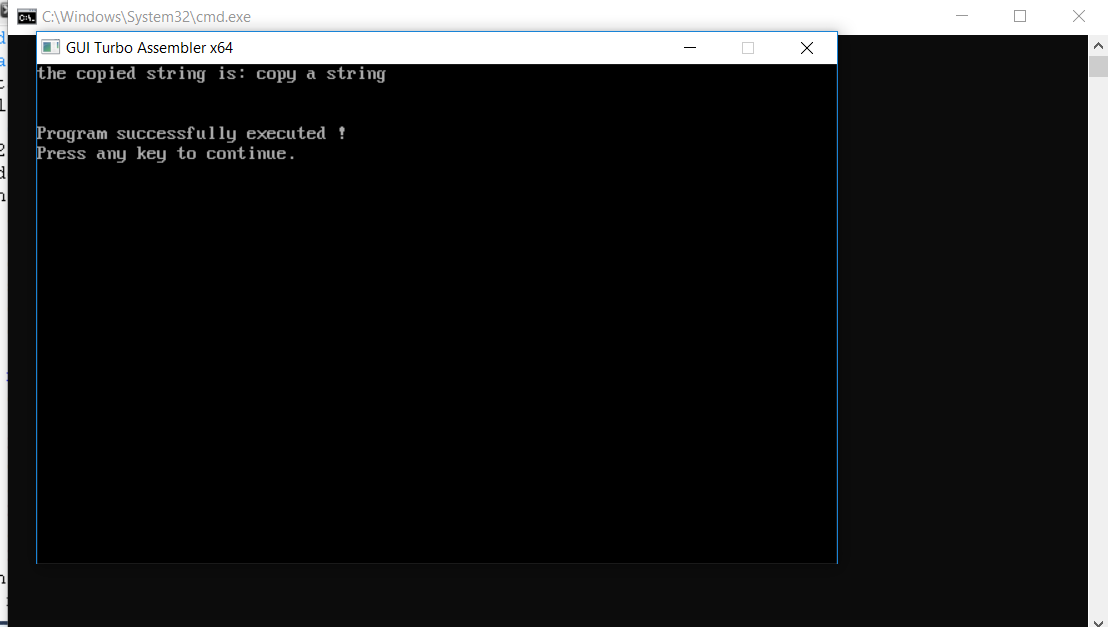
L4:MOV AH,4CH

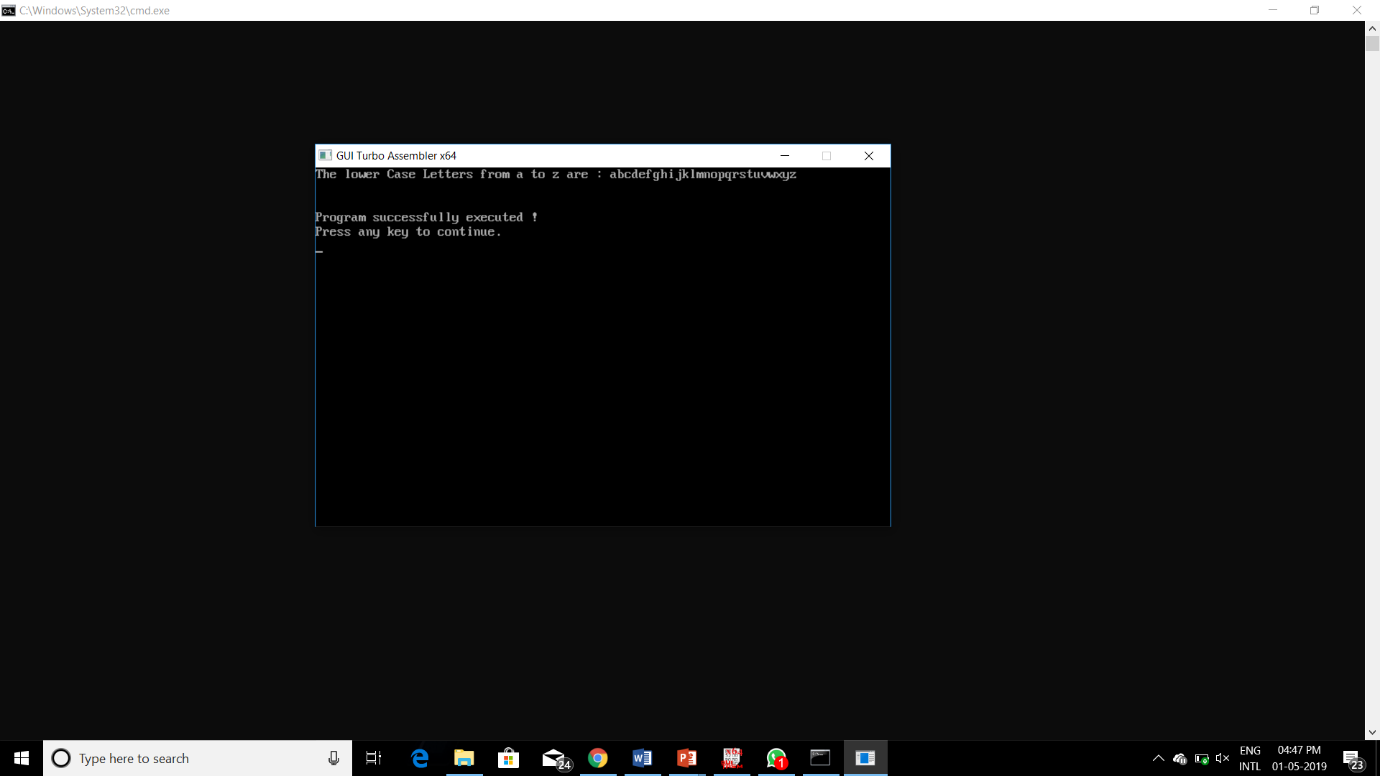
INT 21H

