2002

.   .   :

. – 2-   -   .:

(05899   24.09.2001), 2002. – 115 .

.

.   . .

.

.

(.,   , 1989)

.

,   ,

,   .

:

.   .,   .   .‾ .   .,

.   .,   .   .‾ .   .,

-

.   . .   .

[1, 3, 5, 6, 7, 10, 11]   .

(., , 1989)

;   ,

XXI   .

(ISO 7185:1990,   .,   , [13])

:   –

,

.

. . .

.   [2, 4, 8, 12].

«   »,

.   .

17   .

(   ).

,   ,

,   ,

.   ,

.

,   .

;

.

.

«   .»,

–   –   ,   ,

. .   «

n»   «   *y*»   ,

.   ,

«   », «   », «   »   . .   ,

,   .

,

.

.

.

–

1. .

1.1*. (4
):
) 5!;) LXIV;) 6,38;) -0,7(4);
) 11/4;) -1/6;) $\sqrt{2}$;) ?;
) $5 \cdot 10^6$;) $-24,8 \cdot 10^{-7}$;) 10^6;) 1/100000

1.2*. :
) -0.00027 +4;) 666 -3;) 1 1

1.3*. :
) 0006;) -0;) 7,0;) 7.;) +0.3;) .3;
) 2/3;) E-1;) 8 0;) 0 -4;) 2* 5;)

1.4. ( ) 100
100.0, 20 2 1? ( )
?

1.5. ,
.

1.6*. , (1/3)*3-1
?

1.7. « » ?
(, $a*t$, at)?

1.8*. :
) $+b + yz$;) $[(ax-b) x+c] x-d$;
) $\dfrac{ab}{c} + \dfrac{c}{ab}$;) $\dfrac{x+y}{a_1} \cdot \dfrac{a_2}{x-y}$;
) $10^4 \alpha - 3\dfrac{1}{5}\beta$;) $(1 + \dfrac{x}{2!} + \dfrac{y}{3!}) / (1 + \dfrac{2}{3+xy})$

1.9*. :
) (p+q)/(r+s)-p*q/(r*s);
) 1E3+beta/(x2-gamma*delta)

1.10*. : 24/(3*4)-24/3/4+24/3*4

1.11. 10 10!
1*2*3*...*10 ?

1.12*.
(+1/2)*(+7/10)-3/4 ?
?

1.13.
(, ln(5), ln5)?

1.14. :
) $(1+x)^2$;)* $\sqrt{1+x^2}$;) $| +b |$;) sin 8;)* $\cos^2 x^3$;
) tg x;)* $\log_2 \dfrac{x}{5}$;) ch x;) arcctg 10^3;)* arcsin x

1.15. ($x>0$):
)* $^{-1}$;)* ^{4};) $^{-2}$;) x^5;
)* ^{100};) 2^{1+x};) $x^{\sqrt{2}}$;)* $\sqrt[3]{1+x}$

1.16*.
(   ),   ?   ?
     ?
1.17*.   ?
1.18.   :

) $\sqrt[8]{x^8 + 8^x}$;   ) $\dfrac{xyz - 3,3\,|x + \sqrt[4]{y}\,|}{10^7 + \sqrt{\lg 4!}}$;   ) $\dfrac{\beta + \sin^2 \pi^4}{\cos 2 + |\operatorname{ctg} \gamma|}$

1.19.   :
) (-b+sqrt(sqr(b)-4*a*c))/(2*a);
) a/b*(c+d)-(a-b)/b/c+1E-8;
) x1+arctan(y2-alpha)/2*abs(x4-ln(5)*y5)/exp(-1)
1.20.   :

)* $y = 1 + x + \dfrac{x^2}{2!} + \dfrac{x^3}{3!} + \dfrac{x^4}{4!}$;

) $f = 6{,}673 \cdot 10^{-8} \cdot \dfrac{m_1 \cdot m_2}{r^2}$;

) $b = e^{|x-y|} + \ln(1+e) \cdot \log_2 \operatorname{tg} 2$

1.21.   , d :
)   , , z;
)* (x_1, y_1) (x_2, y_2);
)   arctg(1+ln x) = $\sqrt{2}$;
)*   , b .
1.22.
:=10;   := +3 ?
1.23.   ,
        t.
1.24.
:=2;   : =5;   := ;   := ?
1.25*.   .
1.26.   , z ,
                          ,   - z, z -
          .
1.27*.   :
   a) trunc(6.9);   ) round(6.9);
) trunc(6.2);   ) round(6.2);
) trunc(-1.8);   ) round(-1.8);
) round(0.5);   ) round(-0.5)
1.28*. d .
1.29*.   :
) 20 div 6;   ) 20 mod 6;
) 20 div 4;   ) 20 mod 4;
) 2 div 5;   ) 2 mod 5;
) 123 div 0;   ) 3.0 mod 3
1.30.   div

        .
1.31*.
   -a mod b + a div b * c
1.32.   :
) 3*7 div 2 mod 7/3 - trunc(sin(1));

)* succ(round(5/2) - pred(3))

1.33*.   (   )   :
) 1+0.0;   ) 20/4;   ) sqr(4);   ) sqr(5.0);
) sqrt(16);   e) sin(0);   ) succ(-2);   ) trunc(-3.14)

1.34.   (,)

,

(,   :=7).   ?

1.35*.   -   ,   -   ,
,   ?
) :=n+1;   ) n:= -1;   ) n:=4.0;   ) y:=trunc(y);
) n:=n div 2;   ) := div 2;   ) n:=n/2;   ) n:=sqr(sqrt(n))

1.36.   ?   .
)* k:=k mod 3+k*cos(0);
) x:=x*2 div 6+x/4

1.37*.   *h*
                  k (,   *k*=130985,   *h* =9).

1.38.   *d*
                  (,   *x*=32.597,   *d*=5).

1.39.   *s*   *k*.

1.40.   *k*-   .   ,   (*h*)
    (*m*)   (, *h*=3   *m*=40,   *k*=13257=
3*3600+40*60+57).

1.41.   *f* -   (   )
                 h   ,   *s*   (0?*h*?11, 0?*m,s*?59).

1.42.   *h* -   *m* -   ,
                     (   ),
            f   (0?*k*<360,   *f* -   ).

1.43.   *k* -   1   365.   1, 2,
..., 6   7   ,   (,   , ...,
)   *k*-   ,   1
   -         .

1.44.   *x*   ,
            .

2.

2.1*.   :
) sqr(x)+sqr(y)<=4   = 0.3,   = -1.6;
) k mod 7 = k div 5-1   k = 15;
) odd(trunc(10*p))   = 0.182

2.2*.   ,
                      :
)   *k*   7;
) $ax^2+bx+c=0$ (?0)   ;
)   (,)   *r*   (1, 0);
)   .

2.3*.   :
   a) not odd(n)   n = 0;
) t and (p mod 3=0)   t = true, p = 101010;
) (x*y<>0) and (y>x)   = 2, y = 1;
) (* <>0) or (y>x)   = 2,   = 1;
) a or (not b)   a = false, b = true
2.4*.   ,
                                       :
) 0< <1;
) =m   (, , *z)*;
) ?m   (, , *z)* (not   );
)   *b*   *true*;
)   *b*   *true.*
2.5.   :
   a)* *a* and (not *a*) ? false;   ) *a* or (not *a*) ? true;   ) not (not *a*) ? *a*;
) true or *a* ? true;   ) false and *a* ? false;   ) *a* or *a* ? *a*
2.6*.   :
   a) true or (1/0>0);   ) (1/0>0) or true
2.7*.   :
) 1 and 0;   ) true+false;   ) true<0;
) not 2=5;   )   >0 or y=4;   ) not not b or or d
2.8.   :
)* a and b or not   and d;
) (>=0) or t and odd(x) or (y*y<>4)
2.9.   *a=true*   *b= false*:
   a) a or b and not ;   ) (a or b) and not a;
) not a and b;   ) not (a and b)
 2.10.   ,
                                       :
)   [0, 1];
)   [0, 1];
)*   [2, 5]   [-1, 1];
)*   [2, 5]   [-1, 1];
)   , ,*z*   ;
)   ,   *z*   ;
)   ,*y*   *z*   ;
)   ,   *z*   ;
)   *true,*   *b*
         false;
)*   (,
                   4,   100
      400;   , 1700, 1800   1900 -   ,   2000 -
).
2.11.   (,)   ,
                                       :
)* (>=) and (y+x>=0) and (<=1);
) (sqr(x)+sqr(y)<1) or (>0) and (abs(x) <=1);
) (trunc(y)=0) and (round(x)=0)
2.12*.   ,   ,
       true,

(. . 1).

2.13*.   :
   a) false<true;   ) ord(false)=1;
) pred(true);   ) ord(succ(false))>0
2.14*.   :
   a) not (pred(c) or (ord(c)=1))   c = true;
) (p<true)=(q=false)   p = q = true;
) a and b > a or b   a = false, b = true
2.15.   ,
                        :
)*   k   ;
)*   b   true;
)   , b   c   true.
2.16.   (,)   ,
                        :
)* (abs(x)<=1)>(abs(y)>=1);
) (sqr(x)+sqr(y)<=4)=(y<=x)
 2.17.   :
) not (a or b) ? (not a) and (not b);
) a and (b or c) ? (a and b) or (a and);
)* a<=b ? not   or b;
) a and b ? (a<true)<b;
) not a ? a<true
2.18.   ,
          (b -   ):
)* <b;   ) =b;   ) (<b)=
2.19.   a=true   =1,   d
                              ?

a)* d:=x<2;   ) d:=not a or odd(x);   ) d:=ord(a)<>x

2.20.   ,

   t true,

   ,   false   :

)   , , z ;

)   , , z ;

)   -   ;

) q (q -   );

) $ax^2+bx+c=0$,   , b 0,

;

)   5 k;

)   (1, 1)   (2, 2)   (1, 1,

2 2 -   1   8);

)   ,   (1, 1)   , «   »   (2,

2).

3.

3.1*.
```
program           (input, output);
   var b, c, d: real;
  begin
   read(b, c);
   d:=sqrt(sqr(b)-4*c);
   writeln('x1=',  (-b+d)/2, '   x2=', (-b-d)/2)
  end.
```
                                             1.0   -2.0 ?

3.2*.   ,   ,

                                                                ,

                          .

3.3.
```
program less (input, output);
   var x: real;  t: boolean;
  begin
   read(x);  t:=x<round(x);
   read(x);  t:=t and (x<trunc(x));
   writeln(t)
  end.
```
                        1.5   -0.8 ?

3.4.   ,   true   false

,   .

3.5*.
```
program                (input, output);
   var a: integer; b: real;
  begin read(a, b); writeln(a*b) end.
```
                                       5.0   6.2 ?

3.6.   :

)   read

,   ,   , $x+1$?
)   ,
                    ,   ,
                    ;   ,
    ?
)   ,
            ,   ?
3.7*.
    program aba (input, output);
        var a, b : integer;
    begin read(a, b, a); writeln(a, b, a) end.
                                1, 2   3?
3.8*.
    program   (input, output);
        var x: integer;
    begin x:=2; writeln('x+1') end.
                    : 3   +1 ?
3.9. $writeln(x_1, \ldots, x_n)$:
                                1 x_n?
                        ,
                    :
    write(1); write(2, 3); writeln(4); write(5, 6); writeln; writeln(7, 8)
3.10*.   ,
                        :
    a
    a^3 a^6
    a^6 a^3 a
3.11*.
    program   (output);
        const   =2.71828;
        var e2: real;
    begin e2:=e*e; writeln(e, e2, e*e2, e2*e2) end.
                                            .   (
)   ,
                ??
3.12*.   ,   *true*   *false*
    ,   ?   ?^{e} . (? 10^{-5}
        .)
3.13.   ,
            17-   ,   .
                                                ,
            25-   ?
3.14.   .
)   ,   -
                ?'
)
                    ?
)   ?
)   ?
)   .

?
)   *maxint, true, false*?

3.15*.   .

a) program A (output);
   const d=5;
   begin d:=sqr(d); writeln('d**2=',d) end.
) program   (input, output);
   const k=true;
   var x: real;
   begin read(x); writeln(ord(x)=k) end.
) program   (input, output);
   var a, b, c: integer;
   begin read(a, b); writeln((a+b+c)/3) end.
) program   (input, output);
   var x: real;
   begin read(x); y:=sqrt(x)+1; writeln(y) end.
) program   (input, output);
   const B=2.5;
   var a, b, c: real;
   begin read(a,c); writeln(a*c>b) end.

3.16.   :
program   (input, output);
   const ?=3.14159;
   var a, b: integer;
begin read(A); d:=odd(?*0) and b>a; writeln(d) end.

3.17.   ,
        31.7?
program time(input,output);var
fi: real;h,m: integer; begin read
(fi); h:=trunc(fi/30); m:=trunc(
(fi-30*h) /0.5); writeln(h,'   ',m
)end.

                                                    .

3.18.   :
) x^x (>0);
)   (?, 2?)
       ln(ctg x-1)= ;
)
              ;
)
        $\begin{cases} a_1x + b_1y = c_1 \\ a_2x + b_2y = c_2 \end{cases}$

                                ,   ;
)   ,
              ;
)
       ;
)
              ;

)   (   )
                                                                                    ;
)   ;
)   ,
                                                    .

3.19.   ,
true   *false*   ,   :
)   , *b*   ,
            $^2+b\ +\ =0$;
)   ,   *b* (*<b*)   ,
      arctg(2^x - |p|) = $\sqrt{2}$
                        [, *b*];
)   ,
                                                                    ;
)   ,
                  ;
)   ,
                                                              0;
)   ,
            ;
)   .   ,
                                    ;
)   (1   8)   .
            ,   .

4.   :   ,   ,   ,

4.1.   :
)* $y = \begin{cases} \cos^2 x & \text{при } 0 < x < 2 \\ 1 - \sin x^2 & \text{иначе} \end{cases}$

)*
        arcsin(1+ln *x*)=*a*,
                        ;
)   ,
                        ,   -   ;
)* *d*=max(*a*,*b*,*c*);
)* $z = \begin{cases} \max(x, y) & \text{при } x < 0 \\ \min(x, y) & \text{при } x \geq 0 \end{cases}$
)   *k*   ,
                              (*x•y*?0);
)   , a_1, a_2, a_3 a_4 ,
                  ;   *n.*

4.2. $y=f(x)$, $f(x)$ (. 2),
.

. 2

4.3.   :
)*   (>0)   -
(,   ,   ,   XXI   2001,   2000   );

) $u = \dfrac{\max^2(x,y,z) - 2^x \cdot \min(x,y,z)}{\sin 2 + \max(x,y,z)/\min(x,y,z)}$;

)* $ax^2+bx+c=0$ $(a?0)$,
                 t *true,* $x1$ 2 -
, t *false,* $x1$ 2
;
)   ,   sin   cos
         [0, ?/2], $y=\sin x$;
)*   , b ,
$a? b? c.$

4.4*. z
z:=0; if x>0 then if y>0 then z:=1 else z:=2
                         :
a) $x= y= 1;$) = 1, = -1;   ) = -1, = 1

4.5*. w 0 w
     0.5, $w,$ w 0,
w 1.

4.6.   :
) if 1< <2 then := +1; y:=0;
         else x:=0; y:=y+1;
) if 1< and x<2
     then begin x:=x+1; y:=0 end;
     else begin x:=0; y:=y+1 end

4.7*. b *true,*
,   *false*   .

4.8*.   ,

     := or b and
(-   )
(,   x:=not a   if a then x:=false else
x:=true).

4.9.   ,
if a then x:=b else :=
           -                .

4.10. :
) $f(x) = 0$,

$$f(x) = \begin{cases} 2ax + |a-1| & \text{при } a > 0 \\ \dfrac{e^x}{\sqrt{1+a^2}} - 1 & \text{иначе} \end{cases}$$

)   ch x, $1+|x|$ $(1+x^2)^x$;
) $_1$, b_1, c_1, a_2, b_2, c_2.
    , $a_1x+b_1y=c_1$ $a_2x+b_2y=c_2$,   ,
                ,   ;
)   , b (?0).
$^2+b + =0.$,   ;
)   , b .
                      ,   0,   3, 2   1
              ,   ,
-            ;
) k 1   180.   , k-

    10111213...9899,

                                          ;

) k. k-
    110100100010000100000...,
                                10.
4.11*.   ?
) if x>0 then   :=2 else; y:=x+1;
) if odd(k) then else k:=0;
) begin x:=2; y:=5; end;
) begin a:=true; ; b:=b or a end;
) begin if x=0 then goto 1; y:=x; 1: end
4.12.   ?
a) begin end;   ) begin x:=0 end;   ) begin ; end
4.13*.   =3, f
                                        ?

begin
    f:=1; i:=2;
1: if i>n then goto 9;
    f:=f*i; i:=i+1; goto 1;
9:
end
4.14.

                :
)*   - b;
) u -   cos(ctg n), $n =$
1, 2, 3, ...;
)* $p=(1-1/2^2) (1-1/3^2) \dots (1-1/n^2)$, $n>2$;
) $y=\cos(1+\cos(2+ \dots +\cos(39+\cos 40) \dots))$.
4.15.   .
)   ,
                      ?
)   ,

?
)   ,
                              ?
4.16.   :
   program errors (input, output);
    const L=18;
    label 18, 8;
    var x, y: integer;
   begin
     y:=0;
   8: read(x); if x<0 then goto L; y:=y+x; goto 18;
   L: writeln(y)
    end.
4.17.   :
   a) begin 77: if | |<1 then goto 5
        begin x:=x/2; goto 77; 5: y:=sin(x) end
      end;
) 34: if 0<x<2
            then if y<1 then goto 34 else goto 15
            else 15: y:=sqr(y)
4.18*.   .   50   .
    .
4.19.   . $n>0$,   .
         ,   .
4.20.   .
     ,   0 (   ).
                                   .

                        5.

5.1*.   f=10!   .
5.2*. s
          :
   a) s:=0; i:=0;
      while i<5 do i:=i+1; s:=s+1/i;
) s:=0; i:=1;
      while i>1 do begin s:=s+1/i; i:=i-1 end;
) s:=0; i:=1;
      repeat s:=s+1/i; i:=i-1 until i<=1;
) s:=1; n:=1;
      for i:=2 to n do s:=s+1/i
5.3*.
         ,   :
)   - b;
)   -   cos(ctg n), $n =$
1, 2, 3 ...;
) $p=(1-1/2^2)(1-1/3^2) \ldots (1-1/n^2),$ n>2;
)   y=cos(1+cos(2+ ... +cos(39+cos 40)...)).

5.4*. k?0 .   - k-   { $_n$},
$x_0=1,\ x_n=nx_{n-1}+1/n$ $n = 1, 2, 3, ...$

5.5*. 10^{-5} -
        tg $x = x$,   .

5.6. k -
                              .

5.7. t *true*   *false*
        , k 3   .

5.8.   .   100   .
                        .

5.9*.   .
    ,   0.   .

5.10.   . n>0 n
    ,   .
                                        .

5.11*.   :
) $y=x^{10}+2x^9+3x^8+ ... +10x+11$;
) $y=11x^{10}+10x^9+9x^8+ ... +2x+1$

5.12.   . $t, a_0, a_1, …, a_n$.
        $a_0 x^n + a_1 x^{n-1} + … + a_{n-1} x + \dfrac{a_n}{t^2+0.5}$

5.13.   :
)* $y = (2n-1)!! = 1 \cdot 3 \cdot 5 \cdot … \cdot (2n-1),\ \ n>0$;
) $y = (2n)!! = 2 \cdot 4 \cdot ... \cdot (2n),\ \ >0$;
) $= n!!,\ \ n>0$

5.14.   :
$$= \sqrt{3+\sqrt{6+...+\sqrt{96+\sqrt{99}}}}$$

5.15.   :
)* $y = \cos x + \cos x^2 + \cos x^3 + … + \cos x^{30}$;
)* $= 1! + 2! + 3! + ... + n!\ \ (n>1)$;
) y - $\sin x,\ \sin \sin x,\ \sin \sin \sin x,\ ...,$ 10^{-4}.

5.16.   (f)
        $f_0=f_1=1;\ \ f_n=f_{n-1}+f_{n-2}$ $n = 2, 3, ...$
)* f- 40-   .
) f-   ,   ($n>1$).
) s -   ,
1000.

5.17.   (*abs*),
        eps>0:
)* $y = e^x = 1+x/1!+x^2/2!+ ... +x^n/n!+ ...$;
) $y = $ sh $x = x+x^3/3!+x^5/5!+ … +x^{2n+1}/(2n+1)!+ ...$;
) $y = \cos x = 1-x^2/2!+x^4/4!- ... +(-1)^n x^{2n}/(2n)! + ...$;
) $y = \ln(1+x) = x-x^2/2+x^3/3- … +(-1)^{n-1} x^n/n+ …\ \ (|x|<1)$;
) $y = $ arctg $x = x-x^3/3+x^5/5- … +(-1)^n x^{2n+1}/(2n+1)+ …\ \ (|x|<1)$
        ,   ,
            eps, -   .

