Anjuman-I-Islam's

M.H. Saboo Siddik Polytechnic

8 , M.H. Saboo Siddik Polytechnic Road, Mumbai, 400008

SECOND YEAR DIPLOMA IN COMPUTER ENGINEERING

(2022-23)

PROJECT REPORT ON

FIBONACCI SERIES

By

210451- Abdurrahman Qureshi

210453- Ansari Saad

210460- Arya More

210463- Adnan Kazi

UNDER THE GUIDANCE OF

Mrs. Kousar Akumalla Ma'am

Maharashtra State Board of Technical Education (MS-BTE)

MAHARASHTRA STATE

BOARD OF TECHNICAL EDUCATION

Certificate

This is to certify that Mr. Abdurrahman Qureshi Roll no. 210451 of fourth semester of Diploma in Computer Engineering of institute M.H. Saboo Siddik Polytechnic(code: 0002) has completed microproject satisfactorily in the subject: MIC (22415) for the academic year 2022-23 as prescribed in the curriculum.

Place: Byculla, Mumbai Enrolment no:2100020112

Date: _____ Exam seat no:

Signature	Signature	Signature
Project guide	H. O. D	Principal

SEAL OF INSTITUTE

MAHARASHTRA STATE

BOARD OF TECHNICAL EDUCATION

Certificate

This is to certify that Mr. Ansari Saad Roll no. 210453 of fourth semester of Diploma in Computer Engineering of institute M.H. Saboo Siddik Polytechnic(code: 0002) has completed microproject satisfactorily in the subject: MIC (22415) for the academic year 2022-23 as prescribed in the curriculum.

Place: Byculla, Mumbai Enrolment no:2100020102

Date: _____ Exam seat no:

Signature	Signature	Signature
Project guide	H. O. D	Principal

SEAL OF INSTITUTE

MAHARASHTRA STATE

BOARD OF TECHNICAL EDUCATION

Certificate

This is to certify that Mr. Arya More Roll no. 210460 of fourth semester of Diploma in Computer Engineering of institute M.H. Saboo Siddik Polytechnic(code: 0002) has completed microproject satisfactorily in the subject: MIC (22415) for the academic year 2022-23 as prescribed in the curriculum.

Place: Byculla, Mumbai Enrolment no:2100020097

Date: _____ Exam seat no:

Signature	Signature	Signature
Project guide	H. O. D	Principal

SEAL OF INSTITUTE

MAHARASHTRA STATE

BOARD OF TECHNICAL EDUCATION

Certificate

This is to certify that Mr. Adnan Kazi Roll no. 210463 of fourth semester of Diploma in Computer Engineering of institute M.H. Saboo Siddik Polytechnic(code: 0002) has completed microproject satisfactorily in the subject: MIC (22415) for the academic year 2022-23 as prescribed in the curriculum.

Place: Byculla, Mumbai Enrolment no:2100020117

Date: _____ Exam seat no:

Signature	Signature	Signature
Project guide	H. O. D	Principal

SEAL OF INSTITUTE

Acknowledgment

We wish to express our profound gratitude to our guide Mrs. Kousar Akumalla Ma,am who guided us endlessly in the framing and completion of the micro project. She guided us on all the main points in that micro project. We are indebted to his/her constant encouragement, cooperation, and help. It was his/her enthusiastic support that helped us in overcoming various obstacles in the micro-project. We are also thankful to our Principal, HOD, faculty members and classmates of Computer Engineering department for extending their support and motivation in the completion of this micro-project.

Names of Team Members with Roll Nos.

1. Abdurrahman Qureshi – 210451
2. Ansari Saad – 210453
3. Arya More – 210460
4. Adnan Kazi – 210463

Annexure-I

Microproject proposal

Title of microproject: Fibonacci Series

I. Aims/Benefits of microproject

Microprocessor is the main component of the computer where 8086 is the base of all upward developed processors till current processors. This course will cover the basic of 8086 and it's procedures and macros.

II. Course outcomes addressed

▪ Analyze the functioning block of 8086 microprocessors.

▪ Write assembly language program for the given problem.

