```
;SINGLE DIGIT
;UVA 10055 - Hashmat The Brave Warrior
;Author: Fahim Rahman
.MODEL SMALL
.STACK 100H
.DATA
A DB ?
B DB ?
SUM DB ?
.CODE
MAIN PROC
   ;INITIALIZE DATA SEGMENT
   MOV AX,@DATA
   MOV DX,AX
WHILE:
   ;INPUT A
   MOV AH,1  ;INPUT
   INT 21H
   SUB AL,30H
   MOV A,AL   ;SAVE
   ;INPUT B
   MOV AH,1   ;INPUT
   INT 21H
   SUB AL,30H
   MOV B,AL   ;SAVE
   MOV AL,A
   CMP AL,B
   JG ABIG  ;IF A>B
   ;ELSE
   MOV AL,B
   SUB AL,A  ;B-A
   MOV SUM,AL
   JMP PRINT
ABIG:
   MOV AL,A
   SUB AL,B   ;A-B
   MOV SUM,AL
   JMP PRINT
PRINT:
   ;NEW LINE
   MOV AH,2
   MOV DL,10
   INT 21H
   MOV DL,13
   INT 21H
   MOV AH,2   ;OUTPUT
   ADD SUM,30H
   MOV DL,SUM
   INT 21H
   ;NEW LINE
   MOV AH,2
   MOV DL,10
   INT 21H
   MOV DL,13
   INT 21H
   JMP WHILE
   ;RETURN TO DOS
   MOV AH,4CH
   INT 21H
   MAIN ENDP
```

```
END MAIN
```

```
;UVA 10055 - Hashmat The Brave Warrior
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
A DW ?
B DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DX,AX
WHILE_LOOP:
    ;SCAN 1ST NUMBER
    CALL SCAN_NUM
    MOV A,CX
    PRINTN
    ;SCAN 2ND NUMBER
    CALL SCAN_NUM
    MOV B,CX
    PRINTN
    ;SAVE TO COMPARE
    MOV AX,A
    MOV BX,B
    CMP AX,BX
    ;IF AX>BX
    JG ALARGE
    ;ELSE
    SUB BX,AX
    ;SAVE TO AX FOR PRINTING
    MOV AX,BX
    CALL PRINT_NUM
    ;NEW LINE
    PRINTN
    JMP WHILE_LOOP
ALARGE:
    SUB AX,BX
    CALL PRINT_NUM
    ;NEW LINE
    PRINTN
    JMP WHILE_LOOP
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    long long int a,b;
    while(scanf("%lld %lld",&a,&b)!=EOF)
    {
        if(a>b)
        {
            printf("%lld\n",a-b);
        }
        else
        {
            printf("%lld\n",b-a);
        }
    }
    return 0;
}
-----------------------------------------------
;UVA 11172 - Relational Operator
;Author: Fahim Rahman
.MODEL SMALL
.STACK 100H
.DATA
T DB ?
A DB ?
B DB ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
    MOV AH,1
    INT 21H     ;TEST
    SUB AL,30H
    MOV T,AL
    ;NEW LINE
    MOV AH,2
    MOV DL,10
    INT 21H
    MOV DL,13
    INT 21H
WHILE:
    MOV BL,T
    CMP BL,0
    JE EXIT
    MOV AH,1
    INT 21H   ;A
    SUB AL,30H
    MOV A,AL
    MOV AH,1
    INT 21H   ;B
    SUB AL,30H
    MOV B,AL
    MOV AL,A
    CMP AL,B
    JE EQP   ;A==B
    CMP AL,B
    JL LP    ;A<B
    ;ELSE
    MOV AH,2
    MOV DL,">"
    INT 21H
    DEC T     ;TEST--
    JMP NEWL
LP:
    MOV AH,2
    MOV DL,"<"
    INT 21H   ;TEST--
    DEC T
    JMP NEWL
EQP:
    MOV AH,2
    MOV DL,"="
    INT 21H
    DEC T    ;TEST--
    JMP NEWL
NEWL:
    ;NEW LINE
    MOV AH,2
    MOV DL,10
    INT 21H
    MOV DL,13
    INT 21H
    JMP WHILE
EXIT:
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
END MAIN
```

```
;UVA 11172 - Relational Operator
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
A DW ?
B DW ?