5.18. var k, i: integer; x,   : real;
                                      :

) :=0; for x:=0.1 to 0.9 do y:=y+sin(x);
) k:=81; y:=1; for i:=1 to sqrt(k) do y:=2*y;
) k:=0; for i:=1 to 9 do k:=k+sqr(i); k:=k*i;
) k:=1; for i:=1 to 64 do begin i:=2*i; k:=k+i end

5.19*.   : $y = \sin 1 + \sin 1.1 + \sin 1.2 + ... + \sin 2$

5.20.   . $\sin x$ $\cos x$
[0,1]   0.1   (,
                                        6   ):

x	sin(x)	cos(x)
0.0000	0.0000	1.0000
0.1000	0.0998	0.9950
…		
1.0000	0.8415	0.5403

5.21.   .
?
$?\ln(2+\sin x)\ dx,$
0

                                        $n=100$:
b
$?f(x)\ dx\ ?\ h\bullet[f(x_1)+f(x_2)+... +f(x_n)],$
a
$h=(b-a)/n,$ $=$ $+ih-h/2.$
i
5.22*.   ?
k:=0; for i:=1 to k+3 do k:=k+1
5.23*. s -   ,
     $(\ln x,\ ^x),\ x>1.$
5.24. k -   ,
                    $R\,(R>0)$.
5.25*. $\sin x^n\ (n = 1, 2, ..., 30)$
     , t true,   -
false.
)   .
)   .
5.26*.   true, $(n>1)$ -
          ,   false   .
5.27.   .   100   .   ,
                              .
5.28.   .   70   .   ,
                              .
5.29.   :

$$)^*\ p =\prod_{i,j=1}^{20} \frac{1}{i + j^2};\qquad)\ s = \dfrac{\sum\limits_{k=1}^{10} \dfrac{\sum\limits_{n=1}^{k}\sin kn}{k!}}{}$$

5.30*. k -   ,
          $(1?n?27).$ $(/,\ \text{div}\quad \text{mod})$.
5.31.   ,
                              (   ).
5.32. t true   false

,
.

5.33.   . $F =$
(A and   ) or not (B or   )   :

  A   B   C   F
----
 true   true   true   true
 true   true   false   true
 true   false   true   false
          . . .
 false   false   false   true

5.34.   .   , d ? ($c<d$, ?>0).   ?

$$\int_{c}^{d} \cos e^{x}\, dx ,$$

                                        :

$$\int_{a}^{b} f(x)\, dx \approx I_n = h \cdot [f(a)/2 + f(a+h) + f(a+2h) + \ldots + f(b-h) + f(b)/2],$$

$h=(b-a)/n.$

              : I_n $n = n_0$, $2n_0$, $4n_0$,
$8n_0$ . ., n_0 -   (, $n_0=10$),
                    $|I_{2n} - I_n|/3 < ?$ $I_{2n}.$
5.35.   .   ?>0.   ?
        ?$x^3 - ex^2 + (2e+1)$ + ?$^2 = 0.$
5.36.   . x_1, y_1, x_2, y_2, …, x_n,
$y_n.$ $_i$' $_i$ (   ),
                    .

5.37.   .   80   .
        ,   -   .
5.38.   .   100   .   ,
                    .

5.39.   . $n>1$ x_1, x_2,   …, $x_n.$
              :

$$M = \frac{\sum x_i}{n} , \qquad D = \sqrt{\frac{\sum (x_i - M)^2}{n-1}}$$

5.40.   .
                          x_1, x_2, …, x_n (   ),
                    .
        $nx_1 + (n-1)x_2 + \ldots + 2x_{n-1} + x_n .$
5.41.   . x_1, x_2, …, $x_{55}.$
        $x_1(x_2+x_3)(x_4+x_5+x_6)(x_7+x_8+x_9+x_{10}) \ldots (x_{46}+x_{47}+ \ldots +x_{55})$
5.42.   . 100 .

.

5.43.   .   ,
0.   .

5.44.   .   200   .   ,
«   », . .   .

5.45.   .   ,
.   0.   ,
. (,   1, -34, 8, 14, -5   3   .)

5.46.   . $n>2$ $a_1, b_1, ..., _n, b_n$ ($_i < b_i$).
$_i$ b_i ,
,   ,
.   ,   .

5.47.   . $ax^3+bx^2+cx+d=0$,   , $b,$
d -   , $a?0$ $d?0$. (:
d.)

5.48.   .   ,
, . .   (   )   ,
(,   6   : 6=1+2+3).

5.49.   .   10   .
.

5.50.   . $n>2$.
$[2, n]$.

5.51.   .
.

5.52.   .
,   0.   ,
-          .

5.53.   .   ,   -
0.   ,   -
(.   . 5.16).

5.54.   .   .

5.55.   .   ,
.

5.56.   .   ,
, . .   ,
.

5.57.   . $k.$ k-

) 12345678910111213...,
;
) 149162536...,
;
) 1123581321...,   .

6.

6.1*.   :
)   -   '0' ?   ? '9',   -   ;

)   -   'a' ? c ? 'z',   -   ;
) '9' < ' ';
) '0' = 0;
) ord('0') = 0;
) ' ' = ' ';
) d -   , $c<d$,   ord(c)<ord(d);
) ord(chr(k)) = k k;
) chr(ord(c)) = c .

6.2*.   :
  a) pred('7');   ) succ('0')=pred('2');
) ord('5')-ord('0');   ) 'd'>'a';
) ord('q')<ord('z');   ) chr(25)>=chr(16);
) ord(pred(chr(24)));   ) chr(ord('5')+1)

6.3*. d. t
    true,   ,   *false*
    :
) d -   '*';
) d -   ' '   'q';
) d -   .

6.4*. d
                            :
)   :='+'; d:=c;   )   :='+'; d:='c'

6.5*. s -   ,
SUM.

6.6*.   ,   65,
71   69.

6.7*.   *next*   ,   ,
                                        dig,   ,   '9'
    '0'.

6.8*. b *true*,
' '   'z'   ,   ,   *false*
.

6.9*.   ' '   'Z',
.

6.10.   :
)* 100...00   ) 999...99   ) 0123456789
    020...00   088...88   1234567890
    …   …   …
    000...09   000...01   9012345678

                                    «   »
    input   (,   ),
    (   ).

6.11*.   .   *true*,
    , b,   *false*   .
6.12.   .   *key*,
        yes,   - *no*.
6.13.   .   ,
                    (. .

).   -

6.14*.   ,
           (,   ).

6.15.   ,   ,
       .   ,
                .   (-
              ):
    repeat read(c) until c='a'; writeln(succ(c)) ?

6.16.   .   :
)*   «+»   «-»;
)*   «+»,   ;
)   *b*,
     ;
)   *ph*   *f.*

6.17.   .   ,   ,
. .   .

6.18.   .   ,
              60   ,   ,
           .

6.19.   .
              ;   ,
        -           .   ,   :
)*   ;
)   *w*;
)   ;
)   *d*;
)   .

6.20*.   *2,   1   0*   .
                    k   ,   (,
2='8', *1*='0'   *0*='5',   *k*=805).

6.21*.   *2, 1   0*   ,
                    k.

6.22*.   , . .   *r   d( ),*   -
        ,   ,
                  «+»   «-»   ,   ,
                    ,   *k.*

6.23.   , . .   *write(c),*   -
              ,   *k* («+»
).

6.24.   .   .
                    0   .

6.25.   .
        ,   ;   -
     ,   -   .
     .

6.26.   .   ,
                          (-   ).
              .

6.27.   .   ,   :

$d_1 \pm d_2 \pm ... \pm d_n$ (d_i -   , $n>1$),   .

6.28.   ,   (
          -                    ),   ,
                                   .

6.29.   ,
                    ;
    $\pm 0.d_1 d_2 ... d_9 \text{E} \pm p_1 p_2$,
    d_j, p_j -   , $d_1 ?0$, $?0$.
6.30*. k 1   80.
(-   1   80):
)   * k-   ;
)   * k-   I   -   ($k=n$
).
    6.31.   . $= ^2$-1   [-1, 2]
0.1.   ,   OY -   .
                                   «   »   (, |)   -
              ,   ;   OY   .
                                                  :
                    *
                * |
              * |
              *|
                    *
                | *
                | *
    6.32.   .   1, 2, ... k, k-
2   70, $?(n)$,   1
    -1,   .
    6.33.   .   (
          -          ).   ,   :
)   ,   9;
)   ;
)   ,   6;
)   ,   4;
)   ,   5;
)   ,
                    ,   ;
) k (1? k ?9),
          - k;
)   0123456789 (: 0, 01,
012);
)   0123456789 (: 9, 89,
789);
)   -   0123456789 (: 2, 678);
)   ,
                                   (: 2468, 741, 3);
)   (   )   ,
                                   ;
)   ,   ,   .

7.   .

7.1*.
   type   = (,   ,   ,   );
   var , :   ;   t : (,   );
                                .

)   ,   *t*?
            :
    1)   :=   ;   2)   := ;   3) t:=   ;   4) y:=t;   5) t:=   ?
)   :
      1)   <   ;   2)   <=   ;   3)   <   ;   4)   <>   ;
      5) succ(   );   6) r d(   );   7) succ(   );   8) r d(   )
)   :
      1) rd(   );   2) rd(   )+ rd(   )
)
      for x:=   to   do ?
)   -   :
      1) read(x);   2) write(   );   3) writeln('   ', t) ?
7.2*.   :
   type   = ('a', 'b', 'c', 'd');
                  = (a, e, i,   , u);
                   = (b..d, f, g);
              = (6, 7, 8, 9, 10,   ,   ,   ,   );
                = (1.00, 1.41, 1.73, 2.00);
                = (,   ,   ,   );
            = (,   ,   );
                  = (,   ,   );
                  = (div, mod);
            = boolean;
7.3*.
   type digit = '0'..'9';
   var d: digit;   k: 0..9;   n: integer;

                               .
)   *digit*?
   ?   :
      1) d:='7';   2) d:='a';   3) d:=7 ?
)   *k*?   ?
                              :
      1) k:=5;   2) k:=10;   3) k:= -0;   4) k:='5';
      5) k:=ord(d);   6) k:=ord(d)-ord('0') ?
)   ,
          ,   ?
      if k+n>7*k then k:=abs(n) mod 10 else d:=chr(k+ord('0')) ?
)   ,
                   ?
7.4*.   :

```
       const n=180;  pi=3.14159;
       type        = '0'..'9';
                   =   ..z;
               = -n..n;
                  = 0..n-1;
                = -pi..pi;
              = '+'..'+';
                = (   ,    ,    ,    ,    ,    ,    );
                =     ..   ;
                  =     ..   ;
   7.5*. type       = (    ,    ,    ,    ,    ,    ,    ,    ,    ,    ,    ,    );
                = 1..31;
         var d1, d2:      ;  m1, m2:        ;  t: boolean;
              t                    true,          d1, m1            (
    )      d2, m2,           false      .
  7.6*.  var m, m1:        ;  {                «       »    .  7.5}
          k: 1..maxint;  n: 1..12;
                       1 :
     )                    ,                          (                ,
                );
     )          k-                       ;
     )          n-                   .
   7.7. type       = (          ,          ,        ,        ,
                            ,          ,      );
                 = (      ,      ,        ,      ,      ,        ,        );
       var st:        ;      :        ;
                        st  (                     )                        cap
                    .
   7.8. var P: (     ,        ,          2,      ,          ,    1,          );
          A: (ada, basic, modula2, lisp, pascal, pl1, fortran);
     P  -
                              .
   7.9. type         = (      ,      ,      ,      ,          ,        );
       var  d: '0'..'5';  n:            ;
           -       d                          n                      .
   7.10. type       = (   ,    ,    ,    ,      ,    ,    );
                     = (          ,        ,        ,        ,        ,          );
         var n1, n2:       ;  i:            ;
              i -            ,                        n1    n2  (n1<n2):           -
                              (          ,          ),         -
  (          ,          )  . .
     7.11*.                ,                                                d

      p:=true; d:=1;
      case  k mod 10  of
      3, 2, 7, 5:  d:=k;
            1:  ;
         4, 8:  begin p:=false; d:=2 end;
         9, 6:  begin p:=false; d:=3 end
      end {of case}
```

k :

) 6;   ) 235;   ) 71;   ) 100

7.12. var u, w: 'a'..'z';

                                                         :

   case u of
    'a'..'w': w:=succ(u);
    'u', 'y': u:='g'; w:=u
   end

7.13*. type   = (,   ,   ,   );
                   = (,   ,   ,   ,   ,   ,   ,   ,   ,   ,   ,   ,   );
      var m: ; s: ;
          s -   ,   .

7.14. type   = (,   ,   ,   ,   ,   );
                   = (,   ,   );
      var s: ; :   ;
  s -   -   .

7.15. type   = (,   ,   ,   ,   );
                = r   l;
      var :   ; :   ;
                        ,   ,
                              .

7.16*. type   = (,   ,   );
      var :   ;
                                  .

7.17. var k: 1..9;

                                k .

7.18*. type letter = (a, b, c, d);
      var : letter;

                                             letter (. . , b,   d)
                        .

7.19. k 1   99   « k »,
     , k «   »   «   »
   «   ».

7.20. k « k »,
                         «   » k

7.21. type   = (,   ,   ,   ,   ,   );
                   = (,   ,   ,   );
      var w: ; :   ;
            w   (,   *w=cmen*
=   ).

7.22. type   = (,   ,   ,);   {   ,   ,   ,   }
                   = (,   ,   ,   );
      var 1, 2:   ;   :   ;
                            K1,
   .   *2* -   .

7.23*. var d: 28..31; m: ;   {   . 7.5}
            d   (
).

7.24. var y: 1901..2099; m: ;   {   . 7.5} d: 1..31;
          t: boolean;

t   *true,*   , *m, d*
,   *false* -   (31   . .).

7.25. var d, d1: 1..31;   m, m1:   ; {   . 7.5}
       y: 1901..2099;   1: 1901..2100;
    d, ,   *d1, m1, y1* -   .

7.26. var k: 1..366; d: 1..31; m: ; {   . 7.5}
)*   *k* -   ,
      d,   ;
)   *d,*   -   *k*-   .

7.27. type   = 1..31;
            = (,   ,   ,   ,   ,   ,   ,   ,   ,   ,   ,   );
              = (,   ,   ,   ,   ,   ,   );
    var d: ; m: ; wd1, wd: ;
       k: 0..12;
    ,   1   *wd1,*
       :
) *wd* -   ,   *d,*   ;
) *k* -   ,   13-   .

7.28.   ,
                  :

a) program   (input, output);
     type   = (,   ,   ,   ,   ,   ,   ,   ,   ,   ,   ,   );
              =   ..   ;
              =   ..   ;
    var m: ; k: 1..12;
   begin
    read(m);
    if m>   then m:=   ;
    for k:=ord(   ) to ord(m) do m:=succ(m);
    writeln(m)
   end.
) program   (input, output);
     type   = '0' . '9';
          = ('+', '-', '*', '/');
    var d: ; t: boolean;
   begin
    read(d);
    case d of
      2, 3, 5, 7: t:=true; d:=succ(d);
   0, 1, 4, 6, 8, 9: t:=false;
    writeln(t, d)
   end.

7.29.   60-   ,
   12-   .   :   ,
     ,   ,   .
          :   ,   , (1984   -   ,
     ,   ,   . (1984   -   -
).
         ,
                       .

7.30.   .   f(*n*)
                            : f(1)=4 (   ), f(3)=3 (   ), f(42)=8 (   )   . .
                                        ,   100,   f(*n*)=*n*.

                        8.   :

8.1*.   :
  type   = (,   ,   );
                    = array [1..30] of real;
  var   :   ;
      b: packed array [-2..2] of (x,y,z);
      c: array ['0'..'9'] of   ;
      d: array [   ] of 0..23;
                            a, b,   d   :
)   ;
)   ;
)   .
8.2*.   *R,*   ,
                                ,   -   .
8.3.   .
)   ?   ?
)
(   )?
)   1, 1.41, 1.73   2 ?
)   ,   ?
)   *integer*   *real* ?
)
                    ?   (   )   ,
                        ?
8.4*.   :
  const n=50;
  type   = packed array [0..n-1] of   ;
                = ' '..'z';
                = array [real] of integer;
                = arr   [true..false] of (1, 2, 3, 4);
  var k: 1..maxint;
      x: array [1..k] of char;
      y: packed array [-n..n] of 0..0;
      z: array [(a, b, c)] of boolean;
8.5*. const n=41;
      var x: array [1..n] of real; y: real;

                                                      :

) $y = \sqrt[n]{|x_1 \cdot x_2 \cdot \ldots \cdot x_n|}$;

) $=\max\limits_{i}$ $_i$;

) $y = x_1 - x_2 + x_3 - \ldots - x_{n-1} + x_n$;
) $= x_1 x_{2n} + x_2 x_{n-1} + \ldots + x_n x_{12}$;
) $y = x_1^2 + x_3^2 + x_5^2 + \ldots + x_n^2$;

$$) = {}_n(x_n + x_{n-1})(x_n + x_{n-1} + x_{n-2}) \ldots (x_n + \ldots + x_1)$$

8.6. const m=50;

   var C: packed array [0..m] of char;

      B: packed array [0..m] of boolean;

                              :   *B[i]*   *true,*   *C[i]* -   ,

      false   .

8.7. type   = (,   ,   ,   ,   ,   ,   ,

               ,   ,   );

   var   : array [   ] of (,   );

         : rr   [   ] of 140..200;

      :   ; Cp: real;

                              :

)*   -   ;

)   -   .

8.8*. type   = packed array [1..72] of char;

            = packed array [char] of char;

    var t: ; k: ;

         t,

k,   .

8.9. type   = (,   ,   ,   ,   ,   ,   );

   var   : array [1..365] of   ;

                        [i]   ,

      i-   ,   ,   1   -

   ([1]=   ,   [2]=   . .).

8.10. type   = (,   ,   );

            = (,   ,   ,   ,   ,   );

   var x: array [1..20] of   ;

                        ,

      (,   ).

8.11.   ,   (   )

               ,   :

   program   (input, output);

   var x, y: array [1..20] of real;

      z, u: array [1..50] of real;

      i: integer;

   begin

     read(x,y);

     if x<>y then begin z:=x; x:=y; y:=z end else x:=x+y;

     for i:=1 to 20 do u[i]:=x[i]+y[i];

     z:=u;

     writeln(x)

   end.

8.12*. const n=30;

      type   = array [1..n] of integer;

      var , b, :   ;

            b   ,   ,

                  a.

8.13*.   .   100   .

6   .

8.14*. var x: array [1..40] of char; y: array [0..39] of char;

. (
:= ?)

8.15*. ,
?
) 50 . .
) 50 . ,
.
) 100 . ,
.
) .

$x_1, x_2, x_3, \ldots,$ $x_1 = a$, $x_n = \mathrm{tg}(x_{n-1})$.
8.16*. .
$(x_1 y_1 + x_3 y_3 + \ldots + x_{29} y_{29}) / (x_2 y_2 + x_4 y_4 + \ldots + x_{30} y_{30})$,
:
а) $x_1, x_2, \ldots, x_{30}, y_1, y_2, \ldots, y_{30}$;
) $x_1, y_1, x_2, y_2, \ldots, x_{30}, y_{30}$
8.17. . $a_0, a_1, \ldots, a_{20}, t$

$a_{20} x^{20} + a_{19} x^{19} + \ldots + a_1 x + a_0$
t.

,

45- ?
8.18*. . 80 . ,
, ,
.

8.19. . , 1 70 ,
. .
8.20. . , .
, (
,).
8.21. const n=1000;
    var s: packed array [1..n] of char;
        s, :
) * (1, 2, 4, 8, 16, ...);
) (1, 4, 9, 16, 25, ...);
) (1, 2, 3, 5, 8, 13, ...).
8.22. var x: array [1..100] of real; a: array [1..30] of 1..100;
        s: real;
        s - ,
        $(a_i ? a_i$ $i ? j)$.
8.23. var x: array [1..9999] of real; s: real;
        (1- -):
$s = (x_1 + x_2 + x_3)(x_4 + x_5 \ldots + x_8)(x_9 + \ldots + x_{15}) \ldots (x_{9801} + \ldots + x_{9999})$
8.24. const n=20;
    var s: packed array [1..n] of char;
        s_i s :
$s_1\ s_2\ s_3\ \ldots\ s_{n-1}\ s_n$
$s_2\ s_3\ s_4\ \ldots\ s_n\ s_1$
. . .
$s_n\ s_1\ s_2\ \ldots\ s_{n-2}\ s_{n-1}$

8.25.   . $a_0, a_1, \ldots, a_{99},$ a_0 -

, $a_n = a_{[n/2]} + a_{n-1}$ $n = 1, 2, \ldots, 99.$

8.26. const n=100;

var x: array [1..n] of real;

($'_k$ -   k-

):

) $x'_k = \max x_i$ $1 ? i ? k;$

)*   ;

) $x'_1 = x_1, \quad x'_n = x_n, \quad x'_k = (x_{k-1} + x_k + x_{k+1})/3$ $k = 2, 3, \ldots, n\text{-}1;$

)*   : $x'_n = x_1,$

$x'_k = x_{k+1}$ $k = 1, 2, \ldots, n\text{-}1;$

)   .

8.27. var x, y: array [1..70] of real;

k: 1..69;

(

):

)   ,

-                ,

,   ;

) k .

8.28*. const k=50; m=20; n=70; {n=k+m}

var x: array [1..k] of real;

y: array [1..m] of real;

z: array [1..n] of real;

.

z ,

.

8.29. var k: 0..99999;

d: packed array [1..5] of '0'..'9';

) d $k.$

) k -   , $d.$

8.30. type   = packed array [1..9] of '0'..'9';

= packed array [1..2] of '0'..'9';

var m: ;   :   ; x: real;

x

$0.m_1 m_2 \ldots m_9 \cdot 10^{p_1 p_2}$

8.31. type   = (,   ,   ,   ,   ,   ,   ,   ,   ,   ,   ,   );

var   : rr   [   ] of 28..31;

,

(   )   .

8.32. var t: array [1..365] of real;   m: ; {   . 8.31}

$t,$

, m -

.