▪ Use instructions for different addressing modes.

▪ Develop an assembly language program using assembler.

▪ Develop assembly language program using procedures, macros, and modular programming approach.

III. Proposed methodology

1. To search the information about the project. (Collect relevant data from different sources i.e. books/internet and others through surveys/interviews etc.).

2. To collect all relevant content / materials to complete the project.

3. To prepare the report of micro project.

4. To prepare presentation.

5. To deliver presentation/ appear for viva-voice

- Discussion of the given topic among group members.

- Literature survey

- Submission of project proposal

- Analysis of data

- Work divided among group members

- Compilation of content

- Representation

- Editing the content as per the instructions

- Report Preparation

- Viva and presentation

Annexure-I

IV. Action Plan

Weeks	Details of activity	Planned start date	Planned finish date	Name of responsible team members
1& 2	Discussions & finalization of topics			
3	Preparation of abstract			
4	Literature review			
5	Submission of Micro-Project proposal(Annexure -I)			
6	Collection of information on given topic			
7	Collection of all relevant contents			
8	Discussion and submission of outline of the project			
9	Analysis/execution of collected data/information and Preparation of prototypes/drawings/charts/graphs/ tables/models/circuits/programs etc.			
10	Compilation of contents of project			
11	Compilation of weekly progress report			
12	Preparation of the project report (Annexure II)			
13	Viva Voce / Delivery of presentation.			

V. Resources required

Sr. no.	Name of resources	Specifications	Qty	Remarks
1.	Online	Learning resources and various websites	5 sites	
2.	Desktop	Microsoft word, Tools with internet facility.	1 for each	

Names of Team Members with Roll Nos.

1. 210451 Abdurrahman Qureshi
2. 210453 Saad Ansari
3. 210460 Arya More
4. 210463 Adnan Kazi

Approved by:
Sign of Faculty:
Name of faculty: Mrs. Kousar Akumalla

Annexure-II

Micro-Project Report

Title of Micro-project: Fibonacci Series

I. Rationale

Microprocessor is the main component of the computer where 8086 is the base of all upward developed processors till current processors. This course will cover the basic of 8086 and it's procedures and macros

II. Aims/Benefits of microproject

The microprocessor is the central unit of a computer system that performs arithmetic and logic operations, which generally include adding, subtracting, transferring numbers from one area to another, and comparing two numbers. It's often known simply as a processor, a central processing unit, or as a logic chip.

III. Course outcomes achieved

✓ Write assembly language program for the given problem.
✓ Use instructions for different addressing modes.
✓ Develop an assembly language program using assembler.
✓ Develop assembly language program using procedures

IV. Literature Review

A microprocessor is a computer processor that incorporates the functions of a central processing unit on a single integrated circuit(IC) or sometimes up to 8 integrated circuits. The microprocessor is a multipurpose , clock driven , register based , digital integrated has accepts binary data as input, process it according to instructions stored in its memory and provides result (also in binary form) as output. Microprocessor contains both logic and sequential digital logic. Microprocessor operates on numbers and symbols represented in the binary number system. Capacity have since rendered other forms of computers almost completely GENERAL PURPOSE REGISTERS in 8086 MICRO-PROCESSOR.

GENERAL PURPOSE REGISTERS are used to store temporary data within the microprocessor . There are 8 general purpose registers in 8086 MICRO-PROCESSOR.

• AX –

This is the accumulator. It is of 16 bits and is divided into 8 bit registers AH and AL to also perform 8-big instructions.

Example:-

ADD AX, AX (AX=AX+AX)

• BX –

This is the base register. It is of 16 bits and is divided into 8 bits BH and BL to also perform 8-bit instructions. It is used to store the value of the offset.

EXAMPLE:-

Mov BL, [500] (BL=500H)

• CX –

This is the counter register. It is of 16 bit and is divided into 8 bits CL and CH to also perform 8-bit instructions. It is used in looping and rotation.

• DX –

This is the data register. It is of 16 bit and is divided into 8 bits DX and DH to also perform 8-bit instructions. It is used in multiplication an input/output port addressing.

