ALP-PGM 1a

WRITE AN ALP TO SEARCH A KEY ELEMENT IN A LIST OF 16 BIT N NUMBERS USING BINARY SEARCH ALGORITHM.

```
.MODEL SMALL

.DATA

A DB 50D,40D,30D,20D,10D

LEN DW $-A

M1 DB 10,13,'KEY IS FOUND$'

M2 DB 10,13,'KEY IS NOT FOUND$'

KEY DB 40D

.CODE

MOV AX,@DATA

MOV DS,AX

MOV BX,LEN

DEC BX

N2: MOV CX,BX

LEA SI,A

N1: MOV AL,[SI]

INC SI

CMP AL,[SI]

JBE NOSWAP
```

```
XCHG AL,[SI]

MOV [SI-1],AL

NOSWAP:LOOP N1

DEC BX

JNZ N2

LEA SI,A

MOV DI,SI

ADD DI,LEN

DEC DI

L1: CMP SI,DI

JA NF

MOV BX,SI

ADD BX,DI

SHR BX,01H

MOV AL,KEY

CMP AL,[BX]

JZ FOUND

JA NLOW

MOV DI,BX

DEC DI

JMP L1

NLOW:INC SI
```

```
MOV SI,BX

JMP L1

FOUND: LEA DX,M1

JMP FINAL

NF: LEA DX,M2

FINAL: MOV AH,09H

INT 21H

MOV AH,4CH

INT 21H

END
```

```
M:\>link bin.asm

Microsoft (R) Overlay Linker  Version 3.64
Copyright (C) Microsoft Corp 1983-1988.  All rights reserved.

Run File [BIN.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
BIN.ASM : fatal error L1101: invalid object module
 pos: 1 Record type: 573E

M:\>link bin.obj

Microsoft (R) Overlay Linker  Version 3.64
Copyright (C) Microsoft Corp 1983-1988.  All rights reserved.

Run File [BIN.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
LINK : warning L4021: no stack segment

M:\>bin

key is found
M:\>_
```

ALP-PGM 2a

READ AN ALPHANUMERIC CHARACTER AND DISPLAYS ITS EQUIVALENT ASCII CODE AT THE CENTER OF THE SCREEN.

```
.MODEL SMALL

.DATA

M1 DB 10,13,"ENTER THE CHARACTER  $"

M2 DB  "THE ASCII VALUE IS  $"

.CODE

MOV AX,@DATA

MOV DS,AX

MOV AH,06H

MOV AL,0D

MOV CX,00H

MOV BH,0CEH

MOV  DH,24D

MOV DL,79D

INT 10H

LEA DX,M1

MOV AH,09H

INT 21H

MOV AH,01H
```

```
INT 21H

MOV AH,00H

MOV BX,10D

MOV CX,00H

L1:MOV DX,00H

DIV BX

PUSH DX

INC CX

CMP AX,00H

JNZ  L1

MOV AH,2H

MOV BH,0H

MOV DH,12D

MOV DL,39D

INT 10H

LEA DX,M2

MOV AH,09H

INT 21H

L2: POP DX

ADD DL,30H

MOV AH,02H

INT 21H
```

```
DEC CX

JNZ L2

MOV AH,4CH

INT 21H

END
```

ALP-PGM 3a

WRITE AN ALP TO READ AN 8 BIT NUMBER FROM THE KEYBOARD AND CHECK WHETHER IT IS A PRIME NUMBER OR NOT.

```
.MODEL SMALL

.DATA

M1 DB 10,13,'ENTER A 8-BIT NUMBER$'

N1 DB ?

M2 DB 10,13,'THE NUMBER IS PRIME$'

M3 DB 10,13,'THE NUMBER IS NOT PRIME$'

.CODE

MOV AX,@DATA

MOV DS,AX

LEA DX,M1

MOV AH,09H

INT 21H

MOV AH,01H

INT 21H

SUB AL,30H

MOV BL,AL

MOV AH,01H
```

```
INT 21H

SUB AL,30H

MOV AH,BL

AAD

MOV CL,AL

MOV N1,AL

MOV BL,02H

DEC CL

L1:CMP BL,CL

JZ P

MOV AL,N1

MOV AH,00H

DIV BL

CMP AH,00H

JZ NP

INC BL

JMP L1

P:LEA DX,M2

JMP E

NP:LEA DX,M3

E: MOV AH,09H

INT 21H
```