8.33. const n=40;

var x: array [1..n] of integer;

y, k: integer; t: boolean;

:

)* t $true,$

, $false$;

) t true,
                                                                                                    ,
      false   ;
) k ,
      $n+1$, x;
) $y=x_1+x_1x_2+x_1x_2x_3+ \ldots +x_1x_2\ldots x_m$, m -
                                                  , n,
                              .

8.34.   .   80   .   ,   , . .
                                        .

8.35*. type   = (,   ,   ,
                                          ,   );
                = (,   ,   ,   ,   ,   );
      var MM511: array [   ] of   ;
                        : boolean;
                                                    true,   511
            ,   false   .

8.36. var x: array [1..50] of 1..maxint; t: boolean;
            t true   false   ,
                                    :
)   ;
)   .

8.37. type   = packed array [1..10] of char;
      var , :   ;   eq: boolean;
        ,   ,
            eq   true,
                    ,   false   .

8.38. const n=100;
      var x: array [1..n] of real;
                                          (. .   ,
            k $_k$? x_{k+1}),
(   ):
)*   :
            ;   ,
            (   ),   . .;
)   (   ):
                    $_k$ x_{k+1} ($k = 1, 2, 3, \ldots, n-1$)   , $x_k>x_{k+1}$,
            ;
            ;   ,   ,
. .;
)   : k
            ;   ($k+1$)- k
      , $k+1$;
            k 1 $n-1$.

8.39*. const n=500;
      var x: array [1..n] of integer; p: integer; k: 0..n;
                              .
            k ,   ,   0,
      .
                                    (   )   :

(   );

,   ;   ,

,   -   ;

.

8.40. const n=80;

    type   = 0..9;

                = packed array [1..n] of   ;

    var   , b, :   ; t: boolean;

                b

,   -

.   ,

,   *t*   *true*,   *t*

false.

8.41. type   = 0..99;

                = packed array [   ] of boolean;

    var   , , z: ; t: boolean;

                x,   *z*

                (*x*[*k*]=*true*,   *k*   ,   -

$x[k]=false$,   . .),   -

:

)   *t*   *true*,

,   *false*   ;

) $z = x(\ y-$;

) $z = x(\ y-$;

) $z = x \setminus y -$ (*z*   ,

).

8.42. const n=20; n1=21; {n1=n+1}

    var P, Q: array [0..n] of real;

        R: array [0..n1] of real;

        a: real;

  -

    $P(x) = p_0 x^n + p_1 x^{n-1} + \ldots + p_{n-1} x + p_n$

    :

)* *R* -   (-) ( );

) *Q* -   (+).

8.43.   .   100   .

,

(   ).

8.44.   .   *n*   : $x_1, y_1, \ldots, x_n, y_n$

(*n*=20).   ,

(,   ).

8.45.   .   30   .

,

(,

).

8.46.   .   20   .

(. .   ,

: $>_j$ $i<j$).

8.47.   .   ,

.   ,

.

8.48.   .   100   ,
                                    .

8.49.   .   ,
        ,   100   (
).

8.50.   .

                            ;   -   ,   -
        .   ,
                        .

8.51.   .   «   »
                    , . .   ,
                        . (,   «   »
$s_0^2 + s_1^2 + \ldots + s_{27}^2$, s_n -   0   999,
n.)

8.52.   . k 2   20. k-
                        . (: $T_n(x)$

$T_0(x) = 1;\ \ T_1(x) = x;\ \ T_n(x) = 2x T_{n-1}(x) - T_{n-2}(x),\ \ n = 2, 3, \ldots)$

8.53.   . a_0, a_1,   …, a_{15}.
                        $(x - a_0)(x - a_1)\ldots(x - a_{15})$.

8.54.   .   15-
            8-
        .

8.55.   .   ( ) 10-
            $Q(x)$ 6- $P(Q(x))$.

                        9.   :

9.1*.   array [1..15, 0..3] of char   array [1..15] of array [0..3] of
char ?

9.2. type   = rr   [1..20] of char;
                1 = rr   [1..10] of   ;
                2 = rr   [1..15, 1..20] of char;
        var :   1; :   2;
                            ,   [3],   [9][18],   [1,1], ,   [15],   [3,3]   [10][16].

9.3*. var A, B: array [1..15, 0..8] of real; t: boolean;
                                                                            ?

)   := ;   )   := + ;   ) t:=A<>B;
) read(A);   ) [1]:= [15];   ) [2,3]:= [4][8]+ [1,1]

9.4*.   .   4-   ,
                                        ,   .

9.5*. const n=20;
        var A, B, C: array [1..n, 1..n] of real;
            x, y: array [1..n] of real;
                :
    a) $C = A + B$;   )   = ;   )   = A ;   )   = ^{T} (   ).

9.6*. var C, D: array [0..9, -5..3] of integer; t: boolean;

t .... *true,* .... *D* .... ,
false ....

9.7. .... (...................)
.... *A* 5- .... *n*-
$(A^1=A, A^2=A\cdot A, A^3=A^2\cdot A$ ... .).

9.8*. type vector = array [1..10] of real;
           table1 = array [1..8] of vector;
           table2 = array [1..8, 1..10] of real;
       var x: vector; A: table1; B: table2;

                                                                    ?

) := ;   ) [1]:= [8];   ) [3]:= [3];
) [1]:= [8];   ) := [5];   ) [5]:=

9.9. type .... = array [1..20] of integer;
                    = array [1..20] of .... ;
     var : .... ; : .... ;
         B: array [1..20, 1..20] of integer;

                                                             :

)* .... *A* .... ;
)* .... ;
)* .... ;
) .... *A* .... 1- ..... 2- .... , 3- ..... 4- .... , ..., 19-
   20- .... (....);
) ....

9.10. .... (............) .... 7(4.
                         , ... ,
(...............) ....

9.11*. type .... = rr .... [(x,y)] of real;
       var M: array [1..40] of .... ; d: real;
                                                         , ... *d* -
                  .

9.12. type .... = (........ , ... , ... , ... , ... ,
                      , ... , ... , ... , ... , ...);
       var .... : array [....... , ...] of boolean;
            : .... ;
              [,b] .... *true,* .... *b* .... ,
false .... - .... ,
        .

9.13*. type .... = (........ , ... , ... , ... , ... , ...);
                  = (..... , ... , ... , ... , ... , ... , ... , ... , ... , ... , ... , ...);
       var t: array [....... , ...] of real;
           i: .... ; m: .... ;
         t[x,y] ....
             , ... (*m*) .... (*i*) ....

9.14*. var A: array [1..9, 1..9] of real; s: real;
       s - .... (... . 3).

9.15. var D: array [1..10, 1..10] of real; s: real;

$$s = \sum_{k=1}^{10} \max_{1 \le i, j \le k} D_{ij}$$

9.16. var A: array [1..10, 1..10] of integer;

:

)* $\begin{pmatrix} 0 & 0 & 0 & ... & 0 \\ 0 & 1 & 0 & ... & 0 \\ 0 & 0 & 2 & ... & 0 \\ & ... & & \\ 0 & 0 & 0 & ... & 9 \end{pmatrix}$) $\begin{pmatrix} 1 & 2 & ... & 10 \\ 11 & 12 & ... & 20 \\ 21 & 22 & ... & 30 \\ & ... & \\ 91 & 92 & ... & 100 \end{pmatrix}$) $\begin{pmatrix} 1 & 2 & 3 & ... & 10 \\ 0 & 1 & 2 & ... & 9 \\ 0 & 0 & 1 & ... & 8 \\ & ... & & \\ 0 & 0 & 0 & ... & 1 \end{pmatrix}$

9.17. var A: array [1..6, 1..9] of real; x: array [1..9] of real;

$: A_{ij} = x_j^i.$

9.18. type   = (,   ,   ,   ,   ,   ,   ,   ,   ,   ,   ,   );
              = (,   ,   ,   ,   ,   ,   ,   );
                  = array [, 1..31] of   ;

     var   :   ;

                                                              (
)   ,   1   -
([,1]=   ;   [,2]:=   ; ...;   [,29]:=   ; ...).

9.19. var A: packed array [1..20, 1..20] of boolean;
         B: packed array [1..19, 1..19] of boolean;
         n, k: 1..20;

                                        n-   k-   .

9.20. var A: array [1..15, 1..20] of integer;
         b: array [1..15] of boolean;

                           b,   k-   true,
                       ,   false   :
)*   k-   A   ;
)   k-   ;
) k-   .

9.21. const n=8; m=12;
      var k: integer;
         C: array [1..n, 1..m] of integer;

            k -   «   »   ,
«   »,   :

)   ;
)   ,   ,   -
о   .
9.22. var k: integer;
          C: array [1..13, 1..18] of char;
          k -   (. .
).
9.23.   .   20(30.
                        :
)*   ;
)   ;
)   .
9.24.   ,
                                                  (*n*>2):
    T: array [1..n, 1..n] of (,   ,   ,   ),
  T[i,j]=B,   *i*-   *j*-   (*T[j,i]=*), *T[i,j]=* ,   *i*-
j-   ,   *T[i,i]=X.*   (*n*=3):

$$\begin{pmatrix} X & B & \Pi \\ \Pi & X & H \\ B & H & X \end{pmatrix}$$

                              1   ,   - 0.5   ,   - 0   .
                                                                                                    .
9.25. type   = (, …,   );
                = (,   ,   ,   ,   );
      var TP: array [,   ] of   ;
            ,   ,   :   ;   k: integer;
          (   )   ([,]=   ,
                        ,   *x*' ,   [,]=   ,   -   ' ,   . .)
                        :
)*   -   (   )   ,   ;
)   -   (   )   ,
(   );
) *k* -   (   )
                        .
9.26. const n=256;
      type screen = packed array [1..n, 1..n] of 0..1;
      var S: screen;
                              S,   90°
                        .
9.27.   .   ,
10-   (   ).
9.28.   .   ,
                                                                                                        ,
                  ,   .
                                                            10(15
                        .
9.29.   .   7(7,
                                  .   ,
                                      ,   .
9.30.   .   ,

10-   , . .   ,
                                          0,
                            1.
    9.31.   .   ,
9-   , . .   ,
                                                       .

    9.32.   .
                         ;   ,   -   .
                         b_i, i -   , b_i - i-
                         ,   .
    9.33.   . a_{11}, a_{12}, …, a_{1n}, a_{22}, a_{23}, …, a_{2n},
…, a_{nn} ($n=20$, $a_{ii}(0)$ b_1, b_2, …, b_n «   »
                         :

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \quad ... \quad + a_{1n}x_n = b_1 \\ \qquad\quad a_{22}x_2 + a_{23}x_3 + \quad ... \quad + a_{2n}x_n = b_2 \\ \qquad\qquad\qquad a_{33}x_3 + \quad ... \quad + a_{3n}x_n = b_3 \\ \qquad\qquad\qquad\qquad\qquad ... \\ \qquad\qquad\qquad\qquad\qquad\quad a_{nn}x_n = b_n \end{cases}$$

    9.34.   . $_{ij}$ b_i

$$\sum_{j=1}^{n} A_{ij}x_i = b_j \qquad (i = 1, 2, ..., n; \quad n = 10)$$

             ,   0.   (:
                                     «   »   .)
    9.35.   . n n-
             (=7).   ,   .
    9.36.   . n-   ($n=6$).   ,
             ,   ,   . (:

             ,
                              .)

                         10.   :

    10.1*.   ,   ?
       const n=20;
      type   = rr   [1..30] of char;
              b = packed array [1..n] of char,
              c = packed array [0..n] of char;
              d = packed array ['1'..'9'] of char;
              e = packed array [1..n, 1..n] of char;
              f = packed array [1..10] of 'a'..'z';
              g = packed array [1..1] of char;
   10.2*. const n=5;
              var a1, a2: packed array [1..n] of char;
                 b1, b2: array [1..n] of char;

c: packed array [1..5] of char;

?

```
)  1:= 2;          ) b2:=b1;              )  1:=b1;
)   := 2;          )  2:='b1+b2';         ) b1:='12345';
 ) c:='abc';       )  1< 2;               ) b1=b2;
)  2> ;            )  > ='12345';         )  1<>'**';
 ) read(c);        ) write(a1, '=', a2);  ) writeln(b1)
```

10.3*.   .   60   .

,   .

10.4*. type   = array [1..8] of char;
                    = packed array [1..8] of char;
      var a1: array [1..20] of   ;
          a2: array [1..30] of   ;
          k1, k2: integer;
           k1 -   *1,*
      ,   *k2* -   *2,*   .

10.5. type   = packed array [1..5] of char;
                    = rr   [1..60] of   ;
      var C: ;
           :
)*   ,   *hello*;
)   ,   (   )
                               (,   );
)*   ,   ;
)   ,   *d.*

10.6*. var s: packed array [1..6] of char;
                    input   1   6   ,   .
                                         s,
          .

10.7.   .   ,   1   8
                                                -          .
                          . (,   «   »
     -         .)

10.8*. type color = (red, blue, green, yellow, black, white);
       var x: color;

                                                                              ,
                              .   *color,*
                ,   .

10.9.   :

```
program errors (input, output);
  const sign = '+-*/';
  var s: packed array [1..10] of char;
      i, j, k: integer;
begin
  read(s);
  for i:=1 to 10 do
    for j:=1 to 4 do
      if s[i]=sign[j] then k:=k+1;
  if k<6 then writeln(s) else writeln(sign)
end.
```

10.10*. const v = 'aeiou';
        type   = packed array [1..200] of char;
        var s: ; k: 0..200;
             ,   (*k*)   *s*   ,
v.
  10.11*.   .   60   .
       ,   .
  10.12.   .   ,
       .   .
  10.13.   .   ,
          .   ,   .
  10.14.   .
              ,   200   .
  10.15. type   = packed array [1..80] of char;
         var s: ;
              ,   *s*   40   ,
                    .   ,
              :
)   *abc*   *def*,
)*   *w*,   («   »
                          ,   );
)   *th*;
)*   *ks*   ,   ;
)   *q*   *u*;
)   *ph* *f*,   *ed* *ing*.
  10.16.   .   ,   1   30   ,
              1   5   ;
-          ,   -   .   :
)   ,   ;
)   ,
          (   )   ,   -   *o*   ;
)   ,
    ;
)   ,   ;
)   ,
              ;
)   .
  10.17.   .   ,   2   50   ,
              1   8   ;
-                    ,   -   .
              ,   :
)   ;
)   ;
)   ;
)   (a, ab, abc
   . .);   ,   'a'..'z'   ,
     ;
)   (z, yz, xyz
   . .);

)   ;
)   ;
)   ;
)   (, , i, , u)   .
10.18.   .   2   30   ,
                    2   10   ;   -
              ,   -   .   ,
                              ,
                    :
)   ;
)   ;
)   ;
)   ;
)   ;
)   ;
)   ;
)   ,   .
10.19.   .   1   1999
          .
10.20.   .   ,
      .   ,
                    1   999, ,   ,
                    (   ).
10.21.   .   -   (h)   (1
  8),   2   g5.
          ,   ,   ,
                    ,   «   »   ,   ,   .
10.22.   .
10.23.   ,   12
                                            :
    20.1-18.2 -   23.7-22.8   -
    19.2-20.3 -   23.8-22.9   -
    21.3-19.4 -   23.9-22.10   -
    20.4-20.5 -   23.10-22.11   -
    21.5-21.6 -   23.11-21.12   -
    22.6-22.7 -   22.12-19.1   -
                    ,
                            .
10.24.   .   60   .   ,
(   )
                            .
10.25.   .   ,   1   90   ,
                    1   10   ;
-                      ,   -   .
          .
10.26.   .
          .

11.

11.1. var x, y, z: real;

$z=(\text{sign } x+\text{sign } y)\cdot\text{sign}(x+y),$

$$\text{sign } a = \begin{cases} -1 & \text{при } a < 0 \\ 0 & \text{при } a = 0 \\ 1 & \text{при } a > 0 \end{cases}$$

:

)   sign;

)*   sign.

11.2*.   .

$\text{sh}(x)\cdot\text{tg}(x+1) - \text{tg}^2(2+\text{sh}(x-1)).$

11.3*. type   = (,   ,   ,   ,   ,   );

                    = (,   ,   );

     var , :   ; t: boolean;

                         (s),   ,

     s,   t

                    ,   .

11.4*. type   = 1..maxint;

     var a, b: real; k: ;

                              (,)   ,

          (   ) $x^n,$

$b=2.7^k+(a+1)^{-5}.$

11.5.   .   .

          .

11.6. type   = 1..maxint;

     var m, n: ;

     function   (, b:):   ;

     begin

       while a<>b do

          if a>b then a:=a-b else b:=b-a;

                :=

     end;

)*   ,   :

      m:=8; n:=6; writeln((m,n), m, n);

)   ,   (,)

                    n,   .

11.7.   :

   a) function f (a: 'a'..'z'): integer;

       begin f:=ord(a)-ord('p');   if f<0 then f:= -f end;

) function g (k: integer): 0..maxint;

           var i, s: 0..maxint;

       begin s:=0; for i:=1 to k do s:=s+sqr(i) end;

) function h (x: integer): integer;

       begin h(x):=(sqr(x)+x)/2 end;

11.8*   .   , b, c   d.   ,

                              . ((, ,z),

                         ,   z,   .)

11.9. var c, d: integer;
    procedure P (x, y: integer); begin y:=x+1 end;
    procedure Q (x: integer; var y: integer); begin y:=x+1 end;
    procedure R (var x, y: integer); begin y:=x+1 end;
)   ,
          -                ,   -   -   .
)*   ,   :
    :=2; d:=0; P(sqr(c)+c,d); writeln(d);
    c:=2; d:=0; Q(sqr(c)+c,d); writeln(d);

                                              -
                                  ?   ,
      ?
)*   R(sqr(c)+c,d)   R(c,d)?
          ,   ,   -   ?
11.10.   (a,b,p,q)   (b?0)
    /b   p/q.
)
,   -   ?   -
        ,   -   -   ?
)*   (k+1,14,n,7)   (k,sqrt(36),k,n),   k
-                    ?
)*
1+1/2+1/3+ … +1/20   c/d.
  11.11.   maxmin(x,y)
                    ,   -   .
)
,   -   ?   -
        ,   -   -   ?
)*   maxmin(5.2,sin(z))   maxmin(z,k),   z -
                , k-   ?
)*
                                  , b   ,   ? b ? c.
11.12. const n=1000;
    type   = array [1..n] of real;
    var , b, , d: ;
            sum(x,y,z)   z   .
)   ,   ,
                ,   -   .
)*   d=a+b+ .
11.13.   :
                                                      ?
                    -   ?
                            ?   ?   -
          -          ,   -   -   ?
                                                                    .

  a) type   = packed array [1..60] of char;
      var s, t: ; k: integer;
                                  s   12
      t   ' '   'z',   k-   '*',
                    s.

) var a, b: real; t: boolean;

      t   *true,* $x^2+6.2x+a^2=0$ $x^2+ax+b-1=0$

                               ,   *false*   .

) const n=10;

   type   = array [1..n, 1..n] of real;

                   = array [1..n] of real;

    var , , :   ; , , z, u: ;

       u=Ax+By- z+   .

) type   =   rr   [1..40] of integer;

   var , , z: ;

                                   10

                                                      ,

           z,   ,

   . (,   .)

11.14.   :

```
program errors (input, output);
  const n=40;
  type vector = array [1..n] of real;
  var a, b: vector;
  function sum (var x, y: vector): vector;
      var i: integer;  z: vector;
    begin for i:=1 to n do z[i]:=x[i]+y[i];  sum:=z end;
  procedure reverse (x: vector);
      var i: integer;  r: real;
    begin
      for i:=1 to n div 2 do
        begin r:=x[i]; x[i]:=x[n+1-i];  x[n+1-i]:=r end
    end;
  begin
    read(a);  b:=sum(a,reverse(a));  writeln(b)
  end.
```

11.15.   .

$ex^3 - ?x^2 - (2e+1) \ + 2? = 0$

(?=3.1415927, *e*=2.7182818),   0.0001.

11.16.   .   , *b*   .

            ,

             . (:   ,   ,

$0.5\sqrt{2b^2 + 2c^2 - a^2}$.)

11.17.   .   ,   1   6

                                                           .

                         .

11.18*.   . $a_0, a_1, \ldots, a_{30}, b_0, b_1, \ldots,$

$b_{30}, c_0, c_1, \ldots, c_{30}, x, y, z$

$$\frac{(a_0x^{30} + a_1x^{29} + \ldots + a_{30})^2 - (b_0y^{30} + b_1y^{29} + \ldots + b_{30})}{c_0(x+z)^{30} + c_1(x+z)^{29} + \ldots + c_{30}}$$

11.19.   .   20-

$$u = \begin{cases} \sum_{i=1}^{20} x_i^2 & \text{при} \quad \sum_{i=1}^{15} x_i y_i > 0 \\ \sum_{i=10}^{20} y_i^2 & \text{иначе} \end{cases}$$

11.20.   .   50-   , b

$$t = \begin{cases} \dfrac{\min(b_i)}{\max(a_i)} + \dfrac{\max(c_i)}{\min(b_i + c_i)} & \text{при} \quad \min(a_i) < \max(b_i) \\ \max(b_i + c_i) + \min(c_i) & \text{иначе} \end{cases}$$

11.21.   .   30-   , z.
                          (,)-(b,),   ,
                                                              (,   ),
b , (p,q) - q.

11.22.   .   10-   .
                                                                        ,   (
                   ),   ,   .