EXAMPLE:-

Mul BX (DX, AX =AX*BX)

1. SP –

This is the stack pointer . It is of 16-bit. It points to the topmost item of the stack. If the stack is empty the stack pointer will be (FFFE)H. Its offset address relative to stack segment.

2. BP -

This is the base pointer. It is of 16-bit. It is primary used in accessing parameters passed by the stack. Its offset address is relative to stack address.

3. SI –

This is the source index register. It is of 16 bit It is used in the pointer addressing data and as a source in some string related operations. Its offset is relative to data segment.

4. DI –

This is the destination index register. It is of 16-bits. It is used in the pointer addressing of data and as a destination in some string related operations. Its offset is relative to extra segment.

5. DATA TRANSFER INSTRUCTIONS:-

The data transfer instruction is used to transfer the data from one location to another. This transfer of data can be either from register to register , register to memory or memory to register. It is important to note here that memory to memory transfer of data directly is not possible.

• Following are some instructions that are used for data transfer purpose :

1.MOV
2.PUSH
3.POP
4.XCHG
5.LAHF
6.SAHF
7.IN
8.OUT
9.LDS
10.LES

1)MOV –

This instruction simply copies the data from the source to the destination.

Syntax:- MOV destination, source

Example:- MOV al, bl

2)PUSH –

This instruction is used to push data into the stack.

Syntax:- PUSH source

Example:- PUSH CX

Working:- SP <- SP – 1
[SP] <- CH
SP <- SP – 1
3)POP –
This instruction is used to get the data from the stack.
Syntax:- POP destination
Example:- POP CX
Working:- CL <- [SP]
4)XCHG –
It exchanges the contents of the source and the destination.
Syntax:- XCHG destination, source
Example:- XCHG BL, AL
5) LAHF –
It stands for 'Load AH from flag register'. This instruction will therefore load the AH register with the content of lower byte of the flag register.
Syntax:- LAHF
Working:- AH<- lower byte of the flag register.
6) SAHF –
It stands for 'Source AH from flag register'. This instruction stores the content of AH register to the lower.
7)LDS –
This instruction will load the register that is defined in the instruction and the data segment(DS) from the source.
Syntax:- LDS destination , source
Example:- LDS BX , [SI]
Working:- BL<-[SI]
BH<-[SI + 1]
8) LES –
The working and syntax of this instruction is the same as the LDS. The difference is only that instead of data segment register(DS) , Extra segment register (ES) is used.
Byte of flag register
Syntax:- SAHF
Lower byte of flag register<-AH

9) IN –
This instruction is used to transfer data from the input unit to accumulator.
Syntax:- IN accumulator, port address
Working:- The content from the input whose address in mentioned. The instruction is transferred to the accumulator which is the AX register.
10)OUT
This instruction is used to transfer data from accumulator to the output unit.
Syntax:OUT port address,accumulator
Working:the content from the accumulator which is the AX registeris transferred to the output unit whose address in mentioned in the instruction.
Example:OUT 1326H,AL OUT ,DX,AX
String Manipulation instruction:
➢ String is a group of bytes/words and their memorybis always allocated in a sequential order.string is either referred as byte string orword string.here we will see some instructions which are used to manipulate the string related operations.
The 8086 string instructions are namely.
➢ REP/REPE/REPZ/REPNE/REPNZ
➢ MOVS/MOVSB/MOVSW
➢ CMPS/CMPSB/CMPSW
➢ SCAS/SCASB/SCASW
➢ LODS/LODSB/LODSW
➢ STOS/STOSB/STOSW
➢ MOVS/MOVSB/MOVSW
MOVS/MOVSB/MOVSW instruction:
MOVS:Move string.
MOVSB:move string byte.
MOVSW:move string word.

1. MOVS:
Movs instruction is used for moving a string of ASCII characters from one place in the memory to another place.
2. MOVSB:
Movsb instruction will perform all the actions in repeat unit loop.the movsb instruction will copy a byte from the location pointed to by the direct index register.
3. MOVSW:
The movsw instruction can be used to move a string of words depending on the state of the direction flag.SI and DI will automatically be incremented or decremented by 2 after each word move.

