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

MAIN ENDP

MOV AL,B

```
END MAIN
                                               ;ELSE
                                               SUB BX,AX
;UVA 10055 - Hashmat The Brave Warrior
                                               ;SAVE TO AX FOR PRINTING
;Author: Fahim Rahman
                                               MOV AX,BX
INCLUDE EMU8086.INC
                                               CALL PRINT_NUM
.MODEL SMALL
                                               ;NEW LINE
.STACK 100H
                                               PRINTN
.DATA
                                               JMP WHILE_LOOP
A DW?
                                             ALARGE:
B DW?
                                               SUB AX,BX
.CODE
                                               CALL PRINT_NUM
MAIN PROC
                                               ;NEW LINE
  ;INITIALIZE DATA SEGMENT
                                               PRINTN
  MOV AX,@DATA
                                               JMP WHILE_LOOP
  MOV DX,AX
                                               ;RETURN TO DOS
WHILE_LOOP:
                                               MOV AH,4CH
                                               INT 21H
  ;SCAN 1ST NUMBER
  CALL SCAN_NUM
                                               MAIN ENDP
  MOV A,CX
                                               DEFINE_SCAN_NUM
  PRINTN
                                               DEFINE_PRINT_NUM
  ;SCAN 2ND NUMBER
                                               DEFINE_PRINT_NUM_UNS
                                             END MAIN
  CALL SCAN_NUM
  MOV B,CX
  PRINTN
                                             #include<stdio.h>
  ;SAVE TO COMPARE
                                             int main()
  MOV AX,A
  MOV BX,B
                                               long long int a,b;
  CMP AX,BX
                                               while(scanf("%lld %lld",&a,&b)!=EOF)
  ;IF AX>BX
                                               {
  JG ALARGE
                                                 if(a>b)
```

```
{
                                                 MOV DL,10
       printf("%lld\n",a-b);
                                                 INT 21H
                                                 MOV DL,13
    }
    else
                                                 INT 21H
    {
                                              WHILE:
       printf("%lld\n",b-a);
                                                 MOV BL,T
    }
                                                 CMP BL,0
  }
                                                 JE EXIT
  return 0;
                                                 MOV AH,1
}
                                                 INT 21H ;A
                                                 SUB AL,30H
;UVA 11172 - Relational Operator
                                                 MOV A,AL
;Author: Fahim Rahman
                                                 MOV AH,1
.MODEL SMALL
                                                 INT 21H ;B
.STACK 100H
                                                 SUB AL,30H
.DATA
                                                 MOV B,AL
T DB?
                                                 MOV AL, A
A DB?
                                                 CMP AL,B
B DB?
                                                 JE EQP ;A==B
.CODE
                                                 CMP AL,B
MAIN PROC
                                                 JL LP ;A<B
                                                 ;ELSE
  ;INITIALIZE DATA SEGMENT
                                                 MOV AH,2
  MOV AX,@DATA
                                                 MOV DL,">"
  MOV DS,AX
  MOV AH,1
                                                 INT 21H
                                                 DEC T
  INT 21H
           ;TEST
                                                         ;TEST--
  SUB AL,30H
                                                 JMP NEWL
  MOV T,AL
                                              LP:
  ;NEW LINE
                                                 MOV AH,2
  MOV AH,2
                                                 MOV DL,"<"
```

INT 21H ;TEST	A DW ?
DEC T	B DW?
JMP NEWL	TESTCASE DW ?
EQP:	.CODE
MOV AH,2	MAIN PROC
MOV DL,"="	;INITIALIZE DATA SEGMENT
INT 21H	MOV AX,@DATA
DEC T ;TEST	MOV DX,AX
JMP NEWL	;INPUT TESTCASE
NEWL:	CALL SCAN_NUM
;NEW LINE	PRINTN
MOV AH,2	MOV TESTCASE,CX
MOV DL,10	MOV DX, TESTCASE
INT 21H	WHILE_LOOP:
MOV DL,13	CMP DX,0
INT 21H	JZ EXIT
JMP WHILE	;SCAN 1ST NUMBER
EXIT:	CALL SCAN_NUM
;RETURN TO DOS	PRINTN
MOV AH,4CH	;SAVE IT TO A
INT 21H	MOV A,CX
MAIN ENDP	;SCAN 2ND NUMBER
END MAIN	CALL SCAN_NUM
	PRINTN
;UVA 11172 - Relational Operator	CMP A,CX
;Author: Fahim Rahman	;IF A==CX
INCLUDE EMU8086.INC	JE EQAL
.MODEL SMALL	;IF A>CX
.STACK 100H	JG GRETER
.DATA	;ELSE

```
PRINT "<"
                                                  {
  PRINTN
                                                     scanf("%d %d",&a,&b);
  DEC DX
                                                    if(a==b)
  JMP WHILE_LOOP
                                                       printf("=\n");
EQAL:
                                                    else if(a>b)
  PRINT "="
                                                       printf(">\n");
  PRINTN
                                                     else
  DEC DX
                                                       printf("<\n");</pre>
  JMP WHILE_LOOP
                                                    test--;
GRETER:
                                                  }
  PRINT ">"
                                                  return 0;
  PRINTN
                                               }
  DEC DX
  JMP WHILE_LOOP
                                               ;UVA 136 - Ugly Number
EXIT:
                                                ;Author: Fahim Rahman
  ;RETURN TO DOS
                                               ;DEFINE LIBRARY FUNCTION
                                               INCLUDE EMU8086.INC
  MOV AH,4CH
  INT 21H
                                                .MODEL SMALL
  MAIN ENDP
                                                .STACK 100H
  DEFINE_SCAN_NUM
                                                .CODE
                                               MAIN PROC
  DEFINE_PRINT_NUM
  DEFINE_PRINT_NUM_UNS
                                                  ;AS NUMBER IS LARGE TAKING THE
END MAIN
                                                  ;VALUE SEPERATELY
                                                  MOV AX,8599
#include<stdio.h>
                                                  MOV BX,633
                                                  MOV DX,92
int main()
                                                  ;PRINTING OUTPUT
{
                                                  PRINT "The 1500'th ugly number is "
  int test, a,b;
  scanf("%d",&test);
                                                  ;PRINT 8599
  while(test)
                                                  CALL PRINT_NUM
```

```
;TO AVOID OVERRIDE
                                              .MODEL SMALL
  PRINT ""
                                              .STACK 100H
  MOV AX,BX
                                              .DATA
  ;PRINT 8599633
                                             TESTCASE DW?
                                              .CODE
  CALL PRINT_NUM
  PRINT ""
                                             MAIN PROC
  MOV AX,DX
                                                ;INITIALIZA DATA
  ;PRINT 859963392
                                                ;SEGMENT
  CALL PRINT_NUM
                                                MOV AX,@DATA
  PRINTN
                                                MOV DS,AX
  ;RETURN TO DOS
                                             WHILE_LOOP:
  MOV AH,4CH
                                                ;SCAN THE NUMBER
  INT 21H
                                                CALL SCAN_NUM
  MAIN ENDP
                                                PRINTN
  DEFINE_PRINT_NUM
                                                MOV TESTCASE,CX
  DEFINE_PRINT_NUM_UNS
                                                ;IF CX==0
  DEFINE_SCAN_NUM
                                                CMP CX,0
END MAIN
                                                JZ EXIT
                                                ;ELSE
#include<stdio.h>
                                                CMP CX,100
                                                ;NESTED IF CX<=100
int main()
                                                JLE P91
  int num = 859963392;
                                                ;NESTED ELSE
  printf("The 1500'th ugly number is
                                                PRINT "f91"
%d.\n",num);
                                                MOV AX, TESTCASE
                                                CALL PRINT_NUM
                                                PRINT " = "
;UVA 10696 - f91
                                                SUB CX, 10;
;Author: Fahim Rahman
                                                MOV AX,CX
INCLUDE EMU8086.INC
                                                CALL PRINT_NUM
```