11.23.   :

```
a) program errors (output);
      const a='1234';
      type string = packed array [1..4] of char;
      var b: string;
      procedure P (s1: string; var s2: string);
         begin writeln(sl,s2) end;
   begin  P('abcd', 'efgh');  P(a,a);  b:=a;  P(b,b)  end.
) program error (output);
      type string = packed array [1..3] of char;
      var x: string;
      procedure Q (c: char; var d: char);
         begin d:=succ(c) end;
   begin x:='295';  Q(x[1],x[3]);  writeln(x) end.
```

11.24*.   ,   :

```
program print (output);
  type vect = array [1..2] of real;
  var a: vect;  i: integer;
  procedure R (var k: integer;  var x: real);
     begin k:=2; x:=0 end;
  begin
    a[1]:=1;  a[2]:=2;  i:=1;  R(i,a[i]);  writeln(a[1], ' ', a[2])
  end.
```

11.25*. type   = 1..19;
                         = packed array [1..20] of boolean;
                         (s,k), s,
              k , k -   .

11.26. const n=15; m=20;
        type   = array [1..n, 1..m] of real;
                        ( ),
   $x_1 x_n + x_2 x_{n-1} + \ldots + x_n x_1$,
   i- i-   .

11.27*. type   = 0..maxint;

$$F(\ ,\ )= \ !\cdot\ !/(n+\ )!, \qquad n \quad m -$$

(.)

11.28*. type   = rr   [1..20] of char;

(, , ,b)   ,

:

$$x = (a_1,\ a_2,\ \ldots,\ a_8,\ x_9,\ x_{10},\ \ldots,\ x_{20}),$$
$$y = (y_1,\ \ldots,\ y_5,\ b_1,\ \ldots,\ b_6,\ y_{12},\ \ldots,\ y_{16},\ a_1,\ \ldots,\ a_4).$$

11.29. type   = packed array [1..9] of char;

(,),   ,   ,

,   .

11.30. const n=8; m=13;

type   = array [1..n, 1..m] of real;

$swap(A,B),$

A   . (,

.)

11.31. const n=...;   {   > 1}

type   = rr   [1..n] of '0'..'9';

= rr   [1..40] of   ;

$sort(x),$

:

;

(   )

;

;   . . (,

.)

11.32*.   (   )

,   :   .

```
procedure P (x: vect;  var y: integer);
  const z='*';
  var c: index;
begin y:=0;  for c:=a to b do if x[c]>z then y:=y+1 end;
```

11.33*   ?

```
a) program print (output);
     var x, y: char;
     procedure P (x: integer);
       const   =true;
     begin  writeln(x, ' ', y) end;
     procedure Q;
       var x: char;
     begin x:=succ(y);  y:='*';  writeln(x, ' ', y) end;
   begin
     x:='a';  y:='5';
     P(8);  writeln(x, ' ', y);
     Q; writeln(x, ' ', y)
   end.
 ) program print (output);
     type string = packed array [1..5] of char;
     var i: integer;  t: string;
     procedure P (var s: string);
```

```
        begin
         i:=1; while s[i]<'9' do  begin s[i]:=succ(s[i]);  i:=i+1 end
        end;
      begin i:=1;  t:='12945';  P(t);  writeln(t[i]) end.
    ) program print (output);
      var a, b, c, d: integer;
      procedure P (var b: integer; c: integer);
        var d: integer;
      begin a:=5; b:=6;  c:=7;  d:=8;  writeln(a,b,c,d)  end;
      begin
         :=1; b:=2;  c:=3;  d:=4;  P(a,b);  writeln(a,b,c,d)
      end.
```

11.34.   :
```
   procedure errors (var x: boolean);
     const char=0; case=1;
     type a = (true, false);  b = ('a', b');
     var c: char;
   begin if x then x:=(ord(true)=char)  and  false end;
```
11.35. type   = 1..20;
                   = array [,   ] of real;
              max(A,n,k),   -   ,   *k* -   (*n<k*),
                                                                    A(. 4).

. 4

11.36. type table1 = array [1..10, 1..10] of integer;
            table2 = array [1..20, 1..30] of integer;
                  constr(A,B,C,D),   ,   *tabel1*
                        D   *table2*:

$$D = \begin{pmatrix} A & B & C \\ B & N & A \end{pmatrix}$$

  N -   *table1*.
11.37.   .   ,
               1   6   ;   -   ,
                    -        .   ,   «   »,
   . .   .   (
readword(w),   6-
      w,   .)
11.38.   .   6-
                              ,   6-   .
(,   )+(,)/(,   ).

11.39.   .   4-   .

                 ,   (,

).

               .

11.40*.   . d $(c < d)$

$$\int_c^d \operatorname{arctg}^2 x\, dx + \int_0^? \sin e^{10x}\, dx.$$

                                              $n = 20$

                       $n = 100$:

$$\int_a^b f(x)\, dx \approx h \cdot \{ f(a)/2 + \sum_{i=1}^{n-1} f(a+ih) + f(b)/2 \}$$

$h = (b-a)/n.$

11.41.   .   40-   ,

z

$$w = \begin{cases} \prod_i (\sin(x_i) + 2) & \text{при } \prod_i (1 - y_i^2) > 0.5 \\ \prod_i (1 - z_i^2) & \text{иначе} \end{cases}$$

11.42.   .

   $1/(1+x^2) = x,$ $3e^x + x = 0$ $x \ln(1+x) = 0.5$,

                        ? > 0.

11.43. const n=20;

      type   = array [1..n] of real;

                  (, ,z),   , z,

                     (,   ),

                     :   - z,

            -                .

11.44*.   ,   (,

                       ):

```
program sideeffect (output);
  var a, b: integer;
  function f (x: integer): integer;
    begin f:=x;  a:=0 end;
  function g (var x: integer): integer;
    begin g:=x;  x:=0 end;
begin
  a:=1;  write(a+f(a));  a:=1;  write(f(a)+a);
  b:=2;  writeln(g(b)=g(b))
end.
```

11.45.   ,   = ,

+ = 2* , b and $b = b$ (y b-   ).

11.46*. $next$,   .

                ,   ,

                                 k -

        ,   .

11.47. const d=100; m=5;

      type   = packed array [1..d] of char;

                a = packed array [1..m] of char;

                = 1..d;

var : ; , z: ;
no (*s,ss,k,n*), ,
ss *s,* *k*- , , ,
,
ss *s.*
,
z.

11.48. . . ,
о .
11.49. .
.
. (,
.)
11.50. . $_k x + b_k =\ _k$ (*k*
= 1, 2, 3). ,
.
11.51. .
.
11.52. . « »,
2 (, , 41 43).
« » [*n*, 2*n*], *n* - , 2.
11.53. . « »,
,
(, , 220 284).
« » , .
11.54. . ( ) *Q*( ) 15-
. $P(a+Q(a)P(a+1))$.
11.55. . >0

$$\frac{\sqrt[3]{a}-\sqrt[6]{a^2+1}}{1+\sqrt[7]{3+a}}$$
$$=\sqrt[k]{x}$$?=0.0001
( ):
$y_0 = 1;\ \ y_{n+1}=y_n + (x/y_n^{k-1} - y_n)/k$ (*n* = 0, 1, 2, ...),
y_{n+1}, $|y_{n+1} - y_n| <$?.
?>0 *t*

11.56. . ?

$$\sqrt[4]{1-\frac{\cos^4 t}{4}}+\sqrt[5]{1+\frac{\text{arctg}\,t}{2}}\cdot\sqrt[9]{\frac{1}{3+t^2}}$$

:
$$(1+x)^a = 1 + \frac{a}{1!}x + \frac{a(a-1)}{2!}x^2 + \frac{a(a-1)(a-2)}{3!}x^3 + \dots$$ (|*x*|?1, >0).

11.57. . 9(4.
, (,).
11.58. .
, 4- . (*ABC*)^p.
11.59. . , 10(20.

$$\frac{\|A\|+\|B\|+\|C\|}{\|A+B+C\|}$$

$$\|D\| = \max_{j} |D_{1,j}| + \max_{j} |D_{2,j}| + \ldots + \max_{j} |D_{10,j}|$$

11.60.   .   10-   .

,

11.61.   .   , b (($<b$, (>0).   (

(.   . 5.34)

$$\int_{a}^{b} \left(\int_{g(x)}^{f(x)} h(x,y)\,dy \right) dx$$

$h(x,y)=e^{-y}y^{x-1}$, $g(x)=1$, $f(x)=1+x^2$.

11.62.   .   (>0   -

,   . 5, $f1(x) = \dfrac{2}{x}\cos\dfrac{x}{2}$, $f2(x) = \ln(1 + x^2)$,

$f3(x) = e^{-x^2}$.

. 5

11.63.   .   1, 2, ..., k, k -   2   70,

(   )   :

((n) -   1   -1,   ;

((n) -   ;

(( ) -   , n.

11.64.   . 2^{500}

1!+2!+3!+ ... +100!. (:   «   »

.)

11.65.   ,   «   »

,   (,   5   ).

11.66.   .   (n=100).

                             : A

;   ($_1$ $_2$' $_3$ $_4$ . .)

;   ,

,   ;

A B;   . .

11.67.   . n 2   20   (>0.

$(\quad\quad\quad n- \quad\quad\quad\quad\quad\quad\quad\quad _n(\ ),$

$T_0(x){=}1;\quad T_1(x){=}x,\quad _k(\ ){=}2\ _{k-1}(\ ){-}\ _{k-2}(\ )\quad (k=2,3,\dots)$

$(\quad\quad\quad : \quad\quad\quad\quad _k(\ )\quad\quad k \quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad (-1,1);$

$x_1{<}x_2{<}\dots{<}x_k - \quad\quad\quad\quad\quad\quad\quad T_k(x),\quad\quad\quad\quad\quad\quad T_{k+1}(x)$

$(-1,x_1),\ (\ _1,\ _2),\ \dots,\ (\ _k,1).)$

12.

12.1*. function fib (n: integer): integer;
   begin if n<=1 then fib:=1 else fib:=fib(n-1)+fib(n-2) end;
   fib(2)   *fib*(4).

12.2*. $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad$ f(*n*),
   ,$\quad\quad\quad\quad\quad$?
   a) function f (n: integer): integer;
      begin f:=n*f(n-1) end;
) function f (n: integer): integer;
      begin if n=0 then f:=1 else f:=f(n+1)/(n+1) end;
) function f (n: integer): integer;
      begin if n=0 then f:=1 else f:=n*(n-1)*f(n-2) end;
) function f (n: integer): integer;
      begin if n=0 then f:=1 else f:=n*f(n-1) end;

12.3*. $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad pow(x,n) \quad\quad\quad\quad\quad\quad\quad\quad (\ ?0)$
      $_n$
   ,

$$x^n = \begin{cases} 1 & \text{при } n = 0 \\ 1/x^{|n|} & \text{при } n < 0 \\ x \cdot x^{n-1} & \text{при } n > 0 \end{cases}$$

12.4*. $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad(\ ,\ ),\quad\quad 0?\ ?\ ,$
      $\quad\quad\quad\quad\quad\quad\quad\quad C_n^m \quad\quad\quad\quad\quad\quad\quad\quad :$

$$C_n^0 = C_n^n = 1; \quad C_n^m = C_{n-1}^m + C_{n-1}^{m-1} \quad\quad 0{<}m{<}n.$$

12.5*. type $\quad = (\quad\quad,\dots,\quad\quad,\quad);$
      $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad(\ )\quad\quad\quad\quad\quad(\ ),$

      ,$\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad,$
      $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad(\ ,b),\quad\quad\quad\quad\quad\quad,$
      $b \quad\quad\quad\quad(\quad\quad\quad,\quad\quad\quad,\quad\quad\quad . .)\quad\quad\quad\quad\quad\quad\quad\quad .$

12.6. $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad,$
      $(\ ),$
      $(\ ,k),\quad\quad\quad\quad\quad\quad\quad k- \quad\quad\quad\quad\quad\quad\quad\quad\quad\quad (k$
).

12.7*. function f (n: integer): integer;
      begin if n>100 then f:=n-10 else f:=f(f(n+11)) end;
      f(106), f(99)   f(85). $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad$?

12.8. $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad root(f,a,b,eps),$
      $\quad\quad\quad\quad\quad\quad\quad\quad eps \quad\quad\quad\quad\quad\quad\quad\quad$ f(x)=0
[,b]. ($\quad\quad\quad,\quad\quad eps{>}0,\ a{<}b,\ f(a){\cdot}f(b){<}0\quad$ f(x) -

[, *b*].)

12.9*. const n=40;
type = rr [1..n] of real;
min(*x*) ,
min1(*k*),
, *k*- .

12.10. type = packed array [1..100] of char;
(*s,i,j*), ,
s, *i-* *j-*
.

12.11*.
, .
sum
.

12.12. *digits* ,
, (
).

12.13. .
( ).

12.14. . ,
0.
, - ( ).

12.15*. . ( )
:
< > ::= < > | (< > < > < >)
< > ::= + | - | *
< > ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
. (, 5(5, ((2-4)*6) (-12.)

12.16. . , .
, « » (.
).

12.17. .
:
< > ::= true | false | < > (< >)
< > ::= not | and | or
< > ::= < > | < >, < >
< > ::= < >
(*and* *or* , *not* - .)
. (,
and(or(false,not(false)),true,not(true)) (false.)

12.18. . , .
:
< > ::= < > | < > < >
< > ::= | b | (< >) | [< >] | {< >}

12.19. .
(*n*=100) :
- (,)
,
, - ? (
),

12.20. («   »).   ,

,   1 n .

,   . 6, .

(. 6,),

:   , ?

.

(

« q r», q r -   ,   ),

,   -   . (:

,

. 6, .)

. 6

12.21.   .   ,   1

(=10).   .   ,

1-   - .

i j ($i<j$,

, i- j-   ;

-              .

12.22.   .   (n=5).

.

12.23.   8   :   8   ,

«   »   .   ,   :

)   ;

)   92   .

13.   .

13.1.   :

)   ;

)*   ,   ;

)   (,   ,   );

)   (,   ,   ,   );

)   (,   ,   ,

,   );

)   (:   ,   ,   ;

:   ,   ;   );

)*   (,   ,   , 25
              :   ,   ,
).
  13.2*. type   = (,   ,   ,   );
                  = (,   ,   ,   ,   ,
                          ,   ,   ,   );
              = record   :   ;   :   end;
                          (*1*, *2*,   ),   , «   »
1   *2*,   ,   .
  13.3*. type   = packed array [1..15] of char;
                  = record   :   ;   : 1000..9999 end;
                = array [1..30] of   ;
                            (   ),
            .

  13.4.   .
)   ,   ?
)
        ?
)   ,
            ,   ,
                                      ?
)   -   (. .   *r.f*)   (*f*)
                                                                        ?
  13.5.   ,
          ,   :
)   (5, h8   . .);
)   ;
)   50-   .
  13.6. type   1 = array [(x,y)] of real;
                  2 = record x, y: real end;
      var   1:   1;   2:   2; d: real;
)   ,
          (   )   (')?
)*   *d*   *1*   *2*.
)   p1[succ(x)]   p2.succ(x) ?
  13.7*. type   = packed array [1..8] of char;
                  = record
                          ,   :   ;
                          ,   : 1..999
                    end;
        var   1,   2:   ;
                                  ,   *1*   ,
                        «   ,   .   ,   . 1,   . 5».   ,
    2   ,   17.
  13.8*. type   = record
                          : r   l;
                        : record x, y: real end
                    end;
        var :   ;
                                                ,   2.5

(0, 1.8).

, ?

) with do
begin :=2.5; :=0; :=1.8 end;
) with do
begin :=2.5; . :=0; . :=1.8 end;
) with do
begin :=2.5; with do begin x:=0; y:=1.8 end end;
) with , do
begin :=2.5; :=0; y:=1.8 end;
) with , do
begin :=2.5; :=0; y:=1.8 end

13.9. ,
, , -
.

,
: , 16 , 194- ,
9 , 5, 3, 4, 5, 2.

13.10*. type complex = record re, im: real end;
point = record x, y: real end;
var z, w: complex; p: point; re: real;

, *z, w,* *r*
:
with z do begin re:=0; im:=1 end;
w:=z; re:=2;
with z do re:=1;
with z, w do im:= -im;
with p do begin x:=re; y:=2 end

13.11. :
program errors (input, output);
type = (, b);
= record a: integer; b: char end;
var , : ; c: char;
function f (var z:): ;
var : ;
begin for p:=a to b do f.p:=succ(z.p) end;
begin
read(c);
with x do begin a:=ord(c); b:=c end;
y:=x; if x=y then y:=f(x);
with do writeln(a,x)
end.

13.12. type = record x, y: real end;
= record r, fi: real end; { r?0, -(<fi?(}
(*d*,),
d , (,*d*), .

13.13. type = record : (a, b, c, d, e, f, g, h); : 1..8 end;
(*1, 2*), ,
n1 *2.*

13.14. type = record : 0..23; , : 0..59 end;

:

)   *(t1,t2)*   ,

t1   *t2* (   );

)*   *(t,t1),*   *t1*   ,   1

             t (   );

)   *(d,t2,t1),*   *d,*

     t1   *t2*: *d=t2-t1* (,   *t2>t1*).

13.15. type   = (, , , , , , ,);

            = record : (,);   : 140..200 end;

           = array [   ] of   ;

:

)   (   ),

;

)   (   )

      ;

)   (   ),   ,

              .

13.16. type   = record   : integer;   : 1..maxint end;

          = rr   [1..20] of   ;

:

)*   (,*b*),

    b;

)   (, ,*b*),   *b*

                  ;

)   (*r*),   *r*

   ;

)   (,),

           .

13.17. type complex = record re, im: real end;

        coeff = record a, b, c: complex end;   {a?0}

         value(p,x,y),   -

     ^2+b +   .

13.18. type   = record , : 0..6 end;

        = array [1..28] of   ;

                (*r*),   ,

            r (

).

13.19. type   = 1..31;   = 1..12;   = 1..2099;

         = record :   ; :   ; :   end;

           = (, , , , , ,);

   ,   («   »),

   :

)   (*d*),   ,

           d;

)   *ep a ama(d),*   *d*

(. .   31   . .);

)   (*d*),   ,   1

   1-   *d*;

)   *H(d)*   ,

d (,   1   1-   ).

13.20. type = packed array [1..20] of char;
= record
, : ;
: record : ; , : 1..999 end
end;
= array [1..15] of ;
( ), ( )
,
.
13.21. type = packed array [1..16] of char;
= record : 1..31; : 1..12; : 1900..2000 end;
= record
: ;
:(,);
:
end;
= array [1..25] of ;
:
) (,),
(,
);
) (,), ,
,
.
13.22. type = packed array [1..9] of char;
= 1000000..9999999;
=record : ; : end;
= array [1..20] of ;
= array ['A'..'Z'] of ;
, ,
- ,
:
) (, ,), ,
, , ,
;
) (, ,), ,
, , , ,
.
13.23. const n=300;
type = record : integer; : rr [1. .99] of 'a'..'z' end;
= array [1..n] of ;
, :
) ( ),
;
) (, ,), ,
( )
, , , .
13.24. (., , [9]):
- , , ,
, ,
, ;
- , (

«   »

)   . (
,
.)

13.25.   . z (   )
(>0.   (
:

) $e^z = 1+z/1!+z^2/2!+ ... +z^n/n!+ ...$;
) sh $z = z+z^3/3!+z^5/5!+ ... + z^{2n+1}/(2n+1)! + ...$;
) ch $z = 1+z^2/2!+z^4/4!+ ... +z^{2n}/(2n)! + ...$;
) $\sin z = z-z^3/3!+z^5/5!- ... +(-1)^n z^{2n+1}/(2n+1)! + ...$;
) $\cos z = 1-z^2/2!+z^4/4!- ...+(-1)^n z^{2n}/(2n)! + ...$;
) $\ln(1+z) = z-z^2/2+z^3/3- ... +(-1)^{n-1} z^n/n + ...$ ($|z|< 1$);
) arctg $z = z-z^3/3+z^5/5- ... +(-1)^n z^{2n+1}/(2n+1) + ...$ ($|z| < 1$)

13.26.   .
.

13.27.   1
.   -   (400)
:
<   >, <   >, <   1>, <   2>, <   3>
-            12   ,   -   101   116,
-      2, 3, 4   5,   -   ,
-               ,   -   .
.
,
:
)   ,
;
) «   »   , . .   ,
5   4;
)   ,   ;
)   .

13.28.   2000
:
<   >, <   >, <   >, <   >, <   >, <   >
,   -   12   ,
,   -   16   35,   -   1   5.
.
,
:
)   ,   ;
)   ;
)   (   )   ,
.

14.

14.1. type bits = set of 0..1;
      var x: bits; y: set of (a, b, c); z: set of '*'..'*';

)   ?
)*   ,
z?

  14.2*.   ,
                                                        ?