CMPS/CMPSB/CMPSW Instruction:

CMPS:compare string
CMPSB:compare string byte.
CMPSW:Compare string word.
A 8086 string instruction is a series of the same type of data items in sequential memory location.the CMPS instruction can be used to compare a byte in one string with a byte in another string or to compare a word in one string wuth a word in another string.SI is used to hold the offset of a byte or word in the source string and DI is used to hold the offset of a byte or a word in the other string.the comparison is done by subtracting the byte or word pointed to by DI from the byte or word pointed to by SI.the AF,CF,OF,PF,SF,and ZF flag are affected by the comparison.but neither operand is affected.

Example of CMPS/CMPSB/CMPSW:

After the comparison SI and DI will be automatically incremented or decremented according to direction flag to point to the next element in the two string(if DF=0,SI and DI) CX function as a counter which is decremented after each comparison.this will go on until CX=0

SCAS/SCASB/SCASW instruction:

SCAS compares a string byte with a byte in AL or a string word with word in AX.the instruction affects the flags,but it does not

change either the operand in AL (AX) or the operand in the 8086 string instructions.the string to be ‘scanned must be in the extra segment and DI must contain the offset of the byte or the word to be compared.

Example of SCAS/SCASB/SCASW:

SCASB says compare 8086 string instruction as bytes and SCASW says compare string as words.

LODS/LODSB/LODSW Instruction:

This instruction copies a byte from a string location pointed to by SI to AL,or a word from a string location pointed to by SI to AX.lods does not affect any flag .lodsb copies byte and lodsw copies a word.

STOS/STOSB/STOSW Instruction:

The STOS instruction copies a byte from AL or a word form AX to a memory location in the extra segment.DI is used to hold the offset of the memory location in the extra segment .after the copy ,DI is automatically incremented or decremented to print to the next string element in memory.if the direction flag,DF,is cleared ,then DI will automatically be incremented by one for a byte string or incremented by two for a word 8086 string instruction .if the direction flag is set,DI will be automatically decremented by one for a byte string or decremented by two for a word string.STOS does not affect any falgs.STOSB copies byte and STOSW copies a word.

Procedures in assembly language:

Procedures or subroutines are very important in assembly language as the assembly language programs are tend to be in large in size. Procedures are identified by a name, following this name the body of the procedure is described which performs a well defined job.End of the procedure is indicated by a return statement.

Syntax:

Following is the syntax to define a procedure -

```
proc_name
Procedure body
ret
```

The procedure is called from the another function by using CALL instruction. The CALL instruction should have the name of the called procedure as an argument as shown below -
CALL proc_name
The called procedure returns the control to the calling procedure by using the RET instruction.

Procedure:

A procedure is a group of instruction that usually performs a one task.It is a reusable section of the software program which is stored in memory once but can be used as often as necessary.
A procedure can be of two types:
1)Near procedure
2)Far procedure
ENDP Directive:
This directive is used along with the name of the procedure to indicate the end of a procedure to the assembler. The PROC and ENDP Directive are used to bracket a procedure.

CALL instruction:

The CALL instruction is used to transfer execution to a procedure. It performs two operations, When it executes , first it stores the address of instruction after the CALL instruction on the stack.second it changes the content of IP register in case of Near CALL and changes the content of IP register and CS register in case of FAR CALL.
There are two types of CALL.
1) Near call or intra segment call.
2) Far call or inter segment call.

MACROS:

Macros can be used to name a group of instructions being used several times in a program. MACROS are defined at the start of a program. Each time the macro name appears in a program, the assembler replaces it with the group of instructions defined as that macro at the start of the program- Replacing a macro name is also known as macro expansion.

SYNTAX:
Macro_name MACRO
--
--
--
ENDM
Advantages of Macro:
1. Repeated small groups of instructions replaced by one macro
2. Errors in macros are fixed only once, in the definition
3. Duplication of effort is reduced
Disadvantages of Macro:

1. Large programs produce greater code size than procedures

V. Actual Methodology Followed

We were assigned with the microproject topic and time was assigned to us to complete the project in 11 weeks. All team members worked together in these 11 weeks together in order to complete this microproject. Data was collected according to our topic. Presentation was prepared, technical report was prepared and then we prepared ourselves to deliver the presentation.