TESTCASE DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DX,AX
    ;INPUT TESTCASE
    CALL SCAN_NUM
    PRINTN
    MOV TESTCASE,CX
    MOV DX,TESTCASE
WHILE_LOOP:
    CMP DX,0
    JZ EXIT
    ;SCAN 1ST NUMBER
    CALL SCAN_NUM
    PRINTN
    ;SAVE IT TO A
    MOV A,CX
    ;SCAN 2ND NUMBER
    CALL SCAN_NUM
    PRINTN
    CMP A,CX
    ;IF A==CX
    JE EQAL
    ;IF A>CX
    JG GRETER
    ;ELSE
    PRINT "<"
    PRINTN
    DEC DX
    JMP WHILE_LOOP
EQAL:
    PRINT "="
    PRINTN
    DEC DX
    JMP WHILE_LOOP
GRETER:
    PRINT ">"
    PRINTN
    DEC DX
    JMP WHILE_LOOP
EXIT:
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    int test,a,b;
    scanf("%d",&test);
    while(test)
    {
        scanf("%d %d",&a,&b);
        if(a==b)
            printf("=\n");
        else if(a>b)
            printf(">\n");
        else
            printf("<\n");
        test--;
    }
    return 0;
}
```

```
-----------------------------------------------
;UVA 136 - Ugly Number
;Author: Fahim Rahman
;DEFINE LIBRARY FUNCTION
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.CODE
MAIN PROC
    ;AS NUMBER IS LARGE TAKING THE
    ;VALUE SEPERATELY
    MOV AX,8599
    MOV BX,633
    MOV DX,92
    ;PRINTING OUTPUT
    PRINT "The 1500'th ugly number is "
    ;PRINT 8599
    CALL PRINT_NUM
    ;TO AVOID OVERRIDE
    PRINT ""
    MOV AX,BX
    ;PRINT 8599633
    CALL PRINT_NUM
    PRINT ""
    MOV AX,DX
    ;PRINT 859963392
    CALL PRINT_NUM
    PRINTN
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
    DEFINE_SCAN_NUM
END MAIN
```

```
#include<stdio.h>
int main()
{
    int num = 859963392;
    printf("The 1500'th ugly number is %d.\n",num);
}
```

```
-----------------------------------------------
;UVA 10696 - f91
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
TESTCASE DW ?
.CODE
MAIN PROC
    ;INITIALIZA DATA
    ;SEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILE_LOOP:
    ;SCAN THE NUMBER
    CALL SCAN_NUM
    PRINTN
    MOV TESTCASE,CX
    ;IF CX==0
    CMP CX,0
    JZ EXIT
    ;ELSE
    CMP CX,100
    ;NESTED IF CX<=100
    JLE P91
    ;NESTED ELSE
    PRINT "f91 "
    MOV AX,TESTCASE
    CALL PRINT_NUM
    PRINT " = "
    SUB CX,10;
    MOV AX,CX
    CALL PRINT_NUM
    PRINTN
    JMP WHILE_LOOP
P91:
    PRINT "f91 "
    MOV AX,TESTCASE
    CALL PRINT_NUM
    PRINT " = 91"
    PRINTN
    JMP WHILE_LOOP
EXIT:
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_STRING
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    int n;
    while(scanf("%d",&n)!=EOF)
    {
        if(n==0)
            break;
        else
        {
            if(n<=100)
                printf("f91(%d) = 91\n",n);
            else
                printf("f91(%d) = %d\n",n,n-10);
        }
    }
    return 0;
}
```

```
--------------------------------------------------
;UVA 10783 - Odd Sum
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
TESTCASE DW ?
A DW ?
B DW ?
SUM DW ?
I DW ?