  14.3*. type   = (,   ,   ,   ,   ,   ,   );
                                ,   :
)   ;
)   .
  14.4*.   ?
   type   = set of real;
              =   d array [1..8] of 0..1;
              = set of   ;
              = (,   ,   ,   ,   ,   ,   ,   ,   ,   ,   ,   );
        M1 = set of   ;
        M2 = set of   ..   ;
        M3 = set of   ..   ;
        M4 = set of (,   ,   );
  14.5*.   (
            ),   ?
) [9, 6, 3, 0];   ) [2..3, 5, 7];
) [1..15, 4..18];   ) ['*', '*'];
) [0..0];   ) [true..false];
) [2, sqrt(9)];   ) ['=', '>=', '>'];
) [[ ], [5]];   ) [odd(7), 0<2]
  14.6*. var p: set of 0..9;   i, j: integer;
     i=3   *j*=5,
                                           :
) p:=[i+3, j div 2, j..sqr(i)-3];
) p:=[2*i .. j];
) p:=[i, j, 2*i, 2*j]
  14.7. var s: set of char; c, d: char;
             s   :
)   ;
)   (, , i, , u);
)   ;
)   ,   ,   *d (c<d).*
  14.8*.   :
) [2] <> [2, 2, 2];   ) ['a', 'b'] = ['b', 'a'];
) [4, 5, 6] = [4..6];   ) ['c', b'] = ['c'..'b'];
) [2, 3, 5, 7] <= [1..9];   ) [3, 6..8] <= [2..7, 9];
) [ ] <= ['0'..'9'];   ) 'q' in ['a'..'z'];
) trunc(3.9) in [1, 3, 5];   ) odd(4) in [ ];
) [2] < [1..3];   ) 66 = [66]
  14.9.   :

)   in [0, 5, 19]   (p=0) or (p=5) or (p=19)?
)   in [20..50]   (p>=20) and (p<=50)?
14.10*. type   = packed array [1..100] of char;
                     (s),   '+',
'_'   '*',   s.
14.11.   .   100   1   50.   ,
                                                              ,
               - 1   2.
14.12. type   = 1..12;
                                         (   ),
(   ).
14.13*. type M = set of 0..99;
                      card(A),
       . (, card([5, 8, 23])=3.)
14.14*. type letters = set of 'a'..'z';
                     print(   ),
           ,   letters.
14.15. const n=10;
          type   = 1..n;
                       = array [,   ] of real;
                 = set of   ;
                 sum(A,s1,s2),   ,

             s1   s2   .
14.16*. type   = (,   ,   ,   ,   ,   ,   );
                             =   ..   ;
          var wd: ; t: boolean;
                           t   true,   wd -   ,
             false   .
     ?
   a) t:=wd in   ;   ) t:=wd=pa o   e   ;
) t:=wd in [   ];   ) t:=wd in [..   ];
) t:=[wd]<=[..   ];   e) t:=[wd]=[..   ]
14.17*.   :
) [1,3,5]+[2,4];   ) [1,3,5]*[2,4];   ) [1,3,5]-[2,4];
) [1..6]+[3..8];   ) [1..6]*[3..8];   ) [1..6]-[3..8];
) [2, 4]+[1..5];   ) [2, 4]*[1..5];   ) [2, 4]-[1..5];
) [ ]+[4];   ) [ ]*[4];   ) [ ]-[4]
14.18.   :
)* [2..13]*[3, 13..60] + [4..10] - [5..15]*[6];
) [2..10] - [4, 6] - [2..12]*[8..15];
) (['0'..'7']+['2'..'9']) * (['a']+['z'])
14.19.   (A   —   ):
)*   * - ;   )* A-(A-   );
) (+   )-(-   )-(-   );   ) (-   )+(-   )+   *
14.20.   ,
             -                        .
14.21*. var x, y, z: set of 8..22;
                                                              8   22,   -
                                                 ,   z -

14.22.   .   ,
            .   ,   -   (, , i, , u)   -
                    .

14.23*. var A, B: set of char; x: char;
                                              ,   :
)   ;
)   .
14.24. type   = 1..maxint;
        :
)*   *digits*(*n*),   (   )
                                              n;
)   *print*(*n*),   ,
                                                                              .

14.25.   .   ,
        .   :
)*   ,
        ;
)   ,   ;
)   ,   .
  14.26.   .   ,   .
                ( )   (, , , , , , , ,),
            .

14.27.   .
            1..10000, n^2+m^2,   , *m* (0.
14.28.   .
            1..4900, n^2+2k^2,
7*ij*+*j*+3 (*n, k, i, j* (0).
  14.29.   .   2   1000.   «
            »,
  ..2*n*.
    (:   ,   1;
                (2,   )   ,
                ;   (3,
)   ,
      ;   .
                                                  2.)
    14.30. type   = (,   ,   ,   ,   ,   ,
                                    ,   ,   ,   ,   );
                            = set of   ;
                            = array [1..20] of   ;
                                (, , ,),
                        ([*i*] -   ,   *i*-   )
                        ,   :
      A -   ,   ;
        -                                  ,   ;
        -                            ,
14.31. type   = (,   ,   ,   ,   ,   ,
                            ,   ,   ,   ,   );
                        = set of   ;

= array [   ] of   ;
( ),   ,
([ ] -
,   ; ([ ]).

14.32. type   = (a, b, c, d, e, f, g, h);
= set of   ;
= array [   ] of   ;
(, ,),   ([ ] -
,   )
-                       ,   (
)   H.

14.33.   path(G,N,K,D),   ,
G   N   , ,   ,
D   (   )   N   .
:
type   = (b1, b2, b3, b4, b5, b6, b7, b8);
= set of   ;
= array [   ] of   ;
(G[x] -   ,   .)

14.34.   :
type M = set of char;
function f (a, b: M; x: char): M;
begin
  if a*b=0 then a:=[x] else
    if a<b then a:=b+x else
      if ord(x) in a-b then a:= a-[..'<='];
  f:=a+b
end;

14.35.   .
;   -   ,   -   .
:
)   ,   ;
)   ,   ;
)   ,   ;
)   ,   ;
)   ,   ;
)   ,   ;
)   ,   ;
)   ,   ;
)   ,
;
)   ,
.
(:   - , , , , , , , , (
);   -   ,   , , ;
- , , , , , , , , , ;   - , , , , , , , , , .)

14.36.   .   ,   (
).   ,
«   »:

```
<        > ::= <     > | (<          > <     > <          >)
<    > ::= + | - | *
<      > ::= <     > | <       >
<    > ::= <       > | <     > <       > | <     > <       >
<       > ::= <        > | <       > <       >
<        > ::=   |   |   |   |   |   |
<      > ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
```

14.37.   .

-                    (*n*=15),   :

$$\det(A) = \sum_{k=1}^{n} (-1)^{k+1} a_{1k} \left(\det(A_k) \right),$$

A_k -   ,   1-   *k*-   .
(:   *l   s,*
                                                                            ,
                        l   ,
        s.)

15.

15.1.   .

)   ,
                                    ,   (   )
            ,   ?
)   ,   ,
    ?   ?
)   ,   ,   ,
                        ?
)   :   ,
        ,   ,   ,   ?
)   ,
                    :   ,   ,   ,   ?
                                        (   )?
)   ,
                ?   ,   ,   ,
                    ?   ?
)   ?

15.2*. var f: file of integer; x, y: integer;
            f   - 3   7.   ,
                                                        :

    a) reset(f); read(f, y);
        if not eof(f) then read(f, y);
        if not eof(f) then read(f, y);
) reset(f); y:=0;
        while not eof(f) do begin read(f, x); y:=y+x end;
) reset(f); y:=1;
        repeat read(f, x); y:=y*x until eof(f);

15.3. type   = file of char;
                                                                (*w*),
                                                    w.

   function   (w:): integer;
     var k: integer; c: char;
   begin
     reset(w); k:=0;
     repeat read(w, c); k:=k+1 until eof(w);
            :=k
   end;
15.4*. type   = file of real;
                              (*s*),
        s.
15.5. type   = record py : 0..maxint;   : 0..99 end;
                          = file of   ;
                     in (,   ),
                     .
 15.6*. type   = file of 0..999;
                                               (*r*),   ,
                                               r.
 15.7*. type   = file of char;
                                     eq(*t1*,*t2*),   *t1*   *t2*

        .
 15.8. type   = record   : 0..23;   ,   : 0..59 end;
               = file of   ;
                                   eq(*f*,*g*),   *f*   *g*

        .
 15.9. type   = file of char;
                                     less(*w1*,*w2*),   ,
                    w1   *w2*.
 15.10. type FR = file of real;
                                    (*f*,
               f,   *FR*   .
 15.11*. var f: file of integer; i: integer;
                                    f   :
) rewrite(f);
        if eof(f) then write(f, 1) else write(f, 2);
        if eof(f) then write(f, 3) else write(f, 4);
) rewrite(f);
        for i:=3 downto 1 do write(f, sqr(i))
 15.12*. type   = packed array [1..100] of char;
                      = file of char;
                              (*s*,*t*),   *t*   *s*.
 15.13. type   = file of 1..maxint;
                    fib(*f*,*n*),   *f*   (1, 1, 2, 3,
5, ...),   .
   15.14*. type FB = file of boolean;
                             npuc (*f*,*g*)   *FB*,   *f*
                       g.

15.15. type letters = file of 'a'..'z';

               append(f,g,h)   *letters*,
   f   *g*,   -   *h.*

15.16. type   = record

                         : (,   ,   ,   ,   ,   ,   ,
                             ,   ,   ,   );
                    : 1..31
             end;
          = file of   ;
           an(d,s,w)   ,   *d*
          s   ,   *w* -   .

15.17*. type reals = file of real;
             less(f)   *f*   *reals* ,
                      f,
       .

15.18. type   = record   :   cked array [1..9] of char;   : 1..99 end;
           = file of   ;
                (   ),
      ,   .

15.19.   .   ,   1
8   ;   -   ,   -   .
      ,   .

15.20.   .   ,   1
8   ;   -   ,   -   .
                    .

15.21. type   = file of char;
                :
)* *add1(t,c),*   *t;*
) *addlast(t,c),*   *t;*
) *double(t),*   (   )   *t*   ;
) *replace(t,c),*   *t*   ;
) *next(t),*   *t*
    ('9'   '0');
) *delete(t),*   *t*   '+'   '-';
) *del(t),*   *t*   ,   ;
) *firsts(t),*   *t*   .

15.22*. var f: file of integer; x: integer;
        f   1   2.
                      ?

  a) reset(f); if f^=1 then get(f); x:=f^;
) reset(f); x:=0;
     if not eof(f) then begin get(f); x:=x+f^end;
     if not eof(f) then begin x:=x+f^; get(f) end;
     if not eof(f) then x:=x+f^;
) reset(f); get(f); get(f); x:=f^;
) reset(f); read(f, x); if f^>1 then read(f, x)

15.23*. var t: file of char; c: char;
                       t   ?
  a) rewrite(t); if eof(t) then t^:= 'a' else t^:= 'b'; put(t);
) rewrite(t); put(t); t^:='*';

) rewrite(t); for c:='1' to '3' do begin t^:=c; put(t) end;
) rewrite(t); t^:='1'; for c:='3' downto '1' do begin put(t); t^:=c end

15.24. ( ) ,
x^[5].y+[5] .

15.25*. , ( ) *f* *g*,
, *npuc* (*f,g*), *f*
g.

15.26*. type FR = file of real;
mid(*f,m*), *f*
FR , , ,
.

15.27. type FR = file of real;
incr(*f*)

f.

15.28. type FI = file of integer;
f *g* .
h, .
merge(*f,g,h*)
FI.

15.29. type = file of char;
relation(*f,v*), ,
f « » (.), ,
, *v* .

< > ::= < > < > < >
< > ::= < | = | > | <= | <> | >=
< > ::= < > | < >
< > ::= < > < > | < > < >
< > ::= 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
< > ::= 0 | < >

15.30. *t* (*text*),
.
) *text* *file of char* ?
) ,
? ?
?
) *eoln*(*t*) *true,* *t^* ?
) *t* , ?
writeln(*output*) ?
) ,
? ? *k* - ,
read(*t,k*) *write*(*t,k*) ?
) *input* *output*?
? *reset*(*input*) *rewrite*(*output*)?
?
,
write(*input*,5) *read*(*output,x*), - ?
) *read*(*input,x*) *read*(*x*), *eof*(*input*) *eof,*
get(*input*) *get*?

15.31*. *triangle*(*t*), *t* 9
, - '1', - '2', ...,
- '9'.

15.32.   *line40*(*t*),
                                    (   )   *t*,
                    40   (
).
15.33.   ,   :
)*   *t*;
)   *t*.
15.34.   *printlines*(*t*),
                t.

15.35.   *t*   .
count(*t*)   ,   :
)*   *d*;
)   *z*,
)   ;
)   .
15.36*.   *npuc* (*t1*,*t2*),
                    t2   *t1* (   ).
15.37.   *npuc* (*t1*,*t2*),   *t1*
                t2,   .
15.38.   ,   *f*   ,
                80,   *npe*   *p*(*f*,*f80*),   ,
                    f   ,
f80,   80.
15.39*. type   = packed array [1..20] of char;
                = array [1..100] of   ;
                    (*l*,*t*),   *l*
    t.
15.40*.   *t*
                ,   .   *max*(*t*)
                              .
15.41.   *t1*   ,
                .   *positive*(*t1*,*t2*),
            t2   *t1*.
15.42*. type   = record   : 1..31;   : 1..12;   : 1900..2099 end;
        var d: ;
                d   : 25.10.1917, 22.6.1941, 1.1.2001   . .
15.43.   *lines*(*t*),
                        t,
                (4   )   .
15.44.   .   sin *x*   tg *x*
    [0,3]   0.1.   ,
                -            ,   -   .
15.45.   .   10   «   »
                :
)   1   )   1
            1   1   1 1
            1   2   1   1 2 1
            1   3   3   1   1 3 3 1
            1   4   6   4   1   1 4 6 4 1
            ...   ...
        1 9 … 126 126 … 9 1   1 9 ... 9 1

(«   »   1,   -
            .)

15.46.   .   ,
«   »   .   :
        39624
        (
        8503

        118872
    + 198120
      316992

      336922872

15.47*.   *BOOK*.   ,
        ,   ,
,   ,
            60   ,   ’   .

15.48.   .   .
                                        .

15.49.   *1*   ,
                    :

    type   = packed array [1..12] of char;
              = (,   ,   );
              = record
                        : record   ,   ,   :   end;
                        : rr   [   ] of 2..5;
                          : 101..116
                end;
          = file of   ;
              ,   *1*
        ,   ,   :
        ,   :
        ,
15.50.   .   .
                ,   *D*   ,
                                (
).

«   »
(. 7,   ,
).

                                                        D (. 7,).
        i- D (*i* = 1, 2, ...)
                                                            (. 7,).
                                                        D (. 7,)   . . (,

                        A)

16.   .

16.1. type ref = ^integer;
     var p, q: ref;
                        q   ,   . 8.
              .
)   *:*   (   )
                                                    ?   ^*:*
                        ,   5?
p^?

                                        . *8*

)*
            ?
        p^:=q^;
        if p=q then p:=nil else if p^=q^ then q:=p;
        if p=q then q^:=4;
        writeln(p^)
16.2*. type D = record a: boolean; b, c: ^real end;
         var r: ^D;
                    r   ,   . 9.
                    r   :
    if r^.b<>nil then r^.c:=r^.b;
    r^.b^:=r^.c^-1.4;   r^.a:=r^.b=r^.c

                                    . *9*

16.3*. var p, q: ^integer; r: ^char;

?

   a) p:=q;   ) q:=r;   ) p:=nil;   ) r:=nil;
) q:=p^;   e) p^:=nil;   ) r^:=p^;   ) q^:=ord(r^);
) if r<>nil then r^:=nil^;   ) if q>nil then q^:=p^;
) if q=p then write(q);   ) if q<>r then read(r^)
16.4.
   program dynamic (output);
     var x: ^boolean; y: boolean;
   begin
     {A} new(x); {B} x^:=true; y:=not x^;
     {C} dispose(x); {D} writeln(y)
   end.

                                   .

)   , ,   D
                     ?
)   (   ),   new
                       dispose,   ?
        ?
)   y?
                 dispose   ?
16.5*. type A = ^char; B = record f1: char; f2: A end;
       var p: ^B; q: A;

                                                  q

        :
   new(q); q^:='7';
   new(p); p^.f1:=succ(q^); p^.f2:= q;
16.6. type chain = ^elem;
          elem = record data: integer; link: chain end;
       var p, q: chain;

        :
) new(p); p^.data:=4; p^.link:=nil;
) new(p); p^.data:=7; p^.link:=p;
) new(q); q^.data:=2; q^.link:=nil; new(p); p^.data:=1; p^.link:=q;
)* new(p); p^.data:=5; new(p^.link); p^.link^:=p^
16.7.   :
   program errors (input, output);
     var a, b: ^integer;
   begin
    if a=nil then read(a); a^:=5;
    b:=nil; b^:=2;
    new(b); read(b^); writeln(b,b^);
    new(a); b:=a; dispose(a); b^:=4
   end.
16.8*.   ?
   type A = ^0..9;
        = record p: real; q: C end;
       C = ^B;

16.9*.   (,   ,   )
,   ,   . 10.

. 10

16.10. type   = ^   ;
               = record   : integer;   :   end;
     var :   ;
         ,   ,
  . 11, ,   ,   .: 1)* 11, ; 2) 11, ; 3) 11, . (,
      ,   .)

. 11

16.11.   p^[2], q^+[2]   r^^ ?
.
16.12. type   = ^real;
              = array [1..100] of   ;
  ,   nil,   :
)   *( )*   ,
              ;
)*   *neg1( )*,
    ,   ,   nil,
;

)   *same(x),*   ,

         ;

)   *unique(x),*   ,

      ,   .

  16.13.   «   »   -

               (   )   :

```
const d=...;  {            }
      n=...;  {                    }
type       = packed array [1..d] of char;
           = ^      ;
         = array [1..n] of      ;
```

(,   nil;

         nil   .

        , ..   nil,

        .)

                        ,   :

)   ( )   ;

)   *(T,i,j,c),*   ,

       i,   ,   ,   *j-*

  ;

)   *(T,i,j),*   *i-*   *j-*

  ;

)   (,*i,j),*   *i-*   *j-*

  ;

)   (,*i,j),*   *i-*   *T*

j-   ;

)   (,*i),*   *i-*   ;

)   (, ,*i,j),*   ,

 , ,   ,   *i* *j*«   »

       : *i -*   , *j -*   ;

)   ( ),   ;

)   ( ),

                  (,

,   ).

      16.14 - 16.29   (   )

        (. 12,)   (. 12,)

     :

```
type    = ...;  {                ( ,
                 ,              )}
        = ^    ;
      = record    :  ;    :        end;
```

$L, L1$ $L2$,   , 1 2 -
,
.

16.14.   ,   :
)   , L ;
) L (=r   l);
)* L $E1$ 2;
) L;
)*   , L $(T$ ='a'..'z');
) L,
$(T$ =integer).
16.15. type   = packed array [1..10] of char;
=   ;
, L,   :
)   ;
)   ,   ;
)   .
16.16. type   = file of   ;
=   rr   [1..50] of   ;
,   ,
:
)* f,
)   (   ).
16.17.   , L : $L1$ -
$L2$ - L(=r   l).
16.18.   ,   :
) L ;
) L ;
) L;
) L $E1$;
)* L $E1$,
E L;
) L 1 2
;
) L,   ,
,   (=real).
16.19.   ,   :
) L ;
) L ,   ;
) L ,
;
)* L ;
) L ,   (=integer);
) L (=r   l).
16.20*.   .   (   )
.

16.21.   .   ,
0.   ,
.

16.22. $n>1$,

16.23. , :
) $L1$ $L2$;
) , $L1$ $L2$;
) , L ;
) L ;
) L ;
)* $L1$ $L2$;
) L
$L1$, L;
) L, . . ,
;
) L
;
) L .

16.24. , :
)* , L;
) L;
) L ($=r$ l);
) L (T $=char$);
) L $E1$ $E2$;
)* L , ;
) L ;
) $L1$ - L;
) ( ) L;
) L
($=r$ l).

16.25. , L,
, :
) $L1$ $L2$;
) $L1$ $L2$,
) $L1$, $L2$;
) $L1$ $L2$,
.

16.26. ,
$L1$ $L2$ ($=r$ l)
:
) L;
) $L1$ $L2$
$L1$.

16.27. $(L,L1,L2)$, L
$L1$ ( ) $L2$.

16.28. const n=...; { > 1}
type = packed array [1..n] of 0..9;
= ;
$ynop(L)$,
L (. . 13,
2).
10 ( ), ,

, *k-* ,
k, *L,*
k- (*k*+1)- .
(
),   -
. .

16.29. type   = ^   ;
                = record   : 'a'..'z';   :   end;
            =   ;
                            ,   :
)   *L*   ,
L   ;
)*   *L*;
)   *L*   ;
)   *L*;
)   *L,*
        .

16.30.
    $P(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0$
                                                (. 14,),
        $_i$=0,   (. 14,
                                        $S(x) = 52x^{40} - 3x^8 + x)$.

. 14

-                         :
)   (,q),
        q;
)   (,),
              ;
)   (,q),   -
     q;
)   (p,q,r),   -
q   r;
)   (,v),
    ,
   v;   ,   S   (S,' ')

   52y^40-3y^8+y;
)   ( ),
              (-   )   -
   .

16.31.   (
)   *eof1(f)*
reset1(f), read1(f,x), rewrite1(f)   write1(f,x),
                                                                          .

16.32.   (   ) «   »
                     ,   ;   -
         :
    <   > ::= ( ) | (<   >)
    <   > ::= <   > | <   > , <   >
    <   > ::= <   > | <   >
    «   »   ,   1   *n*
   ,   -   .   :
(AD75,(3,(),(7H))).

                                                                    :
)   *member(A,L),*   ,
    L;
)   *equal(L1,L2),*   *L1*
  L2;
)   *printat(L),*   ,   *L;*

) *printlist(L)*, L , ;

) *readlist(L)*, L - .

16.33. L ( )
(. 15) :
   type 2 = ...; {   }
        2 = ^ 2;
        2 = record : 2; , : 2 end;
                                       2.