VI. Actual resources used

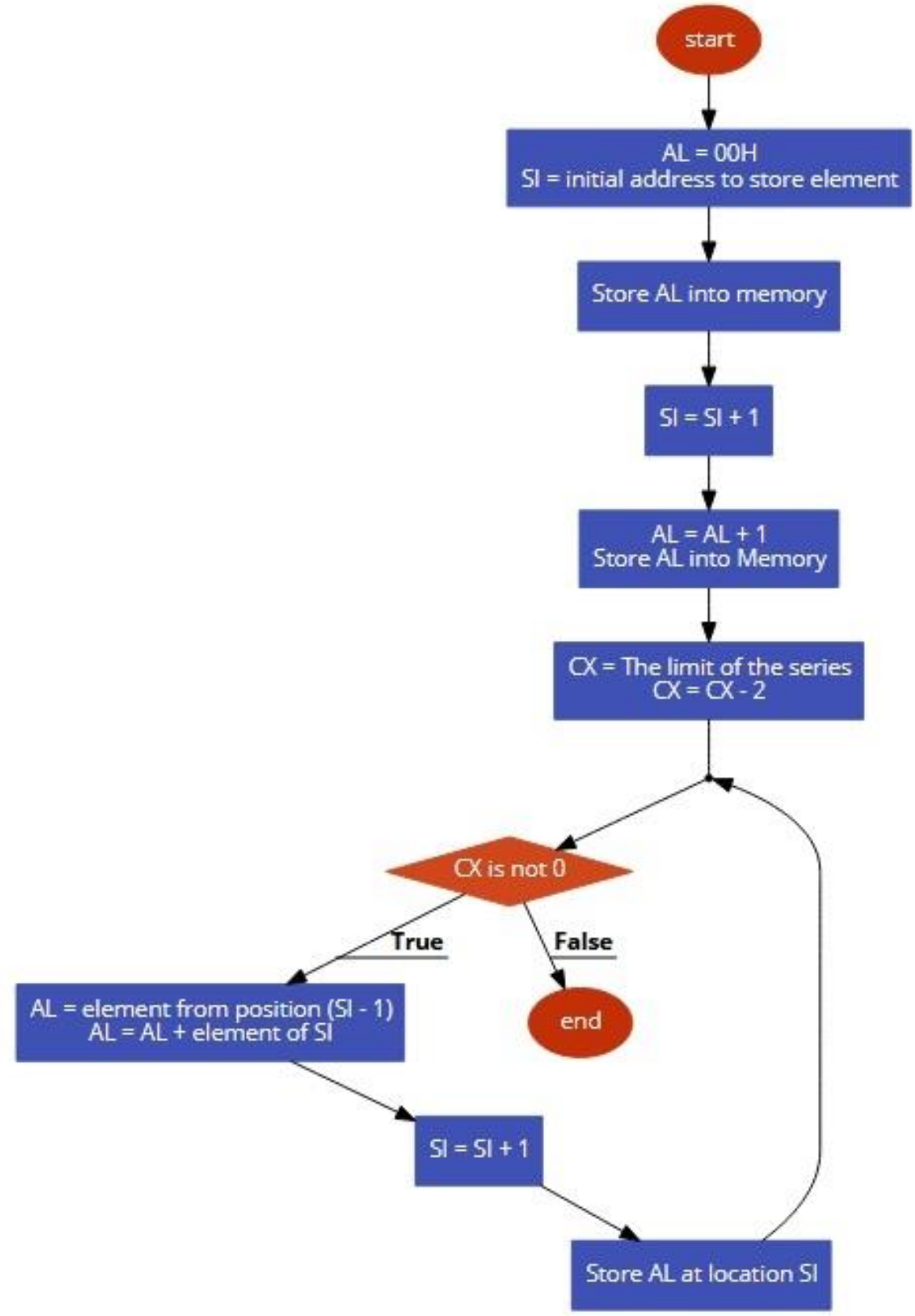
Sr. no.	Name of resources	Specifications	Qty	Remarks
1.	Computer	i3 – i5	1 for each	
2.	Windows	10	1 for each	
3.	Editor	MS – DOS	1 for each	
4.	Assembler	TASM	1 for each	
5.	Linker	TLINK	1 for each	
6.	Debugger	TD	1 for each	