T DW 1
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
    ;TESTCASE
    CALL SCAN_NUM
    MOV TESTCASE,CX
    PRINTN
```

```
OUTERFOR:
    MOV AX,T    ;T=1 THEN INCREMENT
    CMP AX,TESTCASE
    JG EXIT     ;T>TESTCASE
                ;ELSE
    CALL SCAN_NUM   ;A
    MOV A,CX
    PRINTN
    CALL SCAN_NUM   ;B
    MOV B,CX
    PRINTN
    MOV SUM,0       ;SUM=0
    MOV AX,A        ;LOOP INITIALIZATION
    MOV I,AX        ;I=A
INNERFOR:
    ;CMP
    MOV AX, I
    CMP AX, B
    ;IF A>B
    JG PRINT
    ;ELSE
    AND DX,0    ;CLR DX REG
    MOV AX,I    ;TO DIVIDE I
    MOV CX,2    ;TO DIVIDE BY 2
    DIV CX      ;I/2
    CMP DX,0    ;REMENDER IS IN DX
    JE AFTER    ;IF (I%2=0)
    ;ELSE
    MOV AX,I
    ADD AX,SUM
    MOV SUM, AX ;SUM+=I
AFTER:
    INC I      ;I++
    JMP INNERFOR
PRINT:
    PRINT "Case "
    MOV AX,T
    CALL PRINT_NUM
    PRINT ": "
    MOV AX,SUM
    CALL PRINT_NUM
    PRINTN
    INC T       ;T++
    JMP OUTERFOR
EXIT:
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    int test,a,b,sum,i,t;
    scanf("%d",&test);
    for(t=1;t<=test;t++)
    {
        scanf("%d",&a);
        scanf("%d",&b);
        sum=0;
        for(i=a;i<=b;i++)
        {
            if(i%2!=0)
            {
                sum=sum+i;
            }
        }
        printf("Case %d: %d\n",t,sum);
    }
    return 0;
}
```

```
;UVA 10812 - Beat The Speed
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
TESTCASE DW ?
I DW 1
A DW ?
B DW ?
C DW ?
D DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
    ;SCAN TEST CASE
    CALL SCAN_NUM
    MOV TESTCASE,CX
    PRINTN
LOOOP:
    ;1ST FOR LOOP
    MOV AX,I
    CMP AX,TESTCASE
    JG EXIT
    ;SCAN A
    CALL SCAN_NUM
    MOV A,CX    ;SAVE
    PRINTN
    MOV AX,A
    ;SCAN B
    CALL SCAN_NUM
    MOV B,CX    ;SAVE
    PRINTN
    ;IF B>A
    CMP CX,A
    JG IMPPRINT
    ;ELSE
    AND DX,0 ;CLEAR DX REG
    MOV CL,2 ;TO DIVIDE 2
    SUB AX,B ;AX=A-B
    DIV CL   ;(A-B)/2
    ;IF (A-B)%2==0
    CMP DX,0 ;REMENDER IS IN DX
    ;ELSE
    JE ELSE
IMPPRINT:
    PRINTN "impossible"
    INC I   ;I++
    JMP LOOOP
ELSE:
   MOV CL,2 ;TO DIVIDE 2
   MOV AX,A ;TO SUBTRACT
   SUB AX,B ;AX=A-B
   DIV CL   ;(A-B)/2
   MOV C,AX ;RESULT IN AX
   ADD AX,B
   MOV D,AX
   MOV AX,D ;TO USE DEFAULT FUNCTION
   CALL PRINT_NUM
   PRINT " "
   MOV AX,C ;TO USE DEFAULT FUNCTION
   CALL PRINT_NUM
   PRINTN
   INC I   ;I++
   JMP LOOOP
EXIT:
    ;EXIT TO DOS
    MOV AH,4CH
    INT 21H
   MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
 #include<stdio.h>
int main()
{
    long long int test,a,b,i,c,d;
    scanf("%lld",&test);
    {
        for(i=1;i<=test;i++)
        {
            scanf("%lld %lld",&a,&b);
            if(b>a||(a-b)%2!=0)
                printf("impossible\n");
            else
            {
                c=(a-b)/2;
                d=c+b;
                printf("%lld %lld\n",d,c);
            }
        }
    }
    return 0;
}
```

```
----------------------------------------------------
;UVA 10079 - Pizza Cutting
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
```

```
.DATA
TESTCASE DW ?
VALUE DW ?
ADDD DW ?
.CODE
MAIN PROC
    ;INITIALIZA DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILELOOP:
    CALL SCAN_NUM   ;TEST
    MOV TESTCASE,CX
    PRINTN
    CMP CX,0
    JL EXIT     ;IF(TEST<0) THEN BREAK
    ;ELSE
    MOV CX,TESTCASE
    MOV ADDD,CX
    ADD ADDD,1  ;TEST+1
    MOV AX,TESTCASE
    MOV CX,ADDD
    MUL CX      ;TEST*((TEST+1)
    MOV CX,2
    DIV CX      ;(TEST*((TEST+1))/2
    MOV CX,TESTCASE
    ADD AX,1
    CALL PRINT_NUM
    PRINTN
    JMP WHILELOOP
EXIT:
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    long long int test,value;
    while(scanf("%lld",&test)!=EOF)
    {
        if(test>=0 && test<=210000000)
        {
            value=((test*(test+1))/2)+1;
            printf("%lld\n",value);
        }
        else
            break;
    }
    return 0;
}
```

```
----------------------------------------------------
;UVA 10071 - Back To High School Physics
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
```

```
.STACK 100H
.DATA
S DW ?