. 15

, :
)* , L ;
)* L (2=char);
) L,
« »;
) , L ,
   ( ) ;
) L
   , L ;
) L , ;
) L, ,
   , « » (
   );
) L ;
) L ( ) ;
) L L1;
) L ,
   (, 1, 2, 3
   1, 2, 3, 3, 2, 1).
16.34. (« ».) . ,
   k- , .
   .
   (-
   k), ,
   .

16.35 - 16.46 ,
   .

16.35. , (
).
16.36.
   , 0.

16.37.   ;
          -              ,   -   .
          .
16.38.   ,   1   12
                    ;   -   ,   -   .
                            ,
                                    .
16.39.   ,
          .
16.40.   ,   1   8
                    ;   -   ,   -   .
                                    :   -
          ,   -   . . (
).
16.41.   ,
          .
16.42.   ,   .
                #,
          ; k # k (   ).
                          ,   # (,
          # ##   LO#LO   HELLO).
16.43.   .
                                    !
16.44. $n>2$. n-
$_n$( ),
          $T_0(x)=1$; $T_1(x)=x$; $T_k(x)=2xT_{k-1}(x)-T_{k-2}(x)$ ($k=2, 3, \ldots$)
16.45.   (   )
                    ,
            :
          -8x^4-74x +8x^4 +5-x^3.
                                                                    ,
                          .
16.46.
                                            (.,   , [9]).

17.   ,   ,

17.1.   , . .   ,
                              ,   («   -
          »),   :
                (Q) - Q (   );
                (Q) -   , Q ;
                (Q,) - Q ;
                (Q,) - Q ,
          .

'   ,

'

(

,   *KA*(*k*),

,   *k* -   : 1 -   , 2 -

).

(-   >1):

)*   *n*   ,

'

(. 16,);   ,   '

;

)   ,   ,

'

(. 16,);

)

'

(. 16,).

. 16

17.2.   (

'

-    .   . 17.1),   ('

).

a)* type FR = file of real;

f   *FR*

f   :   -   ,

,   -   [*a*,*b*],   -   '

(*a*   *b* -

,   <*b*).

)   *f*,   ,

g,

(,
).
) type   = (, …,   );
                    = packed array [,   ] of boolean;
                  = file of   ;
                                                            ([,]=*true,*
                                                    ),

                                                                                        :
            -                                    ,   -   ,   -

  . .
    17.3.   ,   . .   ,
                                                                                    («
  -                  »),   :
                  (*S*) -   *S* (   );
                  (*S*) -   ,   *S*   ;
    ВСТЕК(*S,x*) -   *S*   ;
        СТЕКА(*S,x*) -   *S*   ,
            .

                                                            ,   ,
                ,
                      (,
                                                                                (*k*),
              ,   *k* -   : 1 -   , 2 -   ).
                              (-   >1):
)   ,
                                                                    ,
                      ,   (. 17,);

. 17

)   ,
                                                                (. 17,).
        17.4.   (=*char*,
                                                        ,   -   .   . 17.3),
                                  (   ).
)   *t,*
                                      .
)   ,   *t*
                                                                :

$< \quad > ::= < \quad > \mid < \quad > + < \qquad > \mid < \quad > - < \qquad >$

$< \quad > ::= < \quad > \mid (< \qquad >) \mid [< \qquad >] \mid \{< \qquad >\}$

$< \quad > ::= \quad \mid \quad \mid z$

)*   *f*   :

$< \qquad > ::= < \quad > \mid \quad (< \qquad >,< \qquad >) \mid$

$m(< \qquad >,< \qquad >)$

$< \quad > ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

max, a m - min.

(   )   (,

M(5,m(6,8)) (6).

)   *LOG*   ( )

:

$< \quad > ::= \text{true} \mid \text{false} \mid (\neg < \quad >) \mid (< \quad > (\ < \quad >) \mid (< \quad > (\ < \quad >)$

¬, ((,

.

(*boolean*)   .

17.5.   /   (

-    .   . 17.1   17.3),   (

).

t   ,   :

$< \quad > ::= < \quad > \mid < \qquad > < \quad >$

$< \qquad > ::= < \quad > \mid (< \quad >)$

,

:

)   ;

)   .

,   +(45-F( )*(-))   :

) 8 10; 12 16; 3 17;

) 3 17; 8 10; 12 16.

17.6.   «   »   :

$< \qquad > ::= < \quad > \mid < \quad > < \quad \pm > < \qquad >$

$< \quad \pm > ::= + \mid -$

$< \quad > ::= < \qquad > \mid < \qquad > * < \quad >$

$< \qquad > ::= < \quad > \mid < \qquad > \mid (< \qquad >) \mid$

$< \qquad > \wedge < \quad >$

$< \quad > ::= < \quad >$

$< \qquad > ::= < \quad >$

^   .

? *b*   ,

: *b*?.   :

a-b   (ab-

a*b+c   (ab*c+ (. . (ab*)c+)

a*(b+c)   (abc+* (. . (a(bc+)*)

a+b^c^d*e (abc^d^e*+

)   *value*(*postfix*),

(   ),

postfix

.

.   (   ),   (   )

,   ),

-                                    .
)   *translate*(*infix*, *postfix*),   ,
                (   )   *infix*,
                                                        postfix
.

,   .
                (   ),
postfix   ,   ,
                                ,
                                        ,   ,
            (   )   *postfix*
                    ,   (,
)   ,
                                ,   *postfix*,
                                    .
            ,   .
)   (   )   *infixprint*(*postfix*),
            (   )   ,
                postfix (.)

                    17.7 - 17.14   (. 18)
                    :
    type   = ...;   {   }
                = ^   ;
                    = record   :   ;   ,   :   end;
                        ,   *1*   *2*   ,   -   .

                                    . 18

        17.7.   (
                    -        .   . 17.1   17.3),   ,
            :
)
            (-   ,   );

)*   T;
)
(=r 1);
)
         (=r 1);
)
      ,   (=r 1);
)   T (=char);
)*   T   :   -   ,
        (   ) -   ,   ,   (
) -   ,   ,   . .
(=integer);
)   (   )
                                  ;   ,   -1;
)   n-   (
                 0-   ).
      17.8.   ,   :
)   ,   ;
)*   E   ;
)   (=r 1);
)   (=r 1);
)   (=char);
)*   T, . .
                                      ;
)   n-   (
                 0-   ).
      17.9.   equal(T1, 2),
                                  1   2.
      17.10*.   (, 1),   1 -   .
      17.11.   create(T,n),   n -   ,
                  -             ,   . 19.

                                        . 19

      17.12.   same(T),   ,
                                  (=char).

17.13.
    <   > ::= <   > | (<   ><   ><   >)
    <   > ::= + | - | *
    <   > ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

(«   -   »)   =*char*

    :   (   )

(f1 s f2) -   ,   -   *s,*   -
      f1   *f2.* (. 20   -
,   (5*(3+8)).)

   . 20   . 21

    ,   :
)   (   )   -   *T*;
)   *f*   -
  ;
)   -   ;
)   ,   -   .
17.14.   -   (. . 17.13)
    ,   ,   .
    ,   :
)*   -   ,   ,
     (f+0), *(0+f)*, *(f-0)*, *(f*1)*   *(1*f)*,   ,
    f,   ,   *(f*0)*
*(0*f)*, -   0;
)   -   ,   ,
    $((f_1 \pm f_2)*f_3)$ $(f_1*(f_2 \pm f_3))$,
    $((f_1*f_3) \pm (f_2*f_3))$ $((f_1*f_2) \pm (f_1*f_3))$;
)   -   ,   ,
    . ;
)   -   *1* -   -
   ,
  v.
17.15.
    . 17.13,
    .
   ,   . 17.13.
17.16.   ,   ,   ,
   ,   ,

, - ?

(, ( )

);

. 21.

(.) :

type   = file of   ;

, :

)   , ;

)   *f*   ;

)   , *T*;

)   *f,*   ,

.

17.17.   *PROG*   ( )

. ,

( )' 9

/ .

,

. (,

, , -

.)

,

-

.

17.18. ,

,

.

17.19.   17.17   ,

.

17.20.   17.18   ,

.

1.   .

1.1.   ) 120;   ) 64;   ) 6.38;   ) -0.7444;   ) 2.75;   ) -0.1667;
) 1.4142;   ) 3.1416;   ) 5 6;   ) -24.8 -7;   ) 1 6;   ) 1 -5
1.2.   ) -2.7;   ) 0.666;   ) 10.0
1.3.   :),   ),   ),   ),   ),   ),   )
1.6.   .
1.8.   a) a+b*x+c*y*z;   ) ((a*x-b)*x+c)*x-d;
) a*b/c+c/(a*b);   ) (+)/ 1* 2/(-);
) 1E4*alpha-3.2*beta;   ) (1+ /2+ /6)/(1+2/(3+ *))

1.9.   ) $\dfrac{p+q}{r+s} - \dfrac{pq}{rs}$) $10^3 + \dfrac{\beta}{x_2 - n\delta}$

1.10. 32.0
1.12. 7   ; (x+0.5)*(y+0.7)-0.75
1.14.   ) sqrt(1+sqr(x));   ) sqr(cos(x*x*x));
) ln(/5)/ln(2);   ) arctan(x/sqrt(1-sqr(x)))
1.15. a) 1/x;   ) sqr(sqr(x));   ) exp(100*ln(x));   ) exp(ln(1+x)/3)
1.16. exp(1);   4*arctan(1)
1.17. sin(3.1415927*x/180)
1.20. a) y:=1+x*(1+x/2*(1+x/3*(1+x/4)))
1.21.   ) d:=sqrt(sqr(x1-x2)+sqr(y1-y2));
) p:=(a+b+c)/2; d:=sqrt(p*(p-a)*(p-b)*(p-c))
1.25. r:=x; x:=y; x:=r;
1.27. a) 6;   ) 7;   ) 6;   ) 6;   ) -1;   ) -2;   ) 1;   ) -1
1.28. d:=x-trunc(x)
1.29. a) 3;   ) 2;   ) 5;   ) 0;   ) 0;   ) 2;   )   ) -
1.31. (-(a mod b))+((a div b)*c)
1.32.   ) 2
1.33.   :),   ),   )
1.35.   :),   ),   )
1.36.   )   : k -   ,
            k mod 3, k -   ,
                             (cos(0) -   ),   .
1.37. h:=k div 100 mod 10

2.

2.1.   a) true;   ) false;   ) true
2.2.   a) k mod 7=0;   ) sqr(b) - 4*a*c<0;
) sqr(x-1)+sqr(y)>sqr(r);   ) sqr(round(sqrt(n)))=n
2.3.   ),   ),   ) - true;   ),   ) - false
2.4.   ) (>0) and (<1);   ) (>=) and (x>=z);
) (<) or (x<z);   ) a or b;   ) a and b
2.5.   )   a   not a   a and (not a)
            ----
             true   false   false
             false   true   false

2.6.   .

2.7.   ),) -   ;   ) -   ;   ),) -

                                (not 2=5

                   (not 2)=5,   >0 or =4 -   >(0 or y)=4); e) -   or

          .

2.8.   a) (a and b) or ((not) and b)

2.10.   ) (x>=2) and (x<=5) or (abs (x)<=1);

) (x<-1) or (x>1) and (x<2) or (x>5);

) (y mod 400 = 0) or (y mod 4 = 0) and (y mod 100 <> 0)

2.11.   )   .   . 22

                     . 22   . 23

2.12.   ) (sqr(x)+sqr(y)<=1) and (abs(x)+abs(y)>=1)

2.13.   ),) - true;   ),) - false

2.14. a) false;   ) true;   ) false

2.15. a) odd(n)=odd(k);   ) a<>b (ord(a)+ord(b)=1)

2.16.   )   .   . 23

2.17.   )   --

a	b	a<=b	not a	not a or b
true	true	true	false	true
true	false	false	false	false
false	true	true	true	true
false	false	true	true	true

2.18. a) not a and b

2.19. a) true

3.

3.1.   1=1.0   2=-2.0

3.2. program   (input, output);

     var x1, x2, p, q: real;

   begin

    read(x1,x2);   {   }

    p:=-(x1+x2); q:=x1*x2; {   }

    writeln('   :', 1.0, , q)

   end.

3.5.   ,   (*a*)

                (5.0).

3.7. 3 2 3

3.8.   +1

3.10. program   (input, output);
     var a, a3, a6: integer;
    begin
     read(a); a3:=a*a*a; a6:=sqr(a3);
     writeln(a); writeln(a3,' ',a6); writeln(a6,' ',a3,' ',a)
    end.
3.11.   2.71828   3.14159.
3.12. program   (output);
     const e=2.71828; pi=3.14159;
    begin writeln(exp(pi)>exp(e*ln(pi))) end.
3.15.   )   d:=sqr(d)   ,
        .
)   ord   ( )   .
ord(x)=k   ,   .
)   ,   .
)   .
)   -   (
                              ,   b,   ).

   4.   :   ,   ,   ,

4.1.   a) if (x>0) and (x<2) then y:=sqr(cos(x)) else y:=1-sin(sqr(x));
) if abs(a)<=3.14159/2 then x:=exp(sin(a)-1);
) if (a>b) and (a>c) then d:=a else
           if b>c then d:=b else d:=c;
) if x<0 then if x>y then z:=x else z:=y
               else if x<y then z:=x else z:=y
4.3.   a) c:=y div 100; if   mod 100 <> 0 then c:=c+1;
) d:=sqr(b)-4*a*c; t:=d>=0;
         if t then begin
                  d:=sqrt(d); a2:=2*a;
                  x1:=(-b+d)/a2; x2:=(-b-d)/a2
               end;
) if a<b then begin r:=a; a:=b; b:=r end;   {a>=b}
         if a<c then begin r:=a; a:=c; c:=r end;   {a>=c}
         if b<c then begin r:=b; b:=c; c:=r end;   {b>=c}
4.4.   a) 1;   ) 2;   ) 0
4.5.   if w<>0 then
         begin if cos(w)/sin(w)<0.5 then w:=-w end
       else w:=1
4.7.   b:=x=y
4.8.   if a then x:=true else if b then x:=c else x:=false
4.11.   :)   else «;»;)   then else;
)   «;» end;)   ;)   «1:» end.
4.13. 6
4.14. a) *1-   :*
    {   a>b   (,b)=   (-b,b) }
        a1:=a; b1:=b; {   b}
    1:   if 1=b1 then goto 2;
        if a1>b1 then a1:=a1-b1 else b1:=b1-a1;
        goto 1;
    2:   c:=a1;
         2-   :
    {   a>b>0   (,b)=   (mod b,b);   (0,b)=b}

```
        a1:=a;  b1:=b;
    1:   if (a1=0) or (b1=0) then goto 2;
         if a1>b1 then a1:=a1 mod b1 else b1:=b1 mod a1;
         goto 1;
    2:   if a1=0 then c:=b1 else c:=a1;
       ) p:=1;  i:=2;
     9:  p:=p*(1-1/sqr(i));
         i:=i+1; if i<=n then goto 9
4.18.  program max (input, output);
       label 4, 9;
       const n=50; {                              }
       var x, max: real;  {x -              , i -          ,}
           i: integer;      {max -                                    }
      begin
       read(max);  i:=1;
    4:  i:=i+1;  if i>n then goto 9;
        read(x);  if x>max then max:=x;
        goto 4;
    9:  writeln('max=', max)
      end.



                              5.

5.1.   {                                    :}
        f:=1;  i:=2;  while i<=10 do begin f:=f*i;  i:=i+1 end;
        {                                    :}
        f:=1;  i:=2;  repeat f:=f*i;  i:=i+1 until i>10;
        {                               :}
        f:=1;  for i:=2 to 10 do f:=f*i
5.2.   a) 0.2;    ) 0.0;    ) 1.0;    ) 1.0
5.3.   a) a1:=a; b1:=b; {                           b}
         while (a1<>0) and (b1<>0) do
           if a1>b1 then a1:=a1 mod b1 else b1:=b1 mod a1;
         if a1<>0 then c:=a1 else c:=b1;
        ) n:=0; repeat n:=n+1; u:=cos(cos(n)/sin(n)) until u<0;
        ) p:=1; for i:=2 to n do p:=p*(1-1/sqr(i));
        ) y:=cos(40); for n:=39 downto 1 do y:=cos(n+y)
5.4.   x:=1; for n:=1 to k do x:=n*x+1/n
5.5.   {                        [l, r]=[?, 3?/2-?]: tg(l)<l, tg(r)>r}
       l:=3.14; r:=4.71;
       repeat
         x:=(l+r)/2; {                    [l, r]}
         if sin(x)/cos(x)<x then l:=x   { [l, r]:=[x, r] }
                            else r:=x   { [l, r]:=[l, x] }
        until r-l<1  -5;
        :=(l+r)/2  {            -                           [l, r] }
5.9.   program index (input, output);
        var x, i, min, k: integer;
          {x   i -                        ; min   k -
                                        }
       begin
        read(min);  k:=1;  read(x);  i:=2;
        while x<>0 do
         begin
```

```
           if x<min then begin min:=x;  k:=i end;
             read(x);  i:=i+1
          end;
        writeln('                           - ', k)
     end.
```
5.11. a) y:=1; for i:=2 to 11 do y:=y*x+i;
) y:=11; for i:=10 downto 1 do y:=y*x+i
5.13. a) y:=1; for i:=2 to n do y:=y*(2*i-1)
5.15. a) y:=0; u:=1; {u = x^i =x*x^(i-1)}
          for i:=1 to 30 do begin u:=x*u; y:=y+cos(u) end;
) y:=1; f:=1; {f = i! =(i-1)!*i}
          for i:=2 to n do begin f:=f*i; y:=y+f end
5.16. a) g:=1; f:=1; {g=f_0, f=f_1}
          for n:=2 to 40 do
            begin
             {g=f_{n-2}, f=f_{n-1}} h:=g; g:=f; f:=h+g; {g=f_{n-1}, f=f_n}
            end
5.17. a) y:=1; {   } n:=0; {   }
          u:=1; {n-e   = x^n/n! = x^{n-1}/(n-1)!*(x/n)}
          repeat n:=n+1; u:=u*x/n; y:=y+u until abs(u)<eps
5.19. y:=0; for i:=0 to 10 do y:=y+sin(1+0.1*i)
5.22. 3   (-
).
5.23. s:=0;
          for k:=trunc(ln(x))+1 to trunc(exp(x)) do s:=s+sqr(k)
5.25. a) t:=true; y:=1; {y=x^n=x*x^{n-1}}
          for n:=1 to 30 do begin y:=x*y; if sin(y)<0 then goto 15 end;
          t:=false;
     15: …
) y:=1; n:=0;
          repeat n:=n+1; y:=x*y; t:=sin(y)<0 until t or (n=30)
5.26. {   n   k   [2, sqrt(n)],
                    }
          p:=true; k:=2;
          while p and (sqr(k)<=n) do if n mod k = 0 then p:=false else k:=k+1
5.29. a) p:=1;
          for i:=1 to 20 do for j:=1 to 20 do p:=p/(i+sqr(j))
5.30. k:=0;
          {d1 -   , d2 -   , d3 -   }
          for d1:=1 to 9 do
            for d2:=0 to 9 do
              for d3:=0 to 9 do
                if d1+d2+d3=n then k:=k+1

                                6.

6.1.   :),   ),   )
6.2.   ) '6';   ) true;   ) 5;   ) true;   ) true; e) true;   ) 23;   ) '6'
6.3.   a) t:=d='*';   ) t:=(d='a') or (d='q');   ) t:=(d>='0') and (d<='9')
6.4.   a) '+';   ) ' '
6.5.   s:=ord('S')+ord('U')+ord('M')
6.6.   writeln(chr(65), chr(71), chr(69))
6.7.   if dig='9' then next:='0' else next:=succ(dig)
6.8.   b:=ord('z')-ord('a')=25

6.9. for c:='A' to 'Z' do write(c); writeln
6.10. a) for d:='1' to '9' do
        begin
          for c: ='1' to pred(d) do write('0'); write(d);
          for c:=succ(d) to '9' do write('0');   writeln
        end
6.11. program ab (input, output);
         var c: char; {   }
             ka, kb: integer; {   b}
        begin
          ka:=0; kb:=0; read(c);
          while c<>'.' do
            begin
              if c='a' then ka:=ka+1 else if c='b' then kb:=kb+1;
              read(c)
            end;
          writeln(ka>kb)
        end.
6.14. program   (input, output);
         var c: char; t: boolean;
        begin
          t:=false;
          read(c); if (c='+') or (c='-') then read(c);
          if (>='0') and (<='9') then
            begin
              repeat read(c) until (c<'0') or (c>'9');
              t:=c='.'
            end;
          if t then writeln ('   ') else writeln ('   ')
        end.
6.16. a) program   (input, output);
          var : char;
         begin
           read(c);
           repeat
             if (c='+') or (c='-') then write(c,c) else
             if (c<'0') or (c>'9') then write(c);
             read(c)
           until c='.';
           writeln
         end.
) program   (input, output);
          var a, b: char; {   -   , b -   }
         begin
           read(a,b);
           while b<>'.' do
             begin
               if (a<>'+') or (b<'0') or (b>'9') then write(a);
               a:=b; read(b)
             end;
           writeln(a)
         end.
6.19. a) program   (input, output);
         var c: char; k: integer;
        begin

```
          k:=0;
           repeat {                     }
             read(c); {                      }
             if c='a' then k:=k+l;
           {                              :}
             repeat read(c) until (c=',') or (c='.')
           until  ='.'; {                           }
           writeln ('           "  "              ', k, '      ')
         end.
```

6.20. n0:=ord('0');
    k:=100*(ord(c2)-n0)+10*(ord(c1)-n0)+(ord(c0)-n0)

6.21. n0:=ord('0');
    d:=k div 100; c2:=chr(n0+d);
    d:=k mod 100 div 10; c1:=chr(n0+d);
    d:=k mod 10; c0:=chr(n0+d)

6.22. {var c: char; sign, n0: integer;}
    {   :}
     sign:=1; read(c);
     if c='-' then begin sign:= -1; read(c) end else if c='+' then read(c);
    {   (-   )   k
                      :}
     n0:=ord('0'); k:=0;
     repeat k:=10*k+ord(c)-n0; read(c) until c=' ';
    {   :} k:=sign*k

6.30. a) for i:=1 to k-1 do write(' '); writeln('*');   {   : writeln('*': k) }
) if k>n then max:=k else max:=n;
         for i:=1 to max do
           if i=k then write('*') else
             if i=n then write('I') else write (' ');
          writeln

                    7.   .