{

}

```
PRINTN
                                                     if(n<=100)
  JMP WHILE_LOOP
                                                        printf("f91(%d) = 91\n",n);
P91:
                                                     else
  PRINT "f91 "
                                                        printf("f91(%d) = %d\n", n, n-10);
  MOV AX, TESTCASE
                                                   }
  CALL PRINT_NUM
                                                }
  PRINT " = 91"
                                                 return 0;
  PRINTN
                                              }
  JMP WHILE_LOOP
EXIT:
                                              ;UVA 10783 - Odd Sum
  ;RETURN TO DOS
                                              ;Author: Fahim Rahman
  MOV AH,4CH
                                              INCLUDE EMU8086.INC
  INT 21H
                                              .MODEL SMALL
                                              .STACK 100H
  MAIN ENDP
  DEFINE_SCAN_NUM
                                              .DATA
  DEFINE_PRINT_STRING
                                              TESTCASE DW?
                                              A DW?
  DEFINE_PRINT_NUM
  DEFINE_PRINT_NUM_UNS
                                              B DW?
END MAIN
                                              SUM DW?
                                              IDW?
#include<stdio.h>
                                              T DW 1
                                              .CODE
int main()
                                              MAIN PROC
{
  int n;
                                                 ;INITIALIZE DATA SEGMENT
  while(scanf("%d",&n)!=EOF)
                                                MOV AX,@DATA
                                                MOV DS,AX
  {
    if(n==0)
                                                 ;TESTCASE
       break;
                                                CALL SCAN_NUM
    else
                                                 MOV TESTCASE, CX
    {
                                                 PRINTN
```

```
OUTERFOR:
                                               MOV SUM, AX ;SUM+=I
  MOV AX,T ;T=1 THEN INCREMENT
                                             AFTER:
                                               INC I
  CMP AX, TESTCASE
                                                       ;|++
  JG EXIT ;T>TESTCASE
                                               JMP INNERFOR
                                             PRINT:
         ;ELSE
  CALL SCAN_NUM ;A
                                               PRINT "Case "
  MOV A,CX
                                               MOV AX,T
  PRINTN
                                               CALL PRINT_NUM
  CALL SCAN_NUM ;B
                                               PRINT ": "
  MOV B,CX
                                               MOV AX,SUM
  PRINTN
                                               CALL PRINT_NUM
  MOV SUM,0
              ;SUM=0
                                               PRINTN
                                               INC T
  MOV AX,A
              ;LOOP INITIALIZATION
                                                        ;T++
  MOV I,AX
                                               JMP OUTERFOR
              ;I=A
INNERFOR:
                                             EXIT:
  ;CMP
                                               ;RETURN TO DOS
  MOV AX, I
                                               MOV AH,4CH
                                               INT 21H
  CMP AX, B
  ;IF A>B
                                               MAIN ENDP
  JG PRINT
                                               DEFINE_SCAN_NUM
  ;ELSE
                                               DEFINE_PRINT_NUM
  AND DX,0 ;CLR DX REG
                                               DEFINE_PRINT_NUM_UNS
  MOV AX,I ;TO DIVIDE I
                                             END MAIN
  MOV CX,2 ;TO DIVIDE BY 2
  DIV CX
         ;1/2
                                             #include<stdio.h>
  CMP DX,0 ; REMENDER IS IN DX
                                             int main()
  JE AFTER ; IF (1\%2=0)
                                             {
  ;ELSE
                                               int test,a,b,sum,i,t;
  MOV AX,I
                                               scanf("%d",&test);
  ADD AX,SUM
                                               for(t=1;t<=test;t++)
```

```
{
                                                 ;INITIALIZE DATA SEGMENT
    scanf("%d",&a);
                                                MOV AX,@DATA
    scanf("%d",&b);
                                                MOV DS, AX
    sum=0;
                                                 ;SCAN TEST CASE
    for(i=a;i<=b;i++)
                                                CALL SCAN_NUM
    {
                                                 MOV TESTCASE, CX
       if(i\%2!=0)
                                                 PRINTN
       {
                                              LOOOP:
                                                 ;1ST FOR LOOP
         sum=sum+i;
       }
                                                MOV AX,I
    }
                                                CMP AX, TESTCASE
    printf("Case %d: %d\n",t,sum);
                                                JG EXIT
  }
                                                 ;SCAN A
  return 0;
                                                CALL SCAN_NUM
}
                                                 MOV A,CX ;SAVE
                                                 PRINTN
;UVA 10812 - Beat The Speed
                                                MOV AX,A
;Author: Fahim Rahman
                                                 ;SCAN B
INCLUDE EMU8086.INC
                                                CALL SCAN_NUM
.MODEL SMALL
                                                MOV B,CX ;SAVE
.STACK 100H
                                                 PRINTN
.DATA
                                                 ;IF B>A
TESTCASE DW?
                                                CMP CX,A
I DW 1
                                                 JG IMPPRINT
A DW?
                                                 ;ELSE
B DW?
                                                 AND DX,0 ;CLEAR DX REG
CDW?
                                                MOV CL,2 ;TO DIVIDE 2
D DW?
                                                SUB AX,B;AX=A-B
                                                 DIV CL ;(A-B)/2
.CODE
MAIN PROC
                                                 ;IF (A-B)\%2==0
```

```
CMP DX,0 ; REMENDER IS IN DX
                                                   DEFINE PRINT NUM UNS
                                                END MAIN
  ;ELSE
  JE ELSE
IMPPRINT:
                                                 #include<stdio.h>
  PRINTN "impossible"
                                                int main()