V DW ?
T DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILELOOP:
    CALL SCAN_NUM
    MOV V,CX
    PRINTN
    CALL SCAN_NUM
    MOV T,CX
    PRINTN
    MOV AX,V
    MUL CX ;V*T
    MOV CX,2
    MUL CX  ;2*V*T
    CALL PRINT_NUM
    PRINTN
    JMP WHILELOOP
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    int s,v,t;
    while(scanf("%d %d",&v,&t)!=EOF)
    {
        s=2*v*t;
        printf("%d\n",s);
    }
    return 0;
}
```

```
;UVA 913 - Joana and The Odd Numbers
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
N DW ?
SUM DW ?
ODD DW ?
AD DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILELOOP:
    CALL SCAN_NUM
    MOV N,CX
    PRINTN
    MOV AX,N
    ADD AX,2 ;N+2
    MOV AD,AX  ;AD=N+2
    MOV AX,N
    MOV CX,AD
    MUL CX    ;N*(N+2)
    MOV CX,2
    DIV CX    ;(N*(N+2))/2
    MOV ODD,AX
    MOV CX,3
    MUL CX    ;3*ODD
    SUB AX,6  ;(3*ODD)-6
    CALL PRINT_NUM
    PRINTN
    JMP WHILELOOP
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include <stdio.h>
int main()
{
    long int n,sum,odd;
    while(scanf("%ld",&n)==1)
    {
    odd=(n*(n+2))/2;
    sum=(3*odd)-6;
    printf("%ld\n",sum);
    }
}
```

```
;UVA 10346 - Peter's Smoke
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
N DW ?
K DW ?
S DW ?
B DW ?
TB DW ?
R DW ?
SUM DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILE:
    CALL SCAN_NUM
    MOV N,CX
    PRINTN
    CALL SCAN_NUM
    MOV K,CX
    PRINTN
    MOV AX,N
    MOV S,AX    ;S=N
    MOV TB,0    ;TB=0
INNERWHILE:
    MOV AX,S
    CMP AX,K
    JL PRINT
    AND DX,0   ;CLEAR DX REG
    MOV AX,S
    MOV CX,K
    DIV CX     ;S/K
    MOV B,AX   ;B=S/K
    MOV R,DX   ;B=S%K
    MOV AX,B
    ADD AX,R
    MOV SUM,AX ;B+R
    MOV AX,SUM
    MOV S,AX   ;S=B+R
    MOV AX,TB
    ADD AX,B
    MOV TB,AX
    JMP INNERWHILE
PRINT:
   MOV AX,N
   ADD AX,TB   ;TB=TB+B
   CALL PRINT_NUM
   PRINTN
   JMP WHILE
EXIT:
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    int n,k,s,b,tb,r;
    while(scanf("%d%d",&n,&k)==2)
    {
        s=n;
        tb=0;
        while(s>=k)
        {
            b=s/k;
            r=s%k;
            s=b+r;
            tb=tb+b;
        }
        printf("%d\n",n+tb);
    }
    return 0;
}
```

```
;UVA 10970 - Big Chocolate
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
ROW DW ?
COL DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA DEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILELOOP:
    CALL SCAN_NUM
    MOV ROW,CX
    PRINTN
    CALL SCAN_NUM
    MOV COL,CX
    PRINTN    ;NEW LINE
    MOV AX,ROW
    MUL CX  ;ROW*COL
    SUB AX,1    ;T-1
    CALL PRINT_NUM
    PRINTN
    JMP WHILELOOP
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    int t,row,res,col;
    while(scanf("%d %d",&row,&col)!=EOF)
    {
        t=row*col;
        res=t-1;
        printf("%d\n",res);
    }
    return 0;
}
```

```
;UVA 12149 - Feynman
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
A DW ?
I DW ?
S DW ?
SUM DW ?