7.1.   )   : 4), 5);
) 1) true; 2) true; 3) false; 4)   ; 5)   ; 6)   ;
        7)   ; 8)   ;
) 1) 1; 2) 4;
)   ;
)   : 1), 2), 3)
7.2.   ,   ,   .
7.3.   )   : 2), 3);
)   : 1), 3), 6);   5)   -   ;
)   ;
)
7.4.   ,   ,   ,   ,   .
7.5.   t:=(m1<m2) or (m1=m2) and (d1<d2)
7.6.   a) if m=   then m1:=   else m1:=succ(m);
) m1:=m;
         for i:=1 to k mod 12 do
           if m1=   then m1:=   else m1:=succ(m1);
) m1:=   ; for i:=1 to n-1 do m1:=succ(m1)
7.11   a) p=false, d=3;   ) p=true, d=235;   ) p=true, d=1;   )

7.13. case m of
        ,   ,   : s:=   ;
        ,   ,   : s:=   :
        ,   ,   : s:=   ;
        ,   ,   : s:=
end
7.16. case c of
        : writeln('   ');
      : writeln('   ');
      : writeln('   ')
end
7.18. {var ch: char;}
read(ch);
case ch of
  'a': x:=a; 'b': x:=b; 'c': x:=c; 'd': x:=d
end
7.23. case m of
       ,   ,   ,   ,   ,   ,   : d:=31;
      : d:=28;
      ,   ,   ,   : d:=30
end
7.26. a) {var m1:   ;}
{   :}
  k:=0; m1:=   ;
  while m1<m do
   begin
    case m of
        ,   ,   ,   ,   ,   : k:=k+31;
        : k:=k+29;
        ,   ,   ,   : k:=k+30
    end;
    m1:=succ(m1)
  end;
{   m:} k:=k+d

8.   :

8.1.   :   ) 30;   ) real;   ) [1], [30]
        b:) 5;   ) , , z;   ) b[-2], b[2]
           :   ) 10;   )   ;   ) ['0'], ['9']
        d:) 3;   )   0   23;   ) d[   ], d[   ]
8.2. type R = array [char] of 1..maxint;
8.4.   ,   ,   .
8.5.)   :=1;
    for i:=1 to n do y:=y*x[i];
    y:=exp(ln(abs(y))/n);
) y:=x[1]; for i:=2 to n do if x[i]>y then y:=x[i];
) y:=0; p:=1;
    for i:=1 to n do begin y:=y+p*x[i];   := - end;
) y:=0; for i:=1 to n do y:=y+x[i]*x[n+1-i];
) y:=0; for i:=0 to n div 2 do y:=y+sqr(x[2*i+1]);
  e) y:=1; p:=0;
    for i:=n downto 1 do begin p:=p+x[i]; y:=y*p end

8.7. a) {var max: integer;   1:   ;}
       max:=139;
       for   1:=   to   do
        if ([1]=   ) and ([1]>max) then begin max:=   [1];   := 1 end
8.8.   for i:=1 to 72 do t[i]:=k[t[i]]
8.12. i:=0; repeat i:=i+1 until (i=n) or (a[i]<>b[i]);
       if a[i]=b[i] then c:=a else for i:=1 to n do c[i]:=a[i]+b[i]
8.13. program   (input, output);
        const n=100; d=6;
        var x: array [1..n] of integer; i, k: integer;
       begin
       {   :}
        for i:=1 to n do read(x[i]);
       {   d   :}
        k:=0; {   }
        for i:=n downto 1 do
         begin
          k:=k+1; write(x[i]);
          if k=d then begin k:=0; writeln end
         end;
        if k<>0 then writeln
       end.
8.14. for i:=0 to 39 do y[i]:=x[i+1]
8.15.   )   ).
8.16. a) program A (input, output); {   }
          const n=30; {   }
          var x, y: array [1..n] of real;
             i: integer; s1, s2: real;
         begin
          for i:=1 to n do read(x[i]);
          for i:=1 to n do read(y[i]);
          s1:=0; {   } s2:=0; {   }
          for i:=1 to n div 2 do
           begin s1:=s1+x[2*i-1]*y[2*i-1]; s2:=s2+x[2*i]*y[2*i] end;
          writeln(s1/s2)
         end.
) program   (input, output); {   }
           const n=30;
           var i: integer; x, y, s1, s2: real;
          begin
           s1:=0; s2:=0;
           for i:=1 to n do
            begin
             read(x,y); {   x[i]   y[i]}
             if odd(i) then s1:=s1+x*y else s2:=s2+x*y
            end;
           writeln(s1/s2)
          end.
8.18. program   (input, output);
        const n=80;
        var x: packed array [1..n] of char; c: char; i: integer;
       begin
        k:=0; (}
        for i:=1 to n do
         begin

```
        read(c);
    {         -          ,        -              :}
        if (c>='0') and (c<='9') then write(c)  else begin k:=k+1; x[k]:=c end
      end;
    {                      :}
      for i:=1 to k do write(x[i]); writeln
    end.
8.21.  a) p:=1;  {                                    }
        repeat write(s[p]); p:=2*p until p>n;
        writeln
8.26.    ) for k:=1 to n div 2 do begin r=x[k]; x[k]:=x[n+1-k]; x[n+1-k]:=r end;
         ) r:=x[1];  for k:=1 to n-1 do x[k]:=x[k+1];  x[n]:=r
8.28.  p:=1; {                                    z}  i:=1; {     }  j:=1; {     }
      {                                           ,      :}
      repeat
        if x[i]<y[j] then begin z[p]:=x[i]; i:=i+1 end else begin z[p]:=y[j];j:=j+1 end;
        p:=p+1
      until (i>k) or (j>m);
      {                                  —>           z
                                              :}
       if i>k {                    }
        then repeat z[p]:=y[j]; j:=j+1; p:=p+1 until j>m
        else repeat z[p]:=x[i]; i:=i+1; p:=p+1 until i>n
8.33.  a) i:=1;
        repeat t:=x[i]<x[i+1]; i:=i+1  until (not t) or (i=n)
8.35.  {label 9;  var   1,   2:            ;}
              :=true;
       for    1:=          to pred (            ) do
        for   2:=succ(  1) to            do
         if      511[  1]=      511[  2] then goto 9;
                :=false;
     9:   ...
8.38.  a) for k:=n downto 2 do
          begin {       m -          max x[1..k]:}
           m:=1;
           for i:=2 to k do if x[i]>x[m] then m:=i;
      {                            x[m]    x[k]:}
           r:=x[k];  x[k]:=x[m]; x[m]:=r
          end
8.39.  k:=0;  l:=1;  r:=n;
      {                  x[l..r]:}
      repeat
        m:=(l+r) div 2; {                      l..r}
        if x[m]=p then k:=m else {          l..r     m+1..r       l..m-1}
         if p>x[m] then l:=m+1 else r:=m-1
      until (k<>0) or (l>r)
8.42.  a) R[0]:=P[0];
        for i:=1 to n do R[i]:=P[i]-a*P[i-1];
        R[n+1]:= -a*P[n]
```

9.   :

9.1.   .
9.3.   :),   ),   ).

9.4. program   (input, output);
      const n=4;
      var A: array [1..n, 1..n] of real; i, j: integer;
      begin
      {   :}
        for i:=1 to n do for j:=1 to n do read(A[i,j]);
      {   :}
        for j:=1 to n do begin for i:=1 to n do write(A[i,j],' '); writeln end
      end.
9.5. a) for i:=1 to n do for j:=1 to n do C[i,j]:=A[i,j]+B[i,j];
) { y[i]=?A[i,j]*x[j] }
        for i:=1 to n do
          begin s:=0; for j:=1 to n do s:=s+A[i,j]*x[j]; y[i]:=s end;
) { C[i,j]=?A[i,k]*B[k,j] }
          for i:=1 to n do
            for j:=1 to n do
              begin
                s:=0; for k:=1 to n do s:=s+A[i,k]*B[k,j]; C[i,j]:=s
              end;
) {B[i,j] <--> B[j,i], j>i}
          for i:=1 to n-1 do
            for j:=i+1 to n do
              begin r:=B[i,j]; B[i,j]:=B[j,i]; B[j,i]:=r end
9.6. {label 1;}
      t:=false;
      for i:=0 to 9 do
        for j:= -5 to 3 do
          if A[i,j]<>B[i,j] then goto 1;
      t:=true;
    1: ...
9.8.   :),),).
9.9. a) for i:=0 to 9 do A[2*i+1]:=x;
) for j:=1 to 10 do for i:=1 to 20 do A[i,2*j]:=x[i];
) for i:=1 to 6 do for j:=1 to 20 do B[i,j]:=x[j]
9.11. d:=0;
      for i:=1 to 39 do
        for j:=i+1 to 40 do
          begin
            r:=sqr(M[i,x]-M[j,x])+sqr(M[i,y]-M[j,y]); if r>d then d:=r
          end;
      d:=sqrt(d)
9.13. {var i1:   ; m1:   ;}
      i:=   ; m:=   ;
      for i1:=   to   do
        for m1:=   to   do
          if t[i1,m1]>t[i,m] then begin i:=i1; m:=m1 end
9.14. a) s:=A[1,1]+A[9,1]+A[1,9]+A[9,9];
        for i:=2 to 8 do s:=s+A[1,i]+A[9,i]+A[i,1]+A[i,9]
9.16. a) for i:=1 to 10 do A[1,i]:=0;
        for i:=2 to 10 do begin A[i]:=A[1]; A[i,i]:=i-1 end
9.20. a) for k:=1 to 15 do
          begin
            j:=0; repeat j:=j+1 until (A[k,j]<>0) or (j=20);
            b[k]:=A[k,j]=0
          end

```
9.23.  a) program sort (input, output);
          const n=20; m=30;
          type        = array[1..m] of real;
                       = array [1..n] of        ;
           var  :          ;  :          ;  i, j, k: integer;
         begin
         {        :} for i:=1 to n do for j:=1 to m do read(A[i,j]);
         {                      (   .8.38, ):}
          for k:=n downto 2 do
            begin
         {      j -        max A[1..k,1]:}
            j:=1; for i:=2 to k do if A[i,1]>A[j,1] then j:=i;
         {              k-    j-        :}
            x:=A[k]; A[k]:=A[j];  A[j]:=x
            end;
         {           :}
           for i:=1 to n do
            begin
             writeln (i,'-            : ');
             for j:=1 to m do write(A[i,j] ,' ');
             writeln
            end
         end.
9.25.  a) {label 1;  var  ,    :      ;}
          {          -          ,                -          }
          for  :=      to        do
            if (    [  ,  ]=     ) or (    [  ,  ]=     ) then
              for    :=      to       do
               if    [  ,  ]=      then begin  :=   ; goto 1 end;
       1:  …
```

10.   :

```
10.1.                                       b.
10.2.                            :  ),   ),   ),   ),   ),   ),   ),   ).
10.3.   program                    (input, output);
          const n=60;
          var s: packed array [1..n] of char;  i: integer;
         begin
          for i:=1 to n do read(s[i]);
          writeln(s); writeln(s)
         end.
10.4.    {                    1:}
         k1:=0;
         for i:=2 to 20 do
           begin
            j:=0;  repeat j:=j+1 until (a1[i,j]<>a1[1,j]) or (j=8);
            if a1[i,j]=a1[1,j] then k1:=k1+1
           end;
         {                    2:}
         k2:=0;
         for i:=2 to 20 do if a2[i]=a2[1] then k2:=k2+1
10.5.    a) for i:=1 to 60 do if C[i]<>'hello' then write(C[i],' '); writeln;
          ) for i:=1 to 60 do write(C[i,5]); writeln
```

10.6.   {var c: char; i: integer;}
         {   s   ,
                                                :}
          s:='   ';
          i:=0; read(c);
          while <>'.' do begin i:=i+1; s[i]:=c; read(c) end
10.8.   {type string = packed array [1..6] of char;
            var n: array [0..5] of string;
                y: string; i: integer; c: char; ok: boolean;}
         {   :}
          n[0]:='red   '; n[1]:='blue '; n[2]:='green ';
          n[3]:='yellow';   n[4]:='black '; n[5]:='white ';
         {   6   :}
           :='   '; i:=0; read(c);
          while (<>' ') and (i<6) do begin i:=i+1; y[i]:= ; read(c) end;
          ok:=(i>0) and (c=' '); {   1   6   ?}
          if ok then
            begin {   y=n[i],   x:=i-e   color:}
             i:= -1; repeat i:=i+1 until (y=n[i]) or (i=5);
             if y<>n[i] then ok:=false else
                begin x:=red; for i:=0 to i-1 do x:=succ(x) end
            end;
          if not ok then writeln ('   color')
10.10.   {var v1: packed array [1..5] of char; i, j: integer;}
             k:=0; v1:=v; {   v   v1,   v[j] -   }
             for i:=1 to 200 do
               begin
                j:=0; repeat j:=j+1 until (s[i]=v1[j]) or (j=5);
                if s[i]=v1[j] then k:=k+1
               end
10.11.   program   (input, output);
             const n=60;
             var p: packed array [1..33] of char; c: char; i, j: integer;
            begin
               :='   ';
             for i:=1 to n do
              begin
                 read(c);
                 j:=0; repeat j:=j+1 until (c=p[j]) or (j=33);
                 if c=p[j] then write(c)
                end;
              writeln
             end.
10.15.   ) {   s   'w'   :}
             i:=1; while (s[i]<>'w') and (s[i]<>' ') do i:=i+1;
             if s[i]='w' then
           {   'w'   1
                                                 : }
               repeat i:=i+1; s[i-1]:=s[i] until s[i]=' ';
) {   s   ' '   :}
             i:=1; while (s[i]<>'x') and (s[i]<>' ') do i:=i+1;
             if s[i]='x' then
               begin
               s[i]:='k'; {   ' '   'k'}
           {   's'   'k'   :}

```
      b:='s'; {                           }
      repeat
        i:=i+1; {                 }
        c:=s[i]; {                        b}
        s[i]:=b; {           b}
        b:=   {                           }
      until b=' '
   end
```

11.

11.1.   ) function sign (a: real): integer;
         begin
           if a<0 then sign:= -1 else
             if a>0 then sign:=1 else sign:=0
         end;
          . . .
         z:=(sign(x)+sign(y))*sign(x+y)

11.2.   program shtg (input, output);
         var x, y: real;
         function sh (z: real): real;
           var e: real;
         begin e:=exp(z); sh:=(e-1/e)/2 end;
         function tg (z: real): real;
           begin tg:=sin(z)/cos(z) end;
         begin
           read(x);
           y:=sh(x)*tg(x+1)-sqr(tg(2+sh(x-1)));
           writeln(y)
         end.

11.3.   function   (s:):   ;
         begin
           case s of
                        ,   ,   :   :=   ;
                   :   :=   ;
                   ,   :   :=
           end
         end;
          . . .
         if   ( )<>   ( ) then t:=not t

11.4.   function   (: real; n:): real;
         var i: integer; y: real;
         begin y:=1; for i:=1 to n do y:=y*x;   :=   end;
          . . .
         b:=   (2.7,k) + 1/   (a+1,5)

11.6.   a) 2   8   6

11.8.   program   (input, output);
         var a, b, c, d: real;
         procedure   (,   , z: real);
           var p: real;
         begin
           if (x+y>z) and (y+z>x) and (z+x>y) then
             begin p:=(x+y+z)/2; writeln(sqrt(p*(p-x)*(p-y)*(p-z))) end
         end;
```