MOV AH,4CH

INT 21H

END

```
M:\>link prime.obj

Microsoft (R) Overlay Linker  Version 3.64
Copyright (C) Microsoft Corp 1983-1988.  All rights reserved.

Run File [PRIME.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
LINK : warning L4021: no stack segment

M:\>prime

enter a 8-bit number21
the number is not prime
M:\>

M:\>prime

enter a 8-bit number11
the number is prime
M:\>prime

enter a 8-bit number83
the number is prime
M:\>
```

ALP-PGM 4a

WRITE AN ALP TO READ A STRING FROM THE KEYBOARD AND CHECK WHETHER IT IS A PALINDROME OR NOT.

```
.MODEL SMALL

.DATA

BUFF DB 10

L1 DB ?

STR DB 10 DUP(?)

RSTR DB 10 DUP(?)

M1 DB 10,13,"ENTER THE STRING $"

M2 DB 10,13,"REVERSED STRING IS $"

M3 DB 10,13,"STRING IS PALINDROME$"

M4 DB 10,13,"STRING IS NOT PALINDROME$"

.CODE

MOV AX,@DATA

MOV DS,AX

MOV ES,AX

LEA DX,M1

CALL PRINTF

LEA DX,BUFF
```

```
MOV AH,0AH

INT 21H

LEA SI,STR

LEA DI,RSTR

MOV CL,L1

ADD SI,CX

DEC SI

L2:MOV AL,[SI]

MOV [DI],AL

DEC SI

INC DI

DEC CX

JNZ L2

INC DI

MOV AL,'$'

MOV [DI],AL

LEA DX,M2

MOV AH,09H

INT 21H

LEA DX,RSTR

MOV AH,09H

INT 21H
```

```
LEA SI,STR

LEA DI,RSTR

MOV CL,L1

MOV CH,00H

CLD

REPE CMPSB

JZ L3

LEA DX,M4

CALL PRINTF

JMP EXIT

L3:LEA DX,M3

CALL PRINTF

EXIT:MOV AH,4CH

INT 21H

PRINTF PROC NEAR

MOV AH,09H

INT 21H

RET

PRINTF ENDP

END
```

```
Microsoft (R) Overlay Linker  Version 3.64
Copyright (C) Microsoft Corp 1983-1988.  All rights reserved.

Run File [P1.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
LINK : warning L4021: no stack segment

M:\>p1

enter the string sahil
reversed string is lihas
string is not palindrome
M:\>p1

enter the string madam
reversed string is madam
string is palindrome
M:\>p1

enter the string nothing
reversed string is gnihton
string is not palindrome
M:\>_
```

