

9-11

1.

20

```

:
program solve_01;
  const data: array[0..9] of string =
    ('00000000','00000110','01011011','01001111',
     '01100110','01101101','01101111','00000111',
     '01111111','01101111');
  var fin, fout: text;
      n,i,j: integer; a,res: string;
      s: longint;
BEGIN
  AssignFile(fin, 'input.txt'); reset(fin);
  readln(fin, n);
  s := 0; res := '';
  //
  for i := 1 to n do begin
    readln(fin, a);
    j := 0;
  //
    while a <> data[j] do inc(j);
  //
    s := s + j; res := res + IntToStr(j);
  end;
  CloseFile(fin); AssignFile(fout, 'output.txt');
  rewrite(fout);
  //
  write(fout, res, ' ',s); CloseFile(fout);
END.

```

2.

20

```

:
program solve_02;
  var fin, fout: text;
      n,a,b,i,max,count: integer;
      res: array of integer;
BEGIN
  AssignFile(fin, 'input.txt'); reset(fin);
  readln(fin, n);
  readln(fin, a); max := 0;
  count := 0; SetLength(res, count);
  //
  //
  //
  //
  for i := 2 to n do begin
    readln(fin, b);

```

res

```

//
//
  if max = abs(a-b) then begin
    inc(count); SetLength(res, count);
    res[count-1] := i;
  end;
//
//
  if max < abs(a-b) then begin
    count := 1; SetLength(res, count); max := abs(a-b);
    res[count-1] := i;
  end;
  a := b;
end;
//
AssignFile(fout, 'output.txt'); rewrite(fout);
for i := 0 to count-1 do writeln(fout, res[i]-1, ' ', res[i]);
CloseFile(fout);
END.

```

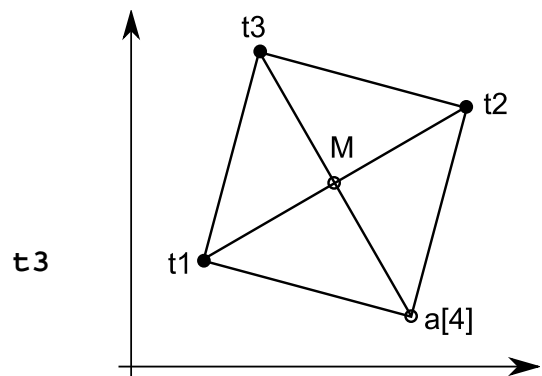
3.

20

```

:
program solve_03;
  type coord = record x,y: real; end;
  var fin, fout: text;
      a: array [1..4] of coord;
      i: integer;
//
//          t1   t2
// -          t1t2
//
function M(t1,t2,t3: coord): coord;
  var b1,b2,k1,k2,x,y: real;
      tmp: coord;
begin
  //k1
  (
  if t2.x-t1.x <> 0 then begin
    k1 := (t2.y-t1.y)/(t2.x-t1.x);
    b1 := (t2.x*t1.y-t1.x*t2.y)/(t2.x-t1.x);
    k2 := - 1/k1;
    b2 := t3.y-k2*t3.x;
    x := (b1-b2)/(k2-k1);
    y := k2*x+b2;
    tmp.x := x; tmp.y := y;
    result := tmp;
  end else begin
//
    tmp.x := t1.x; tmp.y := t3.y;
    result := tmp;
  end;
end;
//
function dist(t1,t2: coord): real;
begin

```



```

    result := sqrt(sqrt(t2.x-t1.x)+sqrt(t2.y-t1.y));
end;
    var tM: coord;
BEGIN
    AssignFile(fin, 'input.txt'); reset(fin);
    for i := 1 to 3 do readln(fin, a[i].x, a[i].y);
    CloseFile(fin);
//
    if round(dist(a[2],a[1])*1000) = round(dist(a[3],a[1])*1000)
        then begin
            tM := M(a[3],a[2],a[1]);
            a[4].x := 2*tM.x-a[1].x;
            a[4].y := 2*tM.y-a[1].y;
        end;
    if round(dist(a[1],a[2])*1000) = round(dist(a[3],a[2])*1000)
        then begin
            tM := M(a[1],a[3],a[2]);
            a[4].x := 2*tM.x-a[2].x;
            a[4].y := 2*tM.y-a[2].y;
        end;
    if round(dist(a[1],a[3])*1000) = round(dist(a[2],a[3])*1000)
        then begin
            tM := M(a[1],a[2],a[3]);
            a[4].x := 2*tM.x-a[3].x;
            a[4].y := 2*tM.y-a[3].y;
        end;
//
    AssignFile(fout, 'output.txt'); rewrite(fout);
    writeln(fout,a[4].x:0:2, ' ', a[4].y:0:2);
    CloseFile(fout);
END.

```

4. ,

20

```

:
program solve_04;
    var fin, fout: text;
        folders: array of string;
        i,n,count: integer;
        s: string; flag: boolean;
//
function folder(a: string): string;
    var j: integer; tmp: string;
begin
    tmp := ''; j := length(a);
    while a[j] <> '\' do dec(j);
    dec(j);
    if j = 2 then tmp := 'ROOT' else
        while a[j] <> '\' do begin
            tmp := a[j] + tmp; dec(j);
        end;
    result := tmp;
end;
end;

```

```

//
//
procedure toFolders(a: string);
  var k,p: integer; tmp: string;
begin
  k := 0;
  //
  while (uppercase(folders[k]) < uppercase(a)) and
    (k < count-1) do begin inc(k); end;
  //
  if uppercase(folders[k]) <> uppercase(a) then begin
    inc(count); SetLength(folders, count);
    if uppercase(folders[k]) >= uppercase(a) then begin
      for p := count-1 downto k+1 do begin
        folders[p] := folders[p-1];
      end;
      folders[k] := a;
    end else folders[count-1] := a;
  end;
end;

BEGIN
  AssignFile(fin, 'input.txt'); reset(fin);
  readln(fin, n); count := 1; SetLength(folders, count);
  //
  readln(fin, s); folders[0] := folder(s);
  //
  for i := 1 to n-1 do begin
    readln(fin, s);
    s := folder(s);
  //
    toFolders(s);
  end;
  CloseFile(fin);
  //
  AssignFile(fout, 'output.txt'); rewrite(fout);
  writeln(fout, count);
  for i := 0 to count-1 do writeln(fout, folders[i]);
  CloseFile(fout);
END.

```

5.

20

```

:
program solve_05;
  type coord = record x,y: word; end;
  var fin, fout: text;
      n,d,i,j,count,kx,ky: integer;
      a: array of coord;
      b,c: coord;
BEGIN
  count := 1; SetLength(a, count);
  AssignFile(fin, 'input.txt'); reset(fin);
  readln(fin, n, d);

```

```

for i := 1 to n do begin
//
  readln(fin, b.x, b.y);
  if i = 1 then a[0] := b;
//
//
//           (1 1)
for kx := 0 to d-1 do for ky := 0 to d-1 do begin
// . .           -           ,   x -           , y -
  c.x := b.x + kx; c.y := b.y - ky;
  j := 0;
  while (j < count)and(a[j] <> c) do inc(j);
  if j = count then begin
    inc(count); SetLength(a, count);
    a[count-1] := c;
  end;
end;
end;
CloseFile(fin);
//
AssignFile(fout, 'output.txt'); rewrite(fout);
write(fout, count);
CloseFile(fout);
END.

```