```
        begin
         read(a,b,c,d);  {a,b,c,d>0}
                 ( ,b, );              (a,b,d);
           e    o  (a,c,d);            (b, ,d)
        end.
11.9.    ) 0   7;    )                R(sqr(c)+c,d)                .
11.10.   )                                   ;
         ) procedure        (a, b: integer; var p, q: integer);
            var a1, b1, f: integer;
          begin
            a1:=abs(a); b1;=abs(b);
            while (a1<>0) and (b1<>0) do
               if a1>b1 then a1:=a1 mod b1 else b1:=b1 mod a1;
             if a1=0 then f:=b1 else f:=a1;
             p:=a div f;  q:=b div f
           end;
            . . .
           c:=0;  d:=1;  {c/d=0}
           for i:=1 to 20 do      (c*i+d, i*d, c, d)   {c/d + 1/i = (c*i+d)/(d*i)}
11.11.   )                                   ;
         ) procedure maxmin (var x, y: real);
            var r: real;
          begin if x<y then begin r:=x; x:=y; y:=r end end;
            . . .
           maxmin(a,b); maxmin(a,c); {a=max}
           maxmin(b,c); {c=min}
11.12.   ) procedure sum (var   ,  , z:          );
              var i: integer;
           begin for i:=1 to n do z[i]:=x[i]+[i] end;
             . . .
           sum(a,b,d); sum(d,c,d)
11.18.  program                (input, output);
          const n=30;
          type      = rr   [0..n] of real;
          var  , b,  :      ; x, y, z, d: real;
          procedure       (var v:     ); {                }
             var i: integer;
           begin for i:=0 to n do read(v[i]) end;
          function        (var v:    ; t: real): real;
            {                                    v         t}
              var s: real;  i: integer;
           begin  s:=v[0]; for i:=1 to n do s:=s*t+v[i];        :=s  end;
          begin
              ( );         (b);         ( );  read(x,y,z);
          d:=(sqr(    (a,x))-      (b,y)) /       (c,x+z);
           writeln(d)
          end.
11.24.   0.0   2.0
11.25.   procedure       (var s:         ; k:       );
             var i: integer;  t:          ; {                            }
          begin
           {t[n-k+1..n]:=s[1..k]:}  for i:=1 to k do t[n-k+i]:=s[i];
           {t[1..n-k]:=s[k+1..n]:}  for i:=k+1 to n do t[i-k]:=s[i];
            s:=t
          end;
```

11.27. function F (m, n:): r   l;
     function fact (k:):   ;
       var i, p: integer;
     begin p:=1; for i:=2 to k do p:=p*i; fact:=p end;
    begin F:=fact(n)*fact(m)/fact(n+m) end;
11.28. procedure npeo   (var , , , b:);
    procedure   (var w, v: ; n, d: integer); { w[n..n+d-1]:=v[1..d] }
      var i: integer;
     begin for i:=1 to d do w[n-1+i]:=v[i] end;
    begin
          (x, ,1,8);   (,b,6,6);   (, ,17,4)
    end;
11.32.   : , , z, ;   : vect, integer, index, a, b
11.33. a) 8 true; a 5;   6 *; a *
) 9
) 6 6 7 8; 6 2 3 4
11.40. program integrals (input, output);
     var c, d: real;
    function f1 (x: real): real;
     begin f1:=sqr(arctan(x)) end;
    function f2 (x: real): real;
     begin f2:=sin(exp(10*x)) end;
    function int (function f (x: real): real; a, b: real; n: integer): real;
     var h, s: real; i: integer;
     begin
      h:=(b-a)/n; s:=(f(a)+f(b))/2;
      for i:=1 to n-1 do s:=s+f(a+i*h);
      int:=h*s
     end; {of int}
    begin
     read(c,d);
     writeln(int(f1,c,d,20)+int(f2,0,3.1415927,100))
    end.
11.44. 2 1 false
11.46. function next: char;
     var c: char;
    begin repeat read(c) until c<>' '; next:=c end;
     . . .
    k:=0; while next<>'.' do k:=k+1

12.

12.1. fib(2)=2; fib(4)=5.
12.2.   :).
12.3. function pow (x: real; n: integer): real;
    begin
     if n=0 then pow:=1 else
      if n<0 then pow:=1/pow(x,abs(n))
         else pow:=x*pow(x,n-1)
    end;
12.4. function C (m, n: integer): integer;
    begin
     if (m=0) or (m=n) then C:=1 else C:=C(m,n-1)+C(m-1,n-1)
    end;

12.5. function   (, b:): boolean;
           var f, m: ;
           begin
             if (=b) or (=   ) or (b=   ) then   :=false else
               begin
                 f:=   (b); m:=   (b);
                 if (a=f) or (a=m) then   :=true
                 else {   -   }
                   if   (,f) then   :=true
                                         else   :=   (,m)
               end;
           end;
12.7.

$$f(n) = \begin{cases} n-10 & \text{при } n > 100 \\ 91 & \text{иначе} \end{cases}$$

12.9. function min (var   :   ): real;
           function min1 (k: integer): real; {min x[k..n]}
             var m: real;
           begin
             if k=n then min1:=x[n] {min x[n..n]=x[n]}
             else {min x[k..n] = min(x[k], min x[k+1..n])}
               begin
                 m:=min1(k+1);
                 if   [k]<m then min1:=x[k] else min1:=m
               end
           end; {of min1}
         begin min:=min1(1) end;
12.11. function sum: real;
           var x: real;
         begin
           read(x); {1-   }
           if x<0 then sum:=0 else
           {   = 1-   +   }
               sum:=x+sum
         end;
12.15. program formula (input, output);
           function F: integer;
           {F   ,   ,
                                                 }
             var c, op: char; x, y: integer;
           begin
             read(c);
             if (c>='0') and (c<='9')
               then {   } F:=ord(c)-ord('0')
               else {   (   )}
                 begin
                   x:=F; read(op); y:=F;
                   case   of
                     '+': F:=x+y; '-': F:=x-y;   '*': F:=x*y
                   end;
                   read(c) {   ')' }
                 end
           end; {of F}
         begin writeln(F) end.

13. .

13.1.　) type 　　 = record 　 : 0..23;　　 ,　　 : 0..59 end;
　　　) type 　　 = packed array [1..12] of char;
　　　　　　 =
　　　　record
　　　　　　　　 :　　 ;
　　　　　　　　　 : integer;
　　　　　　 : record 　 : 1..31;　 : 1..12;　 : integer end;
　　　　　　　 : array [1..25] of
　　　　record
　　　　　　　　 :　　 ;
　　　　　　　　　 : integer;
　　　　　　 : 2..5
　　　　　 end
　　　　end;
13.2.　function 　　 (var 　1, 　2:　　 ;　 :　　): boolean;
　　　begin
　　　 if 　1. = 　2. then 　 := 　1. > 　2. else 　 := 　1. =
　　　end;
13.3.　procedure 　　　　 (var 　 :　　);
　　　 var m, i: integer;
　　　begin
　　　 m:=1;
　　　 for i:=2 to 30 do if 　[i].　　 > 　[m].　　 then m:=i;
　　　 writeln(C[m].　　)
　　　end;
13.6.　　) d:=sqrt(sqr(p1[x]-p2.x)+sqr(p1[y]-p2.y))
13.7.　with 　 1 do
　　　 begin
　　　　　 :='　　 ';　　 :='　　 ';　 :=1;　　　　 :=5
　　　 end;
　　　　 2:= 　1;　 2.　　　 :=17
13.8.　　　　 :),),).
13.10.　z.re=z.im=1;　w.re=0,　w.im= -1;　p.x=p.y=2;　re=2
13.14.　) procedure 　　 (var t, t1:　　);
　　　 begin
　　　 t1:=t;
　　　 with t1 do
　　　　 if 　 <59 then 　 := 　 +1 else
　　　　 begin
　　　　　 :=0;
　　　　 if 　 <59 then 　 := 　 +1 else
　　　　　 begin 　 :=0;　 :=(　+1) mod 24 end
　　　　 end
　　　 end;
13.16.　a) function 　　 (var a, b:　): boolean;
　　　 begin 　 := . 　*b. 　=b. 　* . 　 end;

14.

14.1.　　)　　　　　　　　　　　　　　 :
　　　　 : [ ], [0], [1], [0, 1];

y: [ ], [a], [b], [ ], [a, b], [a, c], [b, c], [, b, c];
z: [ ], ['*']

14.2.　2^n

14.3.　type A = set of 　　　　　; B = set of 　　.. ;

14.4.　　　　　　　　　　　　　　　　　　,　　　　,　　,　*4.*

14.5.　　　　　　　　:　), 　), 　).

14.6.　　) [2, 5, 6]; 　　) [ ]; 　　) 　　　　　.

14.8.　　), 　), 　), 　) - false; 　　), 　), 　), 　), 　), 　) - true; 　　), 　) - 　　　　.

14.10.　function 　　　　(var s:　　　　): integer;
　　　　　var i, k: integer;
　　　　begin
　　　　　k:=0;
　　　　　for i:=1 to 100 do if s[i] in ['0'..'9', '+', '-', '*'] then k:=k+1;
　　　　　　　:=k
　　　　end;

14.13.　function card (A: M): integer;
　　　　　var p, k: integer;
　　　　begin
　　　　　k:=0;
　　　　　for p:=0 to 99 do if p in A then k:=k+1;
　　　　　card:=k
　　　　end;

14.14.　procedure print (A: letters);
　　　　　var c: char;
　　　　begin
　　　　　for c:='a' to 'z' do if 　in A then write(c);
　　　　　writeln
　　　　end;

14.16.　　　　　　　:　), 　).

14.17.　　) [1..5]; 　　　) [ ]; 　　　) [1,3,5]; 　　　) [1..8]; 　　　) [3..6]; 　　　) [1,2];
　　　　) [1..5]; 　　　) [2,4]; 　　) [ ]; 　　　　) [4]; 　　　) [ ]; 　　　) [ ]

14.18.　　) [3..5, 7..10, 13]

14.19.　　) [ ]; 　　) 　*

14.21.　　:=[8..22]; 　:=[11,13,17,19]; z:=x-

14.23.　　) 　:= 　+[ ]; 　　) 　:= 　-[ ]

14.24.　a) function digits (n: 　　　): 　　　;
　　　　　　var sd: set of 0..9; d: 0..9; k: integer;
　　　　　begin
　　　　{ 　　　　　　　　　　　　　　　n 　　　　　　　　　　　　　　sd:}
　　　　　sd:=[ ];
　　　　　repeat d:=n mod 10; sd:=sd+[d]; n:=n div 10 until n=0;
　　　　{ 　　　　　　　　　　　　　　sd:}
　　　　　k:=0;
　　　　　for d:=0 to 9 do if d in sd then k:=k+1;
　　　　　digits:=k
　　　　end;

14.25.　a) program 　　　　　　　　(input, output);
　　　　　　var let: set of 'a'..'z'; c: char;
　　　　　begin
　　　　　let:=[ ]; { 　　　　　　　　　　　　　　　　　　　　　}
　　　　　read(c);
　　　　　while c<>'.' do
　　　　　　begin
　　　　　　　if not(c in let) then {1- 　　　　　　　} begin write(c); let:=let+[c] end;
　　　　　　read(c)

```
                end;
              writeln
            end.
```

15.

15.2.   ) 7;   ) 10;   ) 21

```
15.4.   function        (var s:        ): real;
           var sum, x: real;
         begin
           reset(s); sum:=0;
           while not eof(s) do begin read(s,x); if x<0 then sum:=sum+x end;
                 :=sum
         end;
15.6.   function        (var r:      ): boolean;
           var x, y: 0..999;  ok: boolean;
         begin
           reset(r);
           read(r,y);  ok:=true;
           while not eof(r) and ok do begin x:=y;  read(r,y);  ok:=x<y end;
               :=ok
         end;
15.7.   function eq (var t1, t2:        ): boolean;
           var c1, c2: char;  ok: boolean;
         begin
           reset(t1); reset(t2); ok:=true;
           while not eof(t1) and not eof(t2) and ok do
              begin read(t1,c1);  read(t2,c2);  ok:=c1=c2 end;
           eq:=ok and eof(t1) and eof(t2)
         end;
15.11.  a) 1, 3;     ) 9, 4, 1
15.12.  procedure        (var s:        ; var t:        );
           var i: integer;
         begin
           rewrite(t);
           for i:=1 to 100 do if s[i] in ['0'..'9'] then write(t,s[i])
         end;
15.14.  procedure        (var f, g: FB);
           var b: boolean;
         begin
           reset(g);  rewrite(f);
           while not eof(g) do begin  read(g,b);  write(f,b)  end
         end;
15.17.  function less (var f: reals): integer;
           var k: integer;  x, s: real;
         begin
         {                                    :}
           reset(f);  k:=0; s:=0;
           repeat read(f,x); k:=k+1; s:=s+x until eof(f);
           s:=s/k;
         {                  f                  < s :}
           reset(f); k:=0;
           repeat  read(f,x);  if x<s then k:=k+1  until eof(f);
           less:=k
         end;
```

15.21. a) procedure add1 (var t: ; c: char);
           var d: char; s: ; {   }
         begin
       {   t   s:}
           reset(t); rewrite(s);
           while not eof(t) do begin read(t,d); write(s,d) end;
       {   s   t:}
           reset(s); rewrite(t);
           write(t,c);
           while not eof(s) do begin read(s,d); write(t,d) end
         end;
15.22. a) 2;   ) 4;   )   ;   ) 2
15.23.) ' ';   )   ;   ) '1', '2', '3';   ) '1', '3', '2'
15.25. procedure   (var f, g:);
         begin
           reset(g); rewrite(f);
           while not eof(g) do begin write(f,g^); get(g) end
         end;
15.26. function mid (var f: FR; var m: real): boolean;
           var k, i: integer;
         begin
       {   k -   f:}
           reset(f); k:=0;
           while not eof(f) do begin k:=k+1; get(f) end;
           mid:=odd(k);
           if odd(k) then {   }
             begin reset(f); for i:=1 to k div 2 do get(f);   m:=f^ end
         end;
15.31. procedure triangle (var t: text);
           var c, d: char;
         begin
           rewrite(t);
           for d:= '1' to '9' do begin for c:= '1' to d do write(t,d); writeln(t) end
         end;
15.33. a) function empty (var t: text): integer;
             var k, d: integer;
           begin
             reset(t); k:=0;
             while not eof(t) do {   }
               begin
                 d:=0; {d -   }
                 while not eoln(t) do begin d:=d+1; get(t) end;
                 if d=0 then k:=k+1;
                 get(t) {   «   »}
               end;
             empty:=k
           end;
15.35. a) function count (var t: text): integer;
             var k: integer; c: char;
           begin
             reset(t); k:=0;
             while not eof(t) do {   }
               begin
                 {   :} read(t,c); if c='d' then k:=k+1;
                 {   :} readln(t)

```
                  end;
               count:=k
             end;
15.36.  procedure          (var t1, t2: text);
           begin
             reset(t2); rewrite(t1);
             while not eof(t2) do
               if eoln(t2) then begin writeln(t1); get(t2) end
                           else begin write(tl,t2^); get(t2) end
           end;
15.39.  procedure      (var l:          ; var t: text);
            var i: integer;
           begin
             rewrite(t);
             for i:=1 to 100 do writeln(t, l[i])
           end;
15.40.  function max (var t: text): real;
            var m, x: real;
           begin
             reset(t); read(t,m);
             while not eof(t) do
               begin read(t,x);  if x>m then m:=x end;
             max:=m
           end;
15.42.  with d do writeln(        :1,'.',          :1, '.',      )
15.47.  program formatting (BOOK);
           const d=60; {         «           »           }
           var BOOK, COPY: text;  c: char;  k: integer;
           begin
           {            BOOK    COPY                  «                »:}
             reset(BOOK); rewrite(COPY);
             while not eof(BOOK) do
               if eoln(BOOK) then readln(BOOK)
               else begin read(BOOK,c);  write(COPY,c) end;
           {                  BOOK                    :}
             reset(COPY); rewrite(BOOK);
             k:=0; {k -                                  }
             while not eof(COPY) do
               begin
                 read(COPY,c); write(BOOK,c); k:=k+1;
                 if (c='.') or (k=d) then begin  writeln(BOOK);  k:=0 end
               end
           end.
```

16.   .

16.1.   ) 4
             .   . 24

. 26

16.8.   type D = 0..9; A = ^D;
          C = ^B; B = record p: real; q: C end;
16.9.   ) type C = ^char; A = ^R;
           R = record f1: C; f2: A end;
        var p, q: A; x: C;
         . . .
        new(p); new(q);
        new(x); x^:='a';   ^.f1:=x; p^.f2:=q;
        new(x); x^:='b'; q^.f1:=x; q^.f2:=nil
16.10. 1) q:=p;   := ^.   ; dispose(q)
16.12.   ) function neg1 (var :   ):   ;
            var i: integer; t: boolean;
          begin
            i:=0; repeat i:=i+1; t:=x[i] ^<0 until t or (i=100);
            if t then neg1:=x[i] else neg1:=nil
          end;
16.14.   ) {   }
        procedure   (L: ;   1, 2:   );
         var :   ; {   }
        begin
         p:=L;
         while p<>nil do
          begin
           if ^.   = 1 then ^.   := 2;
            := ^.   {   }
          end
        end;
) {   }
        function   (L:): boolean;
         var , q: ; {   } ok: boolean;
        begin
         ok:=true;
          :=L^.   ; {nil   1-   }
         if p<>nil then
          begin
            q:=p^.c e ; {nil   2-   }
            while (q<>nil) and ok do
             begin

```pascal
              ok:=p^.        <= q^.      ;
              p:=q; q:=q^.        {                  .        }
          end
      end;
          :=ok
      end;
16.16.   ) {                                    }
      function newlist (var f:        ):          ;
        var L, p, q:          ;   :    ;
      begin
        reset(f);
        new(L); {                              }
        p:=L; {                                          }
        while not eof(f) do
          begin
            read(f,x);
            new(q); q^.      :=  ; {                    }
            p^.c e :=q; {                    }
            p:=q  {                      }
          end;
           ^.      :=nil; {            nil                  }
        newlist:=L
      end;
16.18.   ) {                                }
      procedure            (L:        ;   ,  1:     );
        var   , q:          ; eq: boolean;
      begin
      {                    :}
        p:=L; eq:=false;
        while (p<>nil) and not eq do if  ^.      =     then eq:=true else   := ^.      ;
        if eq then {              -->              E1            }
          begin
      {               -    :            E1                                        :}
              ^.       :=  1;  new(q);  q^.        :=E;
            q^.c e :=p^.c e ;  p^.c e :=q
          end
      end;
16.19.   ) {                                    }
      procedure            (L:          );
        var   , q:          ;
      begin
      {                          .( )              .(q)              :}
        p:=L;  q:=p^.        ;
        while q^.c e <>nil do begin p:=q;  q:=q^.        end;
      {                              :}
        dispose(q);   ^.       :=nil
      end;
16.20.   program reverse (input, output);
        type          = ^       ;
                    = record        : char;          :              end;
        var L,  :            ; c: char;
      begin
      {
                  L (                              ):}
        L:=nil; {                                        }
```

```pascal
      read(c);
      while c<>'.' do
       begin {                              :}
         new(p);   ^.      := ;   ^.      :=L; L:=p;
          read(c)
        end;
   {                   L:}
      while L<>nil do begin write(L^.      ); L:=L^.       end;
      writeln
    end.
```

16.23. e) {   }
```pascal
      procedure append (L1, L2:         );
       var   1:          ;
      begin
   {                     L:}
       p1:=L1; while   1^.      <>nil do   1:=  1^.      ;
   {                   L1              1-          L2:}
       p1^.      :=L2^.
      end;
```

16.24. {   }
```pascal
      a) function memb (L:         ;   :    ): boolean;
         begin
          if L=nil then memb:=false else
           if L^.      =   then memb:=true else memb:=memb(L^.      ,  )
          end;
      e) procedure delete (var L:        ;   :    );
          var  :         ;
         begin
           if L<>nil then
            if L^.      =   then {          1-          }
              begin p:=L; L:=L^.       ; dispose(p) end
            else {              «        »                      1-
                                      «        »:}
               delete(L^.c  e  ,E)
          end;
```

16.29.   ) {   }
```pascal
      procedure firstletters (L:        );
      begin
        while L<>nil do
         begin
          if L^.       <>nil then write(L^.       ^.       );  L:=L^.
         end;
        writeln
      end;
```

16.33. a) function empty (L: 2): boolean;
```pascal
         begin empty := L^.       =L end;
        ) procedure revprint (L:         2);
          var   :          2;
         begin
           :=L^.       ; {                          }
          while p<>L do begin write(p^.       );   := ^.       end;
          writeln
         end;
```

a) type = record , : 0..n; : array [1..n] of end;
procedure (var Q:);
 begin Q. :=1; Q. :=0 end;
function (var Q:): boolean;
 begin :=Q. =0 end;
procedure (var Q: ; :);
 var i: integer;
 begin
 with Q do
 begin
 if =n then { }
 if =1 then { } (1)
 else { } for i:= to do [i- +1]:= [i];
 := +1; [ ]:=
 end
 end;
procedure (var Q: ; var :);
 begin
 if (Q) then (2);
 with Q do
 begin
 := [ ];
 if = then { 1 } (Q) else := +1
 end
 end;
17.2. a) procedure (var f: FR; a, b: real);
 var Q1, Q2: ; { =real} : real;
 begin
 reset(f); (Q1); (Q2);
 { < , Q1 [a,b] Q2 > b:}
 while not eof(f) do
 begin
 read(f,x);
 if x<a then write(x,' ') else
 if x<=b then (Q1,) else (Q2,)
 end;
 { Q1 Q2:}
 while not (Q1) do begin (Q1,); write(x,' ') end;
 while not (Q2) do begin (Q2,); write(x,' ') end;
 writeln
 end;
17.4.) function formula (var t: text): integer;
 var S: ; c, op, x, y: char;
 begin
 reset(t); (S);
 while not eof(t) do
 begin
 read(t,c);
 { ('(' ','):}
 if in ['0'..'9', 'M', 'm'] then BCTEK(S,c) else
 if c=')' then { (,)}
 begin
 { , ,

```
                                                              :}
                    (S,  );              (S,  );            (S,  );
            case op of
              'M' {max}: if x>y then c:=x else c:=y;
              'm' {min}: if x<y then c:=x else c:=y
            end;
            BCTEK(S,c)
          end
      end; {of while}
    {                                    -                              ;
                                  :}
                (S,c);  formula:=ord(c)-ord('0')
      end;
17.7.     ) function count ( :          ;   :           ): integer;
          var S:     ; {      =         } k: integer;
        begin
                (S);
        k:=0; {                       }
        while T<> nil do
          begin {   -                                }
           if   ^.     =    then k:=k+1;
        {                                      :}
            if T^.     <> nil then {                   }
             begin {              ,            , -                   :}
              if T^.       <> nil then BCTEK(S,T^.        );
                 :=  ^.
              end
            else
             if T^.        <> nil then {               :} T:=T^.
             else {                --->                                  :}
               if            (S) then T:=nil {              } else          (S,T)
          end; {of while}
        count:=k
      end;
      ) procedure levels ( :          );
         var Q:          ; {      =        } n, k, k1: integer;
        begin
         if T <> nil then
          begin
                  (Q);
           n:=0;            (Q,T); k:=1;
            repeat {               }
      {          Q                 (k)            (                  ) n-                     .
                                  ,                        Q                      (k1)
                              , . .             (n+1)-             .}
            write (n, '-             : ');
            k1:=0;
            for k:=1 to k do
             begin
                       (Q,  ); write (' ',   ^.      );
              if T^.      <> nil then begin            (Q,  ^.    ); k1:=k1+1 end;
              if T^.         <> nil then begin            (Q,T^.        ); k1:=k1+1 end
             end;
            writeln; {               n-           }
            n:=n+1; k:=k1 {                          }
```

```
                until k=0
              end { T<> nil }
            end;
17.8.      ) function count ( :          ; E:        ): integer;
              var k: integer;
           begin
             if T=nil then count:=0 else
              begin
                if   ^.      =    then k:=1 else k:=0;
                count:=k+count(T^.  e ,E)+count(T^.        ,E)
              end
            end;
           ) function height (T:             ): integer;
              var h1, h2: integer;
           begin
             if T=nil then height:= -1 else
              begin
                h1:=height(T^.    ); h2:=height(T^.       );
                if h1>h2 then height:=h1+1 else height:=h2+1
              end
            end;
17.10.  procedure       ( :         ; var   1:         );
         begin
           if T=nil then T1:=nil else
            begin
              new(T1);   1^.       := ^.      ;
                   (  ^.    ,  1^.    ); copy(T^.       ,T1^.      )
            end
         end;
17.14  a) procedure          (var  :          );
          {          .            :                        ,                    ,  T:=nil}
           begin
             if   <> nil then begin       (  ^.   );       (  ^.     ); dispose(T); T:=nil end
           end;
           procedure             (var  :          );
            var T1, T2:          ;
           begin
            if   ^.       in ['+', '-', '*'] then
              begin
             {                                        :}
                        (  ^.    );               (  ^.     );  1:= ^.    ;   2:== ^.      ;
             {                             :}
               if (  ^.      ='+') and (  1^.      ='0') or (  ^.      ='*') and (T1^.      ='1")
                 then { (0+f), (1*f) --> f } begin dispose(T1); dispose(T); T:=T2 end
                 else
               if (  ^.       in ['+', '-']) and (  2^.      ='0') or (  ^.      ='*') and (  2^.      ='1)
                 then { (f+0), (f-0), (f*1) --> f } begin dispose(T2); dispose(T); T:=T1 end
                 else
               if (  ^.       ='*') and ((  1^.      ='0') or (  2^.      ='0'))
                 then { (0*f), (f*0) --> 0 }
                    begin       (  ^.   );          (T^.     );   ^.      :='0' end
              end
            end;
```

1.   . .,   . .,   . .   . -   .:
       , 1988. - 320 .
2.   . .,   . .,   . .,   . .
               . -   .:   , 1988: - 224 .
3.   ,   .,   . .   . -   .:
       , 1987. - 112 .
4.   . .,   . .,   . .,   . .
                            . -   :   ,
  1985. - 80 .
5.   . .,   . .,   . .
                        . -   .;   , 1985. - 216 .
6.   .   . -   .:   , 1982. - 382 .
7.   .,   .   :   . -   .:   ,
  1993. - 256 .
8.   . .,   . .   . -
  .:   , 1986. - 272 .
9.   . .,   . .,   .   . -   .:
      , 1980. - 608 .
10.   . .   . -   .:   ,
  1988. - 224 .
11.   .   .   . -
  .:   , 1987. - 232 .
12.   . .   . -   .:   , 1978. - 208 .
13. *ISO 7185:1990 Programming Languages - PASCAL.* - http://www.pascal-central.com/standards.html