VII. Outputs of Microproject

❖ ALGORITHM FOR FIBONACCI SERIES IN 8086

STEP 1: Move the value stored at offset 00H into CX(this will act as the counter), and decrement it by 2 (because we need to explicitly load the first 2 elements of the sequence)
STEP 2: Move 00H into AL
STEP 3: Move AL into [SI]
STEP 4: Increment both AL and SI by 1, and store AL's value in [SI] (with this, we have loaded the first 2 elements of the sequence into the memory)
STEP 5: Move [SI-1]th value into AL
STEP 6: Move [SI]th value into AH
STEP 7: Move 00H into BH
STEP 8: Add BH and AH (result stored in BH)
STEP 9: Add BH again with AL
STEP 10: Increment SI by 1
STEP 11: Store BH into [SI]
STEP 12: Loop back to Step 5 till counter becomes 0
STEP 13: Stop

❖ **FLOWCHART**

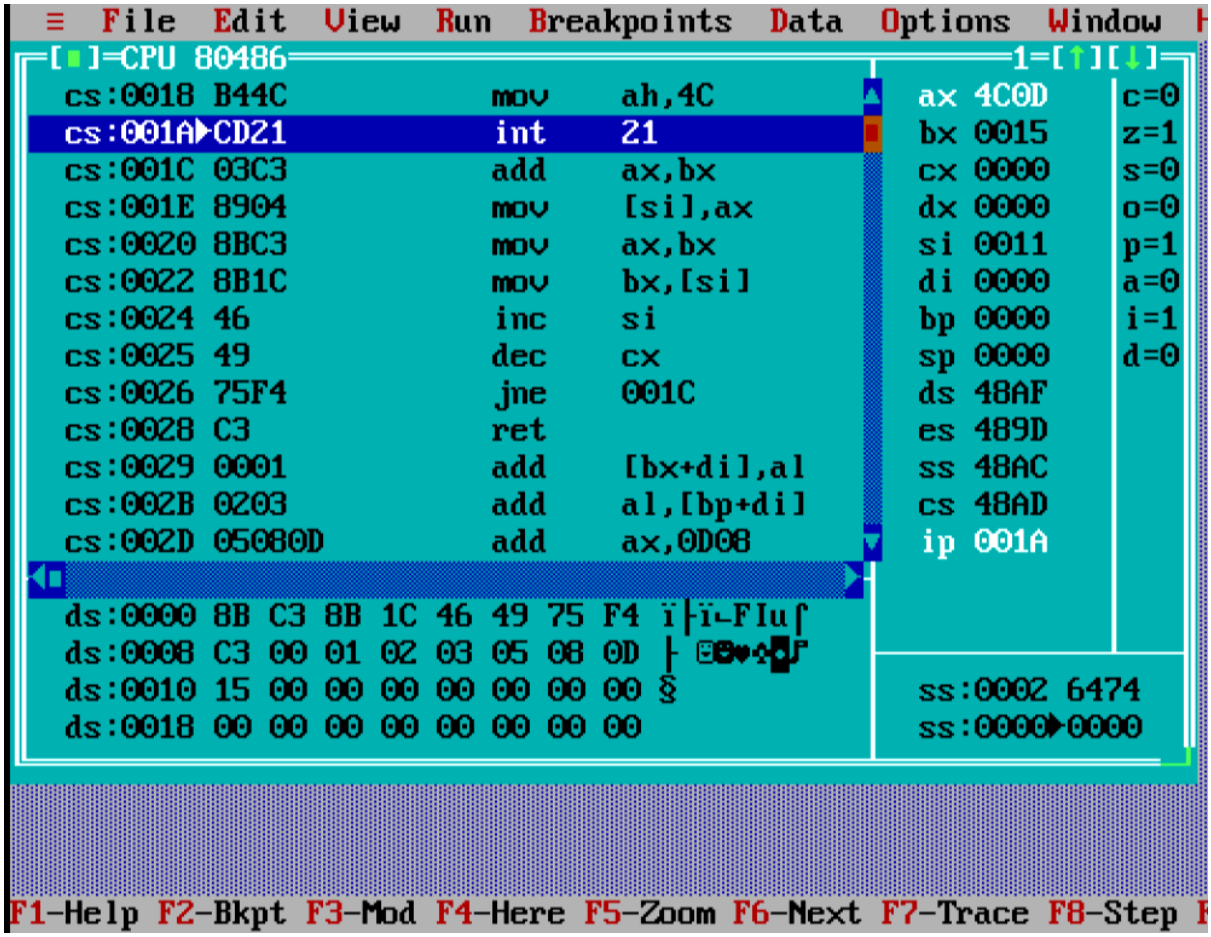

start
AL = 00H
SI = initial address to store element
Store AL into memory
SI = SI + 1
AL = AL + 1
Store AL into Memory
CX = The limit of the series
CX = CX - 2
CX is not 0
True
False
AL = element from position (SI - 1)
AL = AL + element of SI
end
SI = SI + 1
Store AL at location SI

❖ **PROGRAM**

```
.MODEL SMALL
.DATA
RES DB ?
CNT DB 07H      ; Initialize the counter for the no of Fibonacci
No needed
.CODE
MOV AX,@DATA
MOV DS,AX
LEA SI,RES
MOV CX,0000H
MOV CL,CNT      ; Load the count value for CL for looping
MOV AX,00H      ; Default No
MOV BX,01H      ; Default No
call fibo
MOV AH, 4CH
INT 21H

;Fibonacci Part
fibo PROC
L1:ADD AX,BX
MOV [SI],AX
MOV AX,BX
MOV BX,[SI]
INC SI
DEC CX
JNZ L1
ret
fibo endp

ends
end
```



VIII. Skills developed/ Learning outcomes

- Derive: Derive different possible solutions creatively.
- Data Collection: Collect relevant data from different sources (books/the internet/the market/suppliers/experts and others through surveys/interviews)
- Designing- Designing microproject with minimum required resources and at low cost.