  INC I ;I++
                                                {
  JMP LOOOP
                                                   long long int test, a, b, i, c, d;
ELSE:
                                                   scanf("%lld",&test);
 MOV CL,2 ;TO DIVIDE 2
 MOV AX,A ;TO SUBTRACT
                                                     for(i=1;i<=test;i++)</pre>
 SUB AX,B ;AX=A-B
                                                     {
 DIV CL ;(A-B)/2
                                                        scanf("%lld %lld",&a,&b);
 MOV C,AX ;RESULT IN AX
                                                        if(b>a||(a-b)\%2!=0)
 ADD AX,B
                                                          printf("impossible\n");
 MOV D,AX
                                                        else
 MOV AX,D ;TO USE DEFAULT FUNCTION
                                                        {
 CALL PRINT_NUM
                                                          c=(a-b)/2;
 PRINT " "
                                                          d=c+b;
 MOV AX,C ;TO USE DEFAULT FUNCTION
                                                          printf("%lld %lld\n",d,c);
 CALL PRINT_NUM
                                                        }
 PRINTN
                                                     }
 INC I ;I++
                                                   }
  JMP LOOOP
                                                   return 0;
EXIT:
  ;EXIT TO DOS
  MOV AH,4CH
                                                ;UVA 10079 - Pizza Cutting
  INT 21H
                                                ;Author: Fahim Rahman
 MAIN ENDP
                                                INCLUDE EMU8086.INC
                                                .MODEL SMALL
  DEFINE_SCAN_NUM
                                                .STACK 100H
  DEFINE_PRINT_NUM
```

```
.DATA
                                                ;RETURN TO DOS
TESTCASE DW?
                                                MOV AH,4CH
VALUE DW?
                                                INT 21H
ADDD DW?
                                                MAIN ENDP
.CODE
                                                DEFINE_SCAN_NUM
MAIN PROC
                                                DEFINE_PRINT_NUM
  ;INITIALIZA DATA SEGMENT
                                                DEFINE_PRINT_NUM_UNS
  MOV AX,@DATA
                                              END MAIN
  MOV DS,AX
WHILELOOP:
                                              #include<stdio.h>
  CALL SCAN_NUM ;TEST
                                              int main()
  MOV TESTCASE,CX
                                              {
  PRINTN
                                                long long int test, value;
                                                while(scanf("%lld",&test)!=EOF)
  CMP CX,0
  JL EXIT
           ;IF(TEST<0) THEN BREAK
                                                {
  ;ELSE
                                                   if(test>=0 && test<=210000000)
  MOV CX, TESTCASE
                                                   {
  MOV ADDD,CX
                                                     value=((test*(test+1))/2)+1;
  ADD ADDD,1 ;TEST+1
                                                     printf("%lld\n",value);
  MOV AX, TESTCASE
                                                   }
  MOV CX, ADDD
                                                   else
  MUL CX
            ;TEST*((TEST+1)
                                                     break;
  MOV CX.2
                                                }
                                                return 0;
  DIV CX
           ;(TEST*((TEST+1))/2
  MOV CX, TESTCASE
  ADD AX,1
                                              ;UVA 10071 - Back To High School Physics
  CALL PRINT_NUM
  PRINTN
                                              ;Author: Fahim Rahman
  JMP WHILELOOP
                                              INCLUDE EMU8086.INC
EXIT:
                                              .MODEL SMALL
```

```
.STACK 100H
                                            #include<stdio.h>
.DATA
                                            int main()
S DW?
                                            {
V DW?
                                               int s,v,t;
T DW?
                                               while(scanf("%d %d",&v,&t)!=EOF)
.CODE
MAIN PROC
                                                 s=2*v*t;
  ;INITIALIZE DATA SEGMENT
                                                 printf("%d\n",s);
  MOV AX,@DATA
                                               }
  MOV DS,AX
                                               return 0;
                                            }
WHILELOOP:
  CALL SCAN_NUM
                                             ;UVA 913 - Joana and The Odd Numbers
  MOV V,CX
  PRINTN
                                             ;Author: Fahim Rahman
  CALL SCAN_NUM
                                             INCLUDE EMU8086.INC
  MOV T,CX
                                             .MODEL SMALL
  PRINTN
                                             .STACK 100H
  MOV AX,V
                                             .DATA
  MUL CX;V*T
                                             N DW?
  MOV CX,2
                                            SUM DW?
  MUL CX ;2*V*T
                                            ODD DW?
  CALL PRINT_NUM
                                            AD DW?
  PRINTN
                                             .CODE
  JMP WHILELOOP
                                            MAIN PROC
  MAIN ENDP
                                               ;INITIALIZE DATA SEGMENT
  DEFINE SCAN NUM
                                               MOV AX,@DATA
                                               MOV DS, AX
  DEFINE_PRINT_NUM
  DEFINE_PRINT_NUM_UNS
                                            WHILELOOP:
END MAIN
                                               CALL SCAN_NUM
                                               MOV N,CX
```

```
PRINTN
                                               {
  MOV AX,N
                                               odd=(n*(n+2))/2;
  ADD AX,2;N+2
                                               sum=(3*odd)-6;
  MOV AD, AX ; AD=N+2
                                               printf("%ld\n",sum);
  MOV AX,N
                                               }
  MOV CX,AD
                                            }
  MUL CX ;N*(N+2)
  MOV CX,2
                                             ;UVA 10346 - Peter's Smoke
  DIV CX ;(N*(N+2))/2
                                             ;Author: Fahim Rahman
  MOV ODD, AX
                                            INCLUDE EMU8086.INC
  MOV CX,3
                                             .MODEL SMALL
  MUL CX ;3*ODD
                                             .STACK 100H
  SUB AX,6 ;(3*ODD)-6
                                             .DATA
  CALL PRINT_NUM
                                            N DW?