.CODE
```

```
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILE:
    CALL SCAN_NUM
    MOV A,CX
    PRINTN
    CMP CX,0
    JE EXIT ;IF A=0
    ;ELSE
    MOV SUM,0
    MOV I,1
FOR:
    MOV AX,I  ;INITIALIZATION
    CMP AX,A
    JG PRINT  ;IF I>A
    MOV AX,I
    MOV CX,I
    MUL CX    ;S=I*I
    MOV BX,SUM
    ADD BX,AX    ;SUM=SUM+S
    MOV SUM,BX
    INC I
    JMP FOR
PRINT:
    MOV AX,SUM
    CALL PRINT_NUM
    PRINTN
    JMP WHILE
    EXIT:
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include <stdio.h>
int main()
{
    int a,i;
    while(scanf("%d",&a)==1 && a!=0)
    {
        unsigned long long s,sum=0;
        for(i=1; i<=a; i++)
        {
            s=i*i;
            sum=sum+s;
        }
        printf("%llu\n",sum);
    }
    return 0;
}
```

```
;UVA 13012 - Identifying Tea
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
T DW ?
I DW ?
N DW ?
C DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILELOOP:
    CALL SCAN_NUM
    MOV T,CX
    PRINTN
    MOV I,5    ;I=5
    MOV N,0    ;N=0
INERWHILE:
    MOV AX,I
    CMP AX,0
    JE PRINT   ;IF C==T
    CALL SCAN_NUM
    MOV C,CX
    PRINTN
    CMP CX,T
    JNE AFTER
    INC N
AFTER:
    DEC I    ;I--
    JMP INERWHILE
PRINT:
    MOV AX,N
    CALL PRINT_NUM
    PRINTN
    JMP WHILELOOP
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main ()
{
    int t,i,n,c;
    while (scanf("%d",&t)==1)
    {
        i=5;
        n=0;
        while (i--)
        {
            scanf("%d",&c);
            if (c==t)
            n++;
      }
      printf("%d\n",n);
   }
   return 0;
}
------------------------------------------
;UVA 13059 - Tennis Championship
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
N DW ?
.CODE
MAIN PROC
   ;INITIALIZE DATA SEGMENT
   MOV AX,@DATA
   MOV DS,AX
WHILE:
   CALL SCAN_NUM
   MOV N,CX
   PRINTN
   MOV AX,N
   SUB AX,1    ;N-1
   CALL PRINT_NUM
   PRINTN
   JMP WHILE
   ;RETURN TO DOS
   MOV AH,4CH
   INT 21H
   MAIN ENDP
   DEFINE_SCAN_NUM
   DEFINE_PRINT_NUM
   DEFINE_PRINT_NUM_UNS
END MAIN

#include<stdio.h>
int main()
{
   long long int n;
   while(scanf("%lld",&n)!=EOF)
   {
      printf("%lld\n",n-1);
   }
   return 0;
}
------------------------------------------------
;UVA 13025 - Back To the Past
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.CODE
MAIN PROC
   ;JUST OUTPUT
   PRINTN "May 29, 2013 Wednesday"
   ;RETURN TO DOS
   MOV AH,4CH
   INT 21H
```

```
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    printf("May 29, 2013 Wednesday\n");
    return 0;
}
```

```
;UVA 12917 - Prop Hunt
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
H DW ?
P DW ?
O DW ?
SUM DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILE:
    CALL SCAN_NUM
    MOV P,CX
    PRINTN
    CALL SCAN_NUM
    MOV H,CX
    PRINTN
    CALL SCAN_NUM
    MOV O,CX
    PRINTN
    MOV AX,H
    ADD AX,P    ;H+P
    CMP AX,O
    JLE ELSE    ;IF AX<=0
    PRINTN "Hunters win!"
    JMP WHILE
ELSE:
    PRINTN "Props win!"
    JMP WHILE
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
   int h,p,o;
   while(scanf("%d %d %d",&p,&h,&o)!=EOF)
   {
       if(h+p>o)
           printf("Hunters win!\n");
       else
           printf("Props win!\n");
   }
   return 0;
}
```

```
;UVA 11597 - Spanning Subtree
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
VERTICES DW ?
I DW ?
RES DW ?