- Teamwork- Learning to work in team and boost individual confidence.
- Time management- Completion of microproject as scheduled.
- Technical writing- Preparing a report of proposed plan and report.
- Presentation and communication skills: Giving working model presentation of the micro project.
- Confidence: Confidently, answer the questions asked about the project.

- Efficiently gathering details from research papers.
- Writing an assembly language program using procedure

IX. Applications of this microproject

1) Assembly language is used to directly manipulate hardware, access specialized processor instructions, or evaluate critical performance issues.

2) These languages are also used to leverage their speed advantage over high level languages for time-sensitive activities such as high frequency trading

Micro Project Evaluation Sheet

Name:Abdurrahman Qureshi Enrollment no: 2100020112
Name of programme: Computer Engg Semester: 4th
Course title: Microprocessors Code: 22415
Title of microproject: Fibonacci Series
Course outcomes achieved:
i. Write assembly language program for the given problem.
ii. Use instructions for different addressing modes.
iii. Develop an assembly language program using assembler.
iv. Develop assembly language program using procedures

Sr. No.	Characteristics to be assessed	Poor (Marks 1 - 3)	Average (Marks 4 - 5)	Good (Marks 6 - 8)	Excellent (Marks 9- 10)	Sub Total
(A) Process and Product Assessment (Convert above total marks out of 6 Marks)						
1	Relevance to the course					
2	Literature Review/information collection					
3	Completion of the Target as per project proposal					
4	Analysis of Data and representation					
5	Quality of Prototype/Model					
6	Report Preparation					
(B) Individual Presentation/ Viva (Convert above total marks out of 4 Marks)						
7	Presentation					
8	Viva					