  PRINTN
                                            K DW?
  JMP WHILELOOP
                                            S DW?
                                             B DW?
  ;RETURN TO DOS
  MOV AH,4CH
                                            TB DW?
  INT 21H
                                             R DW?
  MAIN ENDP
                                            SUM DW?
                                             .CODE
  DEFINE_SCAN_NUM
                                            MAIN PROC
  DEFINE_PRINT_NUM
  DEFINE_PRINT_NUM_UNS
                                               ;INITIALIZE DATA SEGMENT
END MAIN
                                               MOV AX,@DATA
                                               MOV DS,AX
#include <stdio.h>
                                            WHILE:
int main()
                                               CALL SCAN_NUM
                                               MOV N,CX
{
  long int n,sum,odd;
                                               PRINTN
  while(scanf("%ld",&n)==1)
                                               CALL SCAN_NUM
```

```
MOV K,CX
                                             EXIT:
  PRINTN
                                               ;RETURN TO DOS
  MOV AX,N
                                               MOV AH,4CH
  MOV S,AX ;S=N
                                               INT 21H
  MOV TB,0 ;TB=0
                                               MAIN ENDP
INNERWHILE:
                                               DEFINE_SCAN_NUM
  MOV AX,S
                                               DEFINE_PRINT_NUM
  CMP AX,K
                                               DEFINE_PRINT_NUM_UNS
  JL PRINT
                                             END MAIN
  AND DX,0 ;CLEAR DX REG
  MOV AX,S
                                             #include<stdio.h>
  MOV CX,K
                                             int main()
  DIV CX ;S/K
                                             {
  MOV B,AX ;B=S/K
                                               int n,k,s,b,tb,r;
  MOV R,DX ;B=S%K
                                               while(scanf("%d%d",&n,&k)==2)
  MOV AX,B
                                               {
  ADD AX,R
                                                  s=n;
  MOV SUM, AX; B+R
                                                  tb=0;
  MOV AX,SUM
                                                  while(s>=k)
  MOV S,AX ;S=B+R
                                                  {
  MOV AX,TB
                                                    b=s/k;
  ADD AX,B
                                                    r=s%k;
  MOV TB, AX
                                                    s=b+r;
  JMP INNERWHILE
                                                    tb=tb+b;
PRINT:
                                                  }
 MOV AX,N
                                                  printf("%d\n",n+tb);
 ADD AX,TB ;TB=TB+B
                                               }
 CALL PRINT_NUM
                                               return 0;
 PRINTN
                                             }
 JMP WHILE
```

```
MAIN ENDP
;UVA 10970 - Big Chocolate
                                               DEFINE_SCAN_NUM
;Author: Fahim Rahman
                                               DEFINE_PRINT_NUM
INCLUDE EMU8086.INC
                                               DEFINE_PRINT_NUM_UNS
.MODEL SMALL
                                             END MAIN
.STACK 100H
.DATA
                                            #include<stdio.h>
ROW DW?
                                            int main()
COL DW?
.CODE
                                               int t,row,res,col;
MAIN PROC
                                               while(scanf("%d %d",&row,&col)!=EOF)
  ;INITIALIZE DATA DEGMENT
                                               {
  MOV AX,@DATA
                                                 t=row*col;
  MOV DS,AX
                                                 res=t-1;
WHILELOOP:
                                                 printf("%d\n",res);
  CALL SCAN_NUM
                                               }
  MOV ROW,CX
                                               return 0;
  PRINTN
  CALL SCAN_NUM
  MOV COL,CX
                                             ;UVA 12149 - Feynman
  PRINTN ; NEW LINE
                                             ;Author: Fahim Rahman
  MOV AX, ROW
                                            INCLUDE EMU8086.INC
                                             .MODEL SMALL
  MUL CX ;ROW*COL
  SUB AX,1 ;T-1
                                             .STACK 100H
  CALL PRINT_NUM
                                            .DATA
  PRINTN
                                            A DW?
  JMP WHILELOOP
                                            IDW?
  ;RETURN TO DOS
                                            S DW?
  MOV AH,4CH
                                            SUM DW?
  INT 21H
                                             .CODE
```

```
MAIN PROC
                                                 EXIT:
  ;INITIALIZE DATA SEGMENT
                                                 ;RETURN TO DOS
  MOV AX,@DATA
                                                 MOV AH,4CH
  MOV DS,AX
                                                 INT 21H
WHILE:
                                                 MAIN ENDP
  CALL SCAN_NUM
                                                 DEFINE_SCAN_NUM
  MOV A,CX
                                                 DEFINE_PRINT_NUM
  PRINTN
                                                 DEFINE_PRINT_NUM_UNS
  CMP CX,0
                                              END MAIN
  JE EXIT; IF A=0
  ;ELSE
                                              #include <stdio.h>
  MOV SUM,0
                                              int main()
  MOV I,1
                                              {
FOR:
                                                 int a,i;
  MOV AX,I ;INITIALIZATION
                                                 while(scanf("%d",&a)==1 && a!=0)
  CMP AX,A
                                                 {
  JG PRINT ;IF I>A
                                                   unsigned long long s,sum=0;
  MOV AX,I
                                                   for(i=1; i<=a; i++)
  MOV CX,I
                                                   {
  MUL CX ;S=I*I
                                                     s=i*i;
  MOV BX,SUM
                                                     sum=sum+s;
  ADD BX,AX ;SUM=SUM+S
                                                   }
  MOV SUM, BX
                                                   printf("%llu\n",sum);
  INC I
                                                 }
  JMP FOR
                                                 return 0;
PRINT:
                                              }
  MOV AX,SUM
  CALL PRINT_NUM
  PRINTN
  JMP WHILE
```

```
;UVA 13012 - Identifying Tea