.CODE
MAIN PROC
    ;INITIALIZATION DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
    MOV I,1
WHILELOOP:
    CALL SCAN_NUM
    MOV VERTICES,CX   ;VERTICES
    PRINTN
    CMP CX,0        ;IF
    JE EXIT
    MOV AX,CX
    MOV CX,2
    DIV CX
    MOV RES,AX    ;RES IS IN AX
    PRINT "Case "
    MOV AX,I
    CALL PRINT_NUM
    PRINT ": "
    MOV AX,RES
    CALL PRINT_NUM
    PRINTN
    INC I
    JMP WHILELOOP
EXIT:
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include <stdio.h>
int main()
{
    int vertices,i=1;
    while(1)
    {
        scanf("%d",&vertices);
        if(vertices==0)
            break;
        printf("Case %d: %d\n",i,vertices/2);
        i++;
    }
    return 0;
}
```

```
----------------------------------------------------
```

```
;UVA 11044 - Searching for Nessy
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
T DW ?
ROW DW ?
COL DW ?
A DW ?
B DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
    CALL SCAN_NUM
    MOV T,CX
    PRINTN
WHILELOOP:
    MOV AX,T
    CMP AX,0
    JE EXIT
    CALL SCAN_NUM
    MOV ROW,CX
    PRINTN
    CALL SCAN_NUM
    MOV COL,CX
    PRINTN
    AND AX,0
    MOV AX,ROW
    MOV CX,3
    DIV CX    ;ROW/3
    MOV A,AX
    MOV AX,COL
    MOV CX,3
    AND DX,0
    DIV CX    ;COL/3
    MOV CX,A
    MUL CX    ;(ROW/3)*(COL/3)
    CALL PRINT_NUM
    PRINTN
    DEC T
    JMP WHILELOOP
    EXIT:
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
 int main()
 {
    int t,row,col;
   scanf("%d",&t);
   while(t--)
   {
      scanf("%d %d",&row,&col);
      printf("%d\n",(row/3)*(col/3));
    }
    return 0;
 }
```

```
;UVA 12531 - Hours & Minutes
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILE:
    CALL SCAN_NUM
    PRINTN
    AND DX,0
    MOV AX,CX
    MOV CX,6
    DIV CX
    CMP DX,0  ;IF INP%6==0
    JE YP
    ;ELSE
    PRINTN "N"
    JMP WHILE
YP:
    PRINTN "Y"
    JMP WHILE
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    int inp;
    while(scanf("%d",&inp)==1)
        if(inp%6!=0)
```

```
            printf("N\n");
        else
            printf("Y\n");
    return 0;
}
-----------------------------------------------
;UVA 12478 - Hardest Problem Ever
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.CODE
MAIN PROC
    ;JUST PRINT
    PRINTN "KABIR"
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
END MAIN

#include <stdio.h>
int main()
{
    printf("KABIR\n");
    return 0;
}
---------------------------------------------
;UVA 12468 - Zapping
;Author: Fahim Rahman
```

```
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
A DW ?
B DW ?
I DW ?
S DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILE:
    CALL SCAN_NUM ;A
    MOV A,CX
    PRINTN
    CALL SCAN_NUM ;B
    MOV B,CX
    PRINTN
    MOV AX,A
    CMP AX,0
    JL FEXIT      ;IF A=-1
EQ:
    MOV AX,A
    CMP AX,B
    JGE ABIG       ;IF A>B
    ;ELSE
    MOV AX,B
```

```
   SUB AX,A      ;B-A
   JMP COND
ABIG:
   SUB AX,B      ;A-B
   JMP COND
COND:
    CMP AX,50
    JG SUBT      ;A>50
    JMP PRINT
SUBT:
   MOV BX,AX
   MOV AX,100     ;100-S
   SUB AX,BX
   JMP PRINT
PRINT:
   CALL PRINT_NUM
   PRINTN
   JMP WHILE
FEXIT:
   MOV AX,B
   CMP AX,0     ;IF B=-1
   JL EXIT
   JMP EQ
JMP WHILE
   EXIT:
   ;RETURN TO DOS
   MOV AH,4CH
   INT 21H
   MAIN ENDP
   DEFINE_SCAN_NUM
   DEFINE_PRINT_NUM
   DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
#include<stdlib.h>
int main(){
   int a, b, i, s;
   while (scanf("%d%d", &a, &b) == 2 && (a != -1 && b != -1)){
      s = abs(a - b);
      if(s > 50)
         s = 100 - s;
      printf("%d\n", s);
   }
   return 0;
}
```

```
--------------------------------------------------
;UVA 12461 - Airplane
;Author: Fahim Rahman

INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.CODE
MAIN PROC
WHILE:
   CALL SCAN_NUM
   PRINTN
   CMP CX,0
```

```
    JE EXIT   ;IF N==0
    ;ELSE
    PRINTN "1/2"
    ;INFINITE LOOP
    JMP WHILE
EXIT:
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    int n;
    while(scanf("%d",&n)==1)
    {
        if(n==0)
            break;
        else
            printf("1/2\n");
    }
    return 0;
}
```

```
;UVA 12372 - Packing for Holiday
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
A DW ?