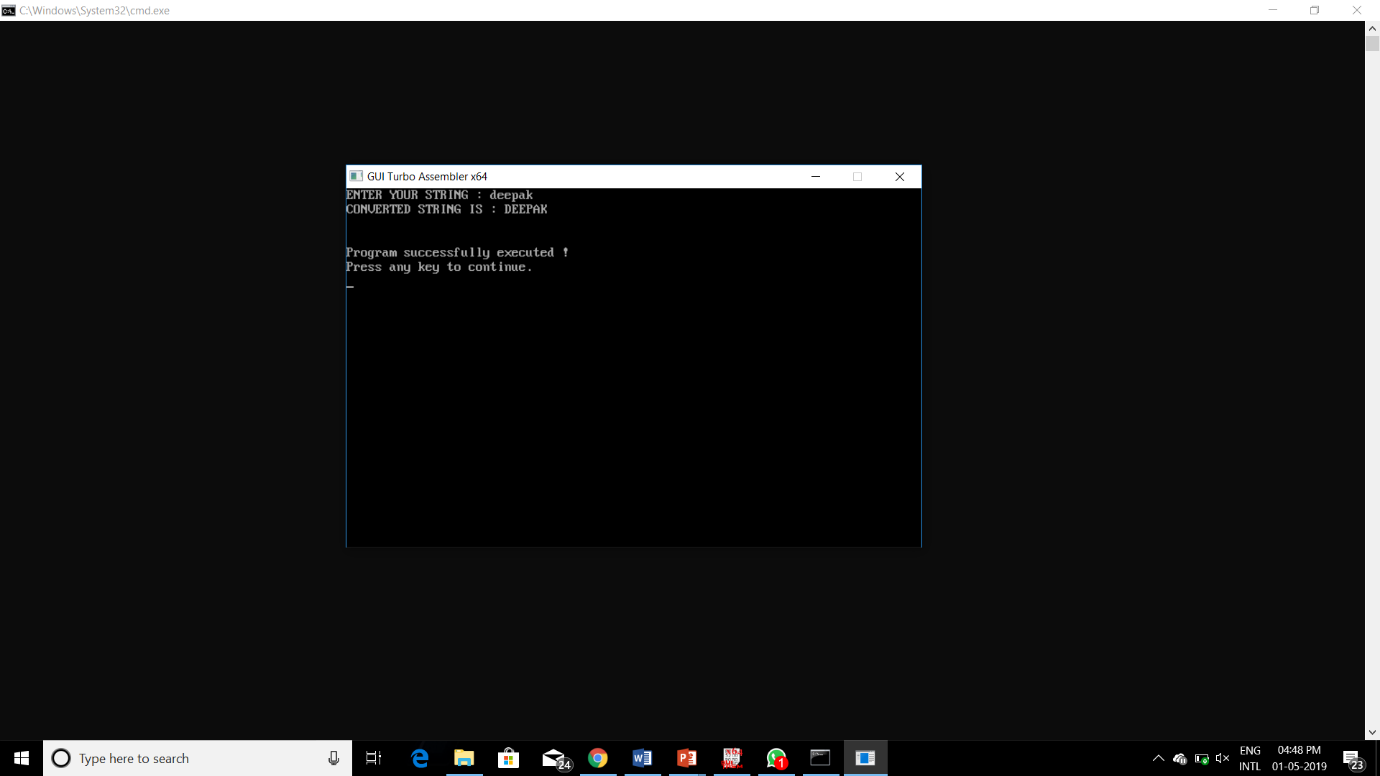
CODE ENDS

END START

SCREENSHOTS :







**References :**

* Internet
* You tube