                                                INC N
;Author: Fahim Rahman
                                             AFTER:
INCLUDE EMU8086.INC
                                                DEC I ;I--
.MODEL SMALL
                                                JMP INERWHILE
.STACK 100H
                                             PRINT:
                                               MOV AX,N
.DATA
T DW?
                                               CALL PRINT_NUM
IDW?
                                                PRINTN
N DW?
                                                JMP WHILELOOP
C DW?
                                                ;RETURN TO DOS
.CODE
                                               MOV AH,4CH
MAIN PROC
                                                INT 21H
  ;INITIALIZE DATA SEGMENT
                                               MAIN ENDP
  MOV AX,@DATA
                                                DEFINE_SCAN_NUM
  MOV DS,AX
                                                DEFINE_PRINT_NUM
WHILELOOP:
                                                DEFINE_PRINT_NUM_UNS
  CALL SCAN_NUM
                                             END MAIN
  MOV T,CX
  PRINTN
                                             #include<stdio.h>
  MOV 1,5 ;I=5
                                             int main ()
  MOV N,0 ;N=0
                                             {
INERWHILE:
                                                int t,i,n,c;
  MOV AX,I
                                               while (scanf("%d",&t)==1)
  CMP AX,0
                                               {
  JE PRINT ;IF C==T
                                                  i=5;
  CALL SCAN_NUM
                                                  n=0;
  MOV C,CX
                                                  while (i--)
  PRINTN
                                                  {
  CMP CX,T
                                                    scanf("%d",&c);
  JNE AFTER
                                                    if (c==t)
```

```
INT 21H
         n++;
                                                MAIN ENDP
    }
    printf("%d\n",n);
                                                DEFINE SCAN NUM
  }
                                                DEFINE_PRINT_NUM
  return 0;
                                                DEFINE_PRINT_NUM_UNS
}
                                              END MAIN
;UVA 13059 - Tennis Championship
                                              #include<stdio.h>
                                              int main()
;Author: Fahim Rahman
INCLUDE EMU8086.INC
                                              {
.MODEL SMALL
                                                long long int n;
.STACK 100H
                                                while(scanf("%lld",&n)!=EOF)
.DATA
                                                {
N DW?
                                                   printf("%lld\n",n-1);
.CODE
                                                }
MAIN PROC
                                                return 0;
  ;INITIALIZE DATA SEGMENT
                                              }
  MOV AX,@DATA
  MOV DS,AX
                                              ;UVA 13025 - Back To the Past
WHILE:
                                              ;Author: Fahim Rahman
                                              INCLUDE EMU8086.INC
  CALL SCAN_NUM
  MOV N,CX
                                              .MODEL SMALL
  PRINTN
                                              .STACK 100H
  MOV AX,N
                                              .CODE
  SUB AX,1 ;N-1
                                              MAIN PROC
                                                ;JUST OUTPUT
  CALL PRINT_NUM
  PRINTN
                                                PRINTN "May 29, 2013 Wednesday"
  JMP WHILE
                                                ;RETURN TO DOS
  ;RETURN TO DOS
                                                MOV AH,4CH
  MOV AH,4CH
                                                INT 21H
```

```
CALL SCAN_NUM
  MAIN ENDP
                                               MOV P,CX
                                               PRINTN
  DEFINE SCAN NUM
  DEFINE_PRINT_NUM
                                               CALL SCAN_NUM
  DEFINE_PRINT_NUM_UNS
                                               MOV H,CX
END MAIN
                                               PRINTN
                                               CALL SCAN_NUM
#include<stdio.h>
                                               MOV O,CX
int main()
                                               PRINTN
{
                                               MOV AX,H
  printf("May 29, 2013 Wednesday\n");
                                               ADD AX,P ;H+P
  return 0;
                                               CMP AX,O
}
                                               JLE ELSE ;IF AX<=0
                                               PRINTN "Hunters win!"
;UVA 12917 - Prop Hunt
                                               JMP WHILE
;Author: Fahim Rahman
                                             ELSE:
INCLUDE EMU8086.INC
                                               PRINTN "Props win!"
.MODEL SMALL
                                               JMP WHILE
.STACK 100H
                                               ;RETURN TO DOS
.DATA
                                               MOV AH,4CH
H DW?
                                               INT 21H
P DW?
                                               MAIN ENDP
O DW?
                                               DEFINE_SCAN_NUM
SUM DW?
                                               DEFINE_PRINT_NUM
.CODE
                                               DEFINE_PRINT_NUM_UNS
MAIN PROC
                                             END MAIN
  ;INITIALIZE DATA SEGMENT
  MOV AX,@DATA
                                             #include<stdio.h>
  MOV DS,AX
                                             int main()
WHILE:
                                             {
```

```
PRINTN
  int h,p,o;
  while(scanf("%d %d
                                                 CMP CX,0
                                                              ;IF
%d",&p,&h,&o)!=EOF)
                                                 JE EXIT
  {
                                                 MOV AX,CX
     if(h+p>o)
                                                 MOV CX,2
       printf("Hunters win!\n");
                                                 DIV CX
     else
                                                 MOV RES,AX ; RES IS IN AX
       printf("Props win!\n");
                                                 PRINT "Case "
  }
                                                 MOV AX,I
  return 0;
                                                 CALL PRINT_NUM
}
                                                 PRINT ": "
                                                 MOV AX, RES
;UVA 11597 - Spanning Subtree