B DW ?
C DW ?
T DW ?
I DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILE:
    CALL SCAN_NUM
    MOV T,CX
    PRINTN
    MOV I,1 ;INITIALIZATION
FOR:
    MOV AX,I
    CMP AX,T
    JG WHILE
    CALL SCAN_NUM ;A
    MOV A,CX
    PRINTN
    CALL SCAN_NUM   ;B
    MOV B,CX
    PRINTN
    CALL SCAN_NUM
    MOV C,CX      ;C
    PRINTN
    MOV AX,A
    CMP AX,20
    JG BADPRINT  ;A>20
    MOV AX,B
    CMP AX,20
    JG BADPRINT  ;B>20
    MOV AX,C
    JG BADPRINT  ;C>20
GOODPRINT:
    PRINT "Case "
    MOV AX,I
    CALL PRINT_NUM
    PRINT ": "
    PRINTN "good"
    INC I  ;I++
    JMP FOR
BADPRINT:
    PRINT "Case "
    MOV AX,I
    CALL PRINT_NUM
    PRINT ": "
    PRINTN "bad"
    INC I   ;I++
    JMP FOR
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    int a,b,c,t,i;
    while(scanf("%d",&t)==1)
    {
        for(i=1;i<=t;i++)
        {
            scanf("%d %d %d",&a,&b,&c);
            if(a<=20 && b<=20 && c<=20)
            printf("Case %d: good\n",i);
            else
            printf("Case %d: bad\n",i);
        }
    }
    return 0;
}
```

```
;UVA 11936 - The Lazy Lumberjacks
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
```

```
.DATA
D DW ?
TESTCASE DW ?
A DW ?
B DW ?
C DW ?
.CODE
MAIN PROC
   ;INITIALIZE DATA SEGMENT
   MOV AX,@DATA
   MOV DS,AX
   CALL SCAN_NUM
   MOV TESTCASE,CX   ;TESTCASE
   PRINTN
WHILE:
   MOV AX,TESTCASE
   CMP AX,0
   JE EXIT
   CALL SCAN_NUM    ;A
   MOV A,CX
   PRINTN
   CALL SCAN_NUM    ;B
   MOV B,CX
   PRINTN
   CALL SCAN_NUM    ;C
   MOV C,CX
   PRINTN
   MOV AX,A
   ADD AX,B ;A+B
   MOV D,AX ;D=A+B
   MOV AX,A
   CMP AX,0
   JE WRPRINT       ;IF A==0
   MOV AX,B
   CMP AX,0
   JE WRPRINT       ;IF B==0
   MOV AX,C
   JE WRPRINT       ;IF C==0
   MOV AX,D
   CMP AX,C        ;IF d>C
   JG OKPRINT
   ;ELSE
WRPRINT:
   PRINTN "Wrong!!"
   DEC TESTCASE
   JMP WHILE       ;TEST--
OKPRINT:
   PRINTN "OK"
   DEC TESTCASE    ;TEST--
   JMP WHILE
EXIT:
   ;RETURN TO DOS
   MOV AH,4CH
   INT 21H
   MAIN ENDP
   DEFINE_SCAN_NUM
   DEFINE_PRINT_NUM
   DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    int d,test,a,b,c;
    scanf("%d",&test);
    while(test)
    {
        scanf("%d %d %d",&a,&b,&c);
        d=a+b;
        if(a==0 ||b==0||c==0)
            printf("Wrong!!\n");
        else if(d>c)
            printf("OK\n");
        else
            printf("Wrong!!\n");
        test--;
    }
    return 0;
}
```

```
;UVA 11636 - Hello World
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
LINE DW ?
COPY DW ?
N DW ?