(A) Process and Product Assessment (6 marks)	(B) Individual Presentation & viva (4 marks)	Total Marks 10

Comments/Suggestions about teamwork/leadership/interpersonal communication

_____
_____

Name and designation of teacher: _____

Dated signature: _____

Micro Project Evaluation Sheet

Name:Ansari Saad Enrollment no: 2100020102

Name of programme: Computer Engg Semester: 4th

Course title: Microprocessors Code: 22415

Title of microproject: Fibonacci Series

Course outcomes achieved:

i. Write assembly language program for the given problem.

ii. Use instructions for different addressing modes.

iii. Develop an assembly language program using assembler.

iv. Develop assembly language program using procedures

Sr. No.	Characteristics to be assessed	Poor (Marks 1 - 3)	Average (Marks 4 - 5)	Good (Marks 6 - 8)	Excellent (Marks 9- 10)	Sub Total
(A) Process and Product Assessment (Convert above total marks out of 6 Marks)						
1	Relevance to the course					
2	Literature Review/information collection					
3	Completion of the Target as per project proposal					
4	Analysis of Data and representation					
5	Quality of Prototype/Model					
6	Report Preparation					
(B) Individual Presentation/ Viva (Convert above total marks out of 4 Marks)						
7	Presentation					
8	Viva					

(A) Process and Product Assessment (6 marks)	(B) Individual Presentation & viva (4 marks)	Total Marks 10

Comments/Suggestions about teamwork/leadership/interpersonal communication

_____

_____

Name and designation of teacher: _____

Dated signature: _____

Micro Project Evaluation Sheet

Name:Arya More Enrollment no: 2100020097

Name of programme: Computer Engg Semester: 4th

Course title: Microprocessors Code: 22415

Title of microproject: Fibonacci Series

Course outcomes achieved:

i. Write assembly language program for the given problem.

ii. Use instructions for different addressing modes.

iii. Develop an assembly language program using assembler.

iv. Develop assembly language program using procedures

Sr. No.	Characteristics to be assessed	Poor (Marks 1 - 3)	Average (Marks 4 - 5)	Good (Marks 6 - 8)	Excellent (Marks 9- 10)	Sub Total
(A) Process and Product Assessment (Convert above total marks out of 6 Marks)						
1	Relevance to the course					
2	Literature Review/information collection					
3	Completion of the Target as per project proposal					
4	Analysis of Data and representation					
5	Quality of Prototype/Model					
6	Report Preparation					
(B) Individual Presentation/ Viva (Convert above total marks out of 4 Marks)						
7	Presentation					
8	Viva					

(A) Process and Product Assessment (6 marks)	(B) Individual Presentation & viva (4 marks)	Total Marks 10

Comments/Suggestions about teamwork/leadership/interpersonal communication

_____

_____

Name and designation of teacher: _____

Dated signature: _____

Micro Project Evaluation Sheet

Name:Adnan Kazi Enrollment no: 2100020117

Name of programme: Computer Engg Semester: 4th

Course title: Microprocessors Code: 22415

Title of microproject: Fibonacci Series

Course outcomes achieved:

i. Write assembly language program for the given problem.

ii. Use instructions for different addressing modes.

iii. Develop an assembly language program using assembler.

iv. Develop assembly language program using procedures

Sr. No.	Characteristics to be assessed	Poor (Marks 1 - 3)	Average (Marks 4 - 5)	Good (Marks 6 - 8)	Excellent (Marks 9- 10)	Sub Total
(A) Process and Product Assessment (Convert above total marks out of 6 Marks)						
1	Relevance to the course					
2	Literature Review/information collection					
3	Completion of the Target as per project proposal					
4	Analysis of Data and representation					
5	Quality of Prototype/Model					
6	Report Preparation					
(B) Individual Presentation/ Viva (Convert above total marks out of 4 Marks)						
7	Presentation					
8	Viva					

(A) Process and Product Assessment (6 marks)	(B) Individual Presentation & viva (4 marks)	Total Marks 10

Comments/Suggestions about teamwork/leadership/interpersonal communication

_____

_____

Name and designation of teacher: _____

Dated signature: _____