                                                 CALL PRINT_NUM
;Author: Fahim Rahman
                                                 PRINTN
INCLUDE EMU8086.INC
                                                 INC I
.MODEL SMALL
                                                 JMP WHILELOOP
.STACK 100H
                                              EXIT:
.DATA
                                                 ;RETURN TO DOS
VERTICES DW?
                                                 MOV AH,4CH
IDW?
                                                 INT 21H
RES DW?
                                                 MAIN ENDP
.CODE
                                                 DEFINE SCAN NUM
MAIN PROC
                                                 DEFINE_PRINT_NUM
  ;INITIALIZATION DATA SEGMENT
                                                 DEFINE_PRINT_NUM_UNS
  MOV AX,@DATA
                                               END MAIN
  MOV DS,AX
  MOV I,1
                                              #include <stdio.h>
WHILELOOP:
                                              int main()
  CALL SCAN_NUM
  MOV VERTICES, CX ; VERTICES
                                                 int vertices, i=1;
```

```
while(1)
                                             WHILELOOP:
                                               MOV AX,T
                                               CMP AX,0
    scanf("%d",&vertices);
    if(vertices==0)
                                               JE EXIT
       break;
                                               CALL SCAN_NUM
    printf("Case %d: %d\n",i,vertices/2);
                                               MOV ROW,CX
    i++;
                                               PRINTN
  }
                                               CALL SCAN_NUM
  return 0;
                                               MOV COL,CX
}
                                               PRINTN
                                               AND AX,0
;UVA 11044 - Searching for Nessy
                                               MOV AX, ROW
;Author: Fahim Rahman
                                               MOV CX,3
                                               DIV CX ;ROW/3
INCLUDE EMU8086.INC
.MODEL SMALL
                                               MOV A,AX
.STACK 100H
                                               MOV AX,COL
                                               MOV CX,3
.DATA
T DW?
                                               AND DX,0
ROW DW?
                                               DIV CX ;COL/3
COL DW?
                                               MOV CX,A
A DW?
                                               MUL CX ; (ROW/3)*(COL/3)
B DW?
                                               CALL PRINT_NUM
.CODE
                                               PRINTN
MAIN PROC
                                               DEC T
  ;INITIALIZE DATA SEGMENT
                                               JMP WHILELOOP
                                               EXIT:
  MOV AX,@DATA
  MOV DS,AX
                                               ;RETURN TO DOS
  CALL SCAN_NUM
                                               MOV AH,4CH
  MOV T,CX
                                               INT 21H
  PRINTN
```

```
MAIN ENDP
                                              WHILE:
  DEFINE_SCAN_NUM
                                                 CALL SCAN_NUM
                                                 PRINTN
  DEFINE_PRINT_NUM
  DEFINE_PRINT_NUM_UNS
                                                 AND DX,0
END MAIN
                                                 MOV AX,CX
                                                 MOV CX,6
#include<stdio.h>
                                                 DIV CX
int main()
                                                 CMP DX,0 ;IF INP%6==0
                                                 JE YP
  int t,row,col;
                                                 ;ELSE
 scanf("%d",&t);
                                                 PRINTN "N"
 while(t--)
                                                 JMP WHILE
                                              YP:
 {
   scanf("%d %d",&row,&col);
                                                 PRINTN "Y"
   printf("%d\n",(row/3)*(col/3));
                                                 JMP WHILE
  }
                                                 ;RETURN TO DOS
  return 0;
                                                 MOV AH,4CH
                                                 INT 21H
}
                                                 MAIN ENDP
;UVA 12531 - Hours & Minutes
                                                 DEFINE_SCAN_NUM
;Author: Fahim Rahman
                                                 DEFINE_PRINT_NUM
INCLUDE EMU8086.INC
                                                 DEFINE_PRINT_NUM_UNS
.MODEL SMALL
                                              END MAIN
.STACK 100H
.DATA
                                              #include<stdio.h>
.CODE
                                              int main()
MAIN PROC
                                              {
  ;INITIALIZE DATA SEGMENT
                                                 int inp;
  MOV AX,@DATA
                                                 while(scanf("%d",&inp)==1)
  MOV DS,AX
                                                   if(inp%6!=0)
```

```
printf("N\n");
                                              INCLUDE EMU8086.INC
    else
                                              .MODEL SMALL
       printf("Y\n");
  return 0;
                                              .STACK 100H
}
                                              .DATA
                                              A DW?
;UVA 12478 - Hardest Problem Ever
                                              B DW?
;Author: Fahim Rahman
                                              IDW?
INCLUDE EMU8086.INC
                                              S DW?
.MODEL SMALL
                                              .CODE
.STACK 100H
                                              MAIN PROC
.CODE
                                                ;INITIALIZE DATA SEGMENT
MAIN PROC
                                                MOV AX,@DATA
  ;JUST PRINT
                                                MOV DS,AX
  PRINTN "KABIR"
                                              WHILE:
  ;RETURN TO DOS
                                                CALL SCAN_NUM; A
                                                MOV A,CX
  MOV AH,4CH
  INT 21H
                                                PRINTN
  MAIN ENDP
                                                CALL SCAN_NUM; B
END MAIN
                                                MOV B,CX
                                                PRINTN
#include <stdio.h>
                                                MOV AX,A
int main()
                                                CMP AX,0
                                                JL FEXIT