ALP-PGM 5a

WRITE AN ALP TO READ TWO STRINGS, STORE THEM IN LOCATIONS STR1 AND STR2. CHECK WHETHER THEY ARE EQUAL OR NOT AND DISPLAY APPROPRIATE MESSAGES. ALSO DISPLAY THE LENGTH OF THE STORED STRINGS.

```
.MODEL SMALL

.DATA

M1 DB 10,13,"ENTER 1ST STRING: $"

BUFF1 DB 100

LEN1 DB ?

STR1 DB 100 DUP(?)

M2 DB 10,13,"ENTER 2ND STRING: $"

BUFF2 DB 100

LEN2 DB ?

STR2 DB 100 DUP(?)

M1LEN DB 10,13,"LEN OF STR1 IS : $"

M2LEN DB 10,13,"LEN OF STR2 IS : $"

M3 DB 10,13,'STRINGS ARE EQUAL!$'

M4 DB 10,13,'STRINGS ARE NOT EQUAL.$'

.CODE

MOV AX,@DATA

MOV DS,AX

MOV ES,AX
```

```
LEA DX,M1

CALL PRINTF

LEA DX,BUFF1

CALL PUTS

LEA DX,M2

CALL PRINTF

LEA DX,BUFF2

CALL PUTS

MOV AL,LEN1

CMP AL,LEN2

JNZ NOTEQUAL

MOV CX,00H

MOV CL,LEN1

LEA SI,STR1

LEA DI,STR2

CLD

REPE CMPSB

JZ STREQUAL

NOTEQUAL:LEA DX,M4

CALL PRINTF

JMP EXIT

STREQUAL:LEA DX,M3
```

```
CALL PRINTF

EXIT: LEA DX,M1LEN

CALL PRINTF

MOV DL,LEN1

CALL PRINTL

LEA DX,M2LEN

CALL PRINTF

MOV DL,LEN2

CALL PRINTL

MOV AH,4CH

INT 21H

PRINTF PROC NEAR

MOV AH,09H

INT 21H

RET

PRINTF ENDP


PUTS PROC NEAR

MOV AH,0AH

INT 21H

RET

PUTS ENDP
```

```
PRINTL PROC NEAR

MOV AX,00H

MOV AL,DL

MOV BX,10D

MOV CX,00H

L1:MOV DX,00H

DIV BX

PUSH DX

INC CX

CMP AX,00H

JNZ L1

L2:POP DX

ADD DX,30H

MOV AH,02H

INT 21H

DEC CX

JNZ L2

RET

PRINTL ENDP

END
```

```
I:\>link\co str1.obj;
Illegal command: link\co.

I:\>link/co str1.obj;

Microsoft (R) Overlay Linker  Version 3.64
Copyright (C) Microsoft Corp 1983-1988.  All rights reserved.

LINK : warning L4021: no stack segment

I:\>str1

Enter 1st string: sahil agarwal
Enter 2nd string: sahil
Strings are not equal.
Len of str1 is : 13
Len of str2 is : 5
I:\>str1

Enter 1st string: madam
Enter 2nd string: madam
Strings are equal!
Len of str1 is : 5
Len of str2 is : 5
I:\>_
```

ALP-PGM 6a

WRITE AN ALP TO GENERATE AND PRINT FIRST N FIBONACCI NUMBERS. READ 8 BIT NUMBER N FROM KEYBOARD.

```
.MODEL SMALL

.DATA

 VAL1 DB 00H

 VAL2 DB 01H

 LP DB 0H

V1 DB 0H

V2 DB 0H

NL DB 10,13,'$'

M1 DB 10,13,'ENTER A NUMBER $'

M2 DB 10,13,'FIBONACCI$'

.CODE

MOV AX,@DATA

MOV DS,AX

LEA DX,M1

MOV AH,09H

 INT 21H

 MOV AH,01H

INT 21H
```

```
MOV CL,AL

SUB CL,30H

SUB CL,2D

LEA DX,NL

MOV AH,09H

INT 21H

MOV AH,02H

MOV DL,VAL1

ADD DL,30H

INT 21H

MOV AH,09H

LEA DX,NL

INT 21H

MOV AH,02H

MOV DL,VAL2

ADD DL,30H

INT 21H

LEA DX,NL

MOV AH,09H

INT 21H

DISP:  MOV BL,VAL1

ADD BL,VAL2
```

```
MOV AH,00H
 MOV AL,BL
 MOV LP,CL
 MOV CL,10
 DIV CL
 MOV CL,LP
 MOV V1,AL
 MOV V2,AH
MOV DL,V1
ADD DL,30H
MOV AH,02H
INT 21H
MOV DL,V2
ADD DL,30H
MOV AH,02H
INT 21H
MOV DL,VAL2
MOV VAL1,DL
MOV VAL2,BL
LEA DX,NL
MOV AH,09H
INT 21H
```

```
LOOP DISP

MOV AH,4CH

INT 21H

END
```

```
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987.  All rights reserved.


  51768 + 465144 Bytes symbol space free

      0 Warning Errors
      0 Severe  Errors

M:\>link/co 1.obj;

Microsoft (R) Overlay Linker  Version 3.64
Copyright (C) Microsoft Corp 1983-1988.  All rights reserved.

LINK : warning L4021: no stack segment

M:\>1

Enter the Number: 08
0 1 1 2 3 5 8 13
M:\>1

Enter the Number: 12
0 1 1 2 3 5 8 13 21 34 55 89
M:\>_
```