T DW 0
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
WHILE:
    CALL SCAN_NUM
    MOV N,CX
    PRINTN
    CMP CX,0 ;IF N<0
    JL EXIT
    INC T
    MOV LINE,1
    MOV COPY,0
INNERWHILE:
    MOV AX,LINE
    CMP AX,N
    JGE PRINT
    MOV BX,LINE
    ADD BX,LINE
    MOV LINE,BX ;LINE+=LINE
    INC COPY     ;COPY+=1
    JMP INNERWHILE
PRINT:
    PRINT "Case "
    MOV AX,T
    CALL PRINT_NUM
    PRINT ": "
    MOV AX,COPY
    CALL PRINT_NUM
    PRINTN
    JMP WHILE
EXIT:
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include <stdio.h>
int main()
{
    int line,copy,n,t=0;
    while(scanf("%d",&n)==1)
    {
        if(n<0)
        {
            break;
        }
        t+=1;
        line=1;
        copy=0;
        while(line<n)
        {
            line+=line;
            copy+=1;
        }
        printf("Case %d: %d\n",t,copy);
    }
    return 0;
}
```

```
;UVA 12577 - Hajj-e-Akbar
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
ARY DB ?
TC DW ?
.CODE
MAIN PROC
    ;INITIALIZE DATA SEGMENT
    MOV AX,@DATA
    MOV DS,AX
    MOV TC,1
WHILE:
    MOV AX,TC
    MOV TC,AX
    MOV BX,0
INPUT:
    ;INPUT A ELEMENT
    MOV AH,1
    INT 21H
    CMP AL," "   ;CHECK SPACE
    JE PRINTINP
    CMP AL,13    ;CHECK ENTER
    JE PRINTINP
    MOV ARY[BX],AL   ;INSERT IN ARRAY
    INC BX
    JMP INPUT
PRINTINP:
    PRINTN
    MOV SI,0
OUTPUT:
    MOV DL,ARY[SI]
    CMP DL,"*"      ;IF * Then BREAK
    JE EXIT
    CMP DL,"H"
    JE HA
    CMP DL,"U"
    JE HAA
    INC TC
    JMP WHILE
HA:
   PRINT "Case "
   MOV AX,TC
   CALL PRINT_NUM
   PRINT ": "
   PRINTN "Hajj-e-Akbar"
   INC TC
   JMP WHIL
HAA:
   PRINT "Case "
   MOV AX,TC
   CALL PRINT_NUM
   PRINT ": "
   PRINTN "Hajj-e-Asghar"
   INC TC
   JMP WHILE
EXIT:
    ;RETURN TO DOS
    MOV AH,4CH
    INT 21H
    MAIN ENDP
    DEFINE_SCAN_NUM
    DEFINE_PRINT_NUM
    DEFINE_PRINT_NUM_UNS
END MAIN
```

```
#include<stdio.h>
int main()
{
    char ary[6];
    int test=1;
    while(1)
    {
        gets(ary);
        if(ary[0]=='*')
        {
            break;
        }
        else if(ary[0]=='H')
        {
            printf("Case %d: Hajj-e-Akbar\n",test++);
        }
        if(ary[0]=='U')
        {
            printf("Case %d: Hajj-e-Asghar\n",test++);
        }
    }
    return 0;
}
```

```
;UVA 1124 - Celebrity Jeopardy
;Author: Fahim Rahman
INCLUDE EMU8086.INC
.MODEL SMALL
.STACK 100H
.DATA
STR DB 20 DUP(?)
.CODE
MAIN PROC
LOOOP:
  ;BX IS FOR USING ARRAY INDEX
  MOV BX,0
INPUT:
  ;INPUT A ELEMENT
  MOV AH,1
  INT 21H
  ;IF 'ENTER' THEN STOP ENTERING
  CMP AL,13
  JE ENDINPUT
  ;ELSE SAVE IT TO ARRAY
  MOV STR[BX],AL
  INC BX
JMP INPUT
ENDINPUT:
  PRINTN      ;DEFAULT NEW LINE
  ;HAVE TO PRINT FROM 1ST INDEX
  MOV SI,0
OUTPUT:
   CMP SI, BX
   JE ENDLOOP
   ;LOAD A ELEMENT TO PRINT
   MOV DL, STR[SI]
   MOV AH,2
   INT 21H
   ;INCREMENT INDEX TO PRINT
   INC SI
   JMP OUTPUT
ENDLOOP:
   PRINTN
   JMP LOOOP   ;INFINITE LOOP
   ;RETURN TO DOS
   MOV AH, 4CH
   INT 21H
   MAIN ENDP
END MAIN
```

```
#include<stdio.h>
int main()
```

```
{
    char inp[20];
    int i;
    while(gets(inp))
    {
        puts(inp);
    }
    return 0;
}
```