{
                                                            ;IF A=-1
  printf("KABIR\n");
                                              EQ:
  return 0;
                                                MOV AX,A
}
                                                CMP AX,B
                                                JGE ABIG
                                                             ;IF A>B
                                                ;ELSE
;UVA 12468 - Zapping
;Author: Fahim Rahman
                                                MOV AX,B
```

```
SUB AX,A
               ;B-A
                                                  DEFINE_PRINT_NUM
  JMP COND
                                                  DEFINE PRINT NUM UNS
ABIG:
                                               END MAIN
  SUB AX,B
               ;A-B
  JMP COND
                                               #include<stdio.h>
COND:
                                               #include<stdlib.h>
   CMP AX,50
                                               int main(){
   JG SUBT
               ;A>50
                                                  int a, b, i, s;
   JMP PRINT
                                                  while (scanf("%d%d", \&a, \&b) == 2 \&\& (a)
                                               != -1 && b != -1)){
SUBT:
                                                    s = abs(a - b);
  MOV BX,AX
                                                    if(s > 50)
  MOV AX,100
               ;100-S
                                                       s = 100 - s;
  SUB AX,BX
                                                    printf("%d\n", s);
  JMP PRINT
                                                  }
PRINT:
                                                  return 0;
  CALL PRINT_NUM
                                               }
  PRINTN
  JMP WHILE
                                               ;UVA 12461 - Airplane
FEXIT:
                                               ;Author: Fahim Rahman
  MOV AX,B
  CMP AX,0
              ;IF B=-1
                                               INCLUDE EMU8086.INC
  JL EXIT
                                               .MODEL SMALL
  JMP EQ
                                               .STACK 100H
JMP WHILE
                                               .CODE
  EXIT:
                                               MAIN PROC
  ;RETURN TO DOS
                                               WHILE:
  MOV AH,4CH
                                                  CALL SCAN_NUM
  INT 21H
                                                  PRINTN
  MAIN ENDP
                                                  CMP CX,0
  DEFINE_SCAN_NUM
```

```
JE EXIT ;IF N==0
                                              ;UVA 12372 - Packing for Holiday
                                              ;Author: Fahim Rahman
  ;ELSE
  PRINTN "1/2"
                                              INCLUDE EMU8086.INC
  ;INFINITE LOOP
                                              .MODEL SMALL
  JMP WHILE
                                              .STACK 100H
EXIT:
                                              .DATA
  ;RETURN TO DOS
                                             A DW?
  MOV AH,4CH
                                              B DW?
  INT 21H
                                             C DW?
  MAIN ENDP
                                             T DW?
  DEFINE_SCAN_NUM
                                             IDW?
  DEFINE_PRINT_NUM
                                              .CODE
                                             MAIN PROC
  DEFINE_PRINT_NUM_UNS
END MAIN
                                                ;INITIALIZE DATA SEGMENT
                                                MOV AX,@DATA
#include<stdio.h>
                                                MOV DS, AX
int main()
                                             WHILE:
                                                CALL SCAN_NUM
{
  int n;
                                                MOV T,CX
  while(scanf("%d",&n)==1)
                                                PRINTN
                                                MOV I,1; INITIALIZATION
  {
                                              FOR:
    if(n==0)
                                                MOV AX,I
       break;
                                                CMP AX,T
    else
       printf("1/2\n");
                                                JG WHILE
  }
                                                CALL SCAN_NUM; A
  return 0;
                                                MOV A,CX
}
                                                PRINTN
                                                CALL SCAN_NUM ;B
                                                MOV B,CX
```

```
PRINTN
                                                 INT 21H
                                                 MAIN ENDP
  CALL SCAN_NUM
  MOV C,CX
             ;C
                                                 DEFINE SCAN NUM
  PRINTN
                                                 DEFINE_PRINT_NUM
  MOV AX,A
                                                 DEFINE_PRINT_NUM_UNS
  CMP AX,20
                                               END MAIN
  JG BADPRINT ;A>20
  MOV AX,B
                                               #include<stdio.h>
                                              int main()
  CMP AX,20
  JG BADPRINT ;B>20
                                              {
  MOV AX,C
                                                 int a,b,c,t,i;
  JG BADPRINT ;C>20
                                                 while(scanf("%d",&t)==1)
GOODPRINT:
                                                 {
  PRINT "Case "
                                                   for(i=1;i<=t;i++)
  MOV AX,I
                                                   {
  CALL PRINT_NUM
                                                      scanf("%d %d %d",&a,&b,&c);
  PRINT ": "
                                                      if(a<=20 && b<=20 && c<=20)
  PRINTN "good"
                                                      printf("Case %d: good\n",i);
  INC I ;I++
                                                      else
  JMP FOR
                                                      printf("Case %d: bad\n",i);
                                                   }
BADPRINT:
  PRINT "Case "
                                                 }
  MOV AX,I
                                                 return 0;
  CALL PRINT_NUM
                                              }
  PRINT ": "
  PRINTN "bad"
                                               ;UVA 11936 - The Lazy Lumberjacks
  INC I ;I++
                                               ;Author: Fahim Rahman
  JMP FOR
                                               INCLUDE EMU8086.INC
  ;RETURN TO DOS
                                               .MODEL SMALL
  MOV AH,4CH
                                               .STACK 100H
```

.DATA

D DW? MOV AX,A

TESTCASE DW ? CMP AX,0

A DW? JE WRPRINT ;IF A==0

B DW ? MOV AX,B

C DW ? CMP AX,0

.CODE JE WRPRINT ;IF B==0

MAIN PROC MOV AX,C

;INITIALIZE DATA SEGMENT JE WRPRINT ;IF C==0

MOV AX,@DATA MOV AX,D

MOV DS,AX CMP AX,C ;IF d>C

CALL SCAN_NUM JG OKPRINT

MOV TESTCASE,CX ;TESTCASE ;ELSE

PRINTN WRPRINT:

WHILE: PRINTN "Wrong!!"

MOV AX, TESTCASE DEC TESTCASE

CMP AX,0 JMP WHILE ;TEST--

JE EXIT OKPRINT:

CALL SCAN_NUM ;A PRINTN "OK"

MOV A,CX DEC TESTCASE ;TEST--

PRINTN JMP WHILE

CALL SCAN_NUM ;B EXIT:

MOV B,CX ;RETURN TO DOS

PRINTN MOV AH,4CH

CALL SCAN_NUM ;C INT 21H

MOV C,CX MAIN ENDP

PRINTN DEFINE_SCAN_NUM

MOV AX,A DEFINE_PRINT_NUM

ADD AX,B;A+B DEFINE_PRINT_NUM_UNS

MOV D,AX ;D=A+B END MAIN

```
T DW 0
#include<stdio.h>
                                               .CODE
                                               MAIN PROC
int main()
{
                                                  ;INITIALIZE DATA SEGMENT
                                                  MOV AX,@DATA
  int d,test,a,b,c;
  scanf("%d",&test);
                                                  MOV DS, AX
  while(test)
                                               WHILE:
  {
                                                  CALL SCAN_NUM
     scanf("%d %d %d",&a,&b,&c);
                                                  MOV N,CX
     d=a+b;
                                                  PRINTN
     if(a==0 | |b==0| |c==0)
                                                  CMP CX,0 ;IF N<0
       printf("Wrong!!\n");
                                                  JL EXIT
     else if(d>c)
                                                  INC T
       printf("OK\n");
                                                  MOV LINE,1
     else
                                                  MOV COPY,0
       printf("Wrong!!\n");
                                               INNERWHILE:
     test--;
                                                  MOV AX,LINE
  }
                                                  CMP AX,N
  return 0;
                                                  JGE PRINT
}
                                                  MOV BX,LINE
                                                  ADD BX,LINE
;UVA 11636 - Hello World
                                                  MOV LINE, BX ; LINE+=LINE
;Author: Fahim Rahman
                                                  INC COPY ;COPY+=1
INCLUDE EMU8086.INC
                                                  JMP INNERWHILE
.MODEL SMALL
                                               PRINT:
                                                  PRINT "Case "
.STACK 100H
.DATA
                                                  MOV AX,T
LINE DW?
                                                  CALL PRINT_NUM
COPY DW?
                                                  PRINT ": "
N DW?
                                                  MOV AX, COPY
```

```
CALL PRINT_NUM
                                                   }
  PRINTN
                                                   printf("Case %d: %d\n",t,copy);
  JMP WHILE
                                                 }
EXIT:
                                                 return 0;
  ;RETURN TO DOS
                                              }
  MOV AH,4CH
  INT 21H
                                              ;UVA 12577 - Hajj-e-Akbar
  MAIN ENDP
                                              ;Author: Fahim Rahman
                                              INCLUDE EMU8086.INC
  DEFINE_SCAN_NUM
  DEFINE_PRINT_NUM
                                               .MODEL SMALL
  DEFINE_PRINT_NUM_UNS
                                              .STACK 100H
END MAIN
                                              .DATA
                                              ARY DB?
#include <stdio.h>
                                              TC DW?
int main()
                                               .CODE
                                              MAIN PROC
{
  int line,copy,n,t=0;
                                                 ;INITIALIZE DATA SEGMENT
  while(scanf("%d",&n)==1)
                                                 MOV AX,@DATA
                                                 MOV DS, AX
  {
     if(n<0)
                                                 MOV TC,1
                                              WHILE:
     {
                                                 MOV AX,TC
       break;
     }
                                                 MOV TC,AX
                                                 MOV BX,0
     t+=1;
     line=1;
                                              INPUT:
                                                 ;INPUT A ELEMENT
     copy=0;
     while(line<n)
                                                 MOV AH,1
                                                 INT 21H
     {
                                                 CMP AL," " ;CHECK SPACE
       line+=line;
       copy+=1;
                                                 JE PRINTINP
```

```
CMP AL,13 ;CHECK ENTER
                                                 PRINT ": "
  JE PRINTINP
                                                 PRINTN "Hajj-e-Asghar"
  MOV ARY[BX], AL ; INSERT IN ARRAY
                                                 INC TC
  INC BX
                                                 JMP WHILE
  JMP INPUT
                                               EXIT:
PRINTINP:
                                                  ;RETURN TO DOS
  PRINTN
                                                  MOV AH,4CH
  MOV SI,0
                                                  INT 21H
OUTPUT:
                                                  MAIN ENDP
  MOV DL, ARY[SI]
                                                  DEFINE_SCAN_NUM
  CMP DL,"*" ;IF * Then BREAK
                                                  DEFINE_PRINT_NUM
  JE EXIT
                                                  DEFINE_PRINT_NUM_UNS
                                               END MAIN
  CMP DL,"H"
  JE HA
  CMP DL, "U"
                                               #include<stdio.h>
  JE HAA
                                               int main()
  INC TC
                                               {
  JMP WHILE
                                                  char ary[6];
HA:
                                                  int test=1;
 PRINT "Case "
                                                  while(1)
 MOV AX,TC
                                                  {
 CALL PRINT_NUM
                                                    gets(ary);
 PRINT ": "
                                                    if(ary[0]=='*')
 PRINTN "Hajj-e-Akbar"
                                                    {
 INC TC
                                                       break;
 JMP WHIL
HAA:
                                                    else if(ary[0]=='H')
 PRINT "Case "
                                                       printf("Case %d: Hajj-e-
 MOV AX,TC
                                               Akbar\n",test++);
 CALL PRINT_NUM
```

```
}
                                                MOV STR[BX],AL
     if(ary[0]=='U')
                                                INC BX
                                              JMP INPUT
       printf("Case %d: Hajj-e-
                                              ENDINPUT:
Asghar\n",test++);
                                                PRINTN
                                                           ;DEFAULT NEW LINE
    }
                                                ;HAVE TO PRINT FROM 1ST INDEX
  }
                                                MOV SI,0
  return 0;
                                              OUTPUT:
}
                                                 CMP SI, BX
                                                 JE ENDLOOP
;UVA 1124 - Celebrity Jeopardy
                                                 ;LOAD A ELEMENT TO PRINT
;Author: Fahim Rahman
                                                 MOV DL, STR[SI]
INCLUDE EMU8086.INC
                                                 MOV AH,2
.MODEL SMALL
                                                 INT 21H
.STACK 100H
                                                 ;INCREMENT INDEX TO PRINT
.DATA
                                                 INC SI
STR DB 20 DUP(?)
                                                 JMP OUTPUT
.CODE
                                              ENDLOOP:
MAIN PROC
                                                 PRINTN
LOOOP:
                                                 JMP LOOOP ; INFINITE LOOP
 ;BX IS FOR USING ARRAY INDEX
                                                 ;RETURN TO DOS
 MOV BX,0
                                                 MOV AH, 4CH
INPUT:
                                                 INT 21H
 ;INPUT A ELEMENT
                                                 MAIN ENDP
 MOV AH,1
                                               END MAIN
 INT 21H
 ;IF 'ENTER' THEN STOP ENTERING
 CMP AL,13
  JE ENDINPUT
                                              #include<stdio.h>
  ;ELSE SAVE IT TO ARRAY
                                              int main()
```

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