

```

program upruz; //пузырьковый алгоритм по возрастанию
type
mas=array[1..1000] of longint;
cmp=function (a,b:longint):boolean;
var
m:mas;
inp,out:text;
n,i,j,k:longint;

function cmp1(a,b:longint):boolean; //компаратор для убывания
begin
if (a<b) then cmp1:=true
else cmp1:=false;
end;

function cmp2(a,b:longint):boolean; //компаратор для возрастания
begin
if (a>b) then cmp2:=true
else cmp2:=false;
end;

procedure sort(var m:mas; cmpprt:cmp; n:longint); //процедура сортировки
var i,j,k:longint;
begin
for i:= 1 to n-1 do begin
for j:= 1 to n-1 do begin
if cmpprt(m[j],m[j+1]) then begin
k:=m[j];
m[j]:=m[j+1];
m[j+1]:=k; end;
end;
end;
end;

begin
assign(inp, 'input.txt');
reset(inp);
readln(inp,n);
for i:=1 to n do
read(inp,m[i]);
close(inp);
sort(m, @cmp2, n);
assign(out, 'output.txt');
rewrite(out);
for i:=1 to n do
write(out,m[i], ' ');
close(out);
end.

```

```

program downpuz; //пузырьковый алгоритм по убыванию
type
mas=array[1..1000] of longint;
cmp=function (a,b:longint):boolean;
var
m:mas;
inp,out:text;
n,i,j,k:longint;

function cmp1(a,b:longint):boolean; //компоратор для убывания
begin
if (a<b) then cmp1:=true
else cmp1:=false;
end;

function cmp2(a,b:longint):boolean; //компоратор для возрастания
begin
if (a>b) then cmp2:=true
else cmp2:=false;
end;

procedure sort(var m:mas; cmp1:cmp; n:longint); //процедура сортировки
var i,j,k:longint;
begin
for i:= 1 to n-1 do begin
for j:= 1 to n-1 do begin
if cmp1(m[j],m[j+1]) then begin
k:=m[j];
m[j]:=m[j+1];
m[j+1]:=k; end;
end;
end;
end;

begin
assign(inp, 'input.txt');
reset(inp);
readln(inp,n);
for i:=1 to n do
read(inp,m[i]);
close(inp);
sort(m, @cmp1, n);
assign(out, 'output.txt');
rewrite(out);
for i:=1 to n do
write(out,m[i], ' ');
close(out);
end.

```

```

program upbistr; //быстрая сортировка по возрастанию
type
mas=array of longint;
cmp=function (a,b:longint):boolean;
var
m:mas;
inp,out:text;
i,n:longint;
    function cmp1(a,b:longint):boolean; //компаратор для убывания
begin
    if (a<b) then cmp1:=true
    else cmp1:=false;
end;
function cmp2(a,b:longint):boolean; //компаратор для возрастания
begin
    if (a>b) then cmp2:=true
    else cmp2:=false;
end;
procedure sort(l,r:longint; cmpprt:cmp; m:mas); //быстрая сортировка
var
    i,j,sr,k,k1:longint;
begin
    i:=l; j:=r; sr:=round ((l+r)/2); k1:=m[sr];
    repeat
        while m[i]<k1 do inc(i);
        while cmp2(m[j],k1)=true do dec(j);
        if cmp2(i,j)=false then
            begin
                k:=m[i]; m[i]:=m[j]; m[j]:=k; inc(i); dec(j);
            end;
    until cmp2(i,j)=true;
    if cmp2(l,j)=false then sort(l,j,@cmp2,m);
    if cmp2(i,r)=false then sort(i,r,@cmp2,m);
end;
begin
assign(inp, 'input.txt');
reset(inp);
readln(inp,n);
setlength(m,n+1);
for i:=1 to n do
read(inp,m[i]);
close(inp);
    sort(1,n,@cmp2,m);
assign(out, 'output.txt');
rewrite(out);
for i:=1 to n do
write(out,m[i], ' ');
close(out);
end.

```

```

program dbistr; //быстрая сортировка по убыванию
type
mas=array of longint;
cmp=function (a,b:longint):boolean;
var
m:mas;
inp,out:text;
i,n:longint;
    function cmp1(a,b:longint):boolean; //компаратор для убывания
    begin
        if (a<b) then cmp1:=true
        else cmp1:=false;
    end;
    function cmp2(a,b:longint):boolean; //компаратор для возрастания
    begin
        if (a>b) then cmp2:=true
        else cmp2:=false;
    end;
procedure sort(l,r:longint; cmpprt:cmp; m:mas); //быстрая сортировка
var
    i,j,sr,k,k1:longint;
begin
    i:=l; j:=r; sr:=round ((l+r)/2); k1:=m[sr];
    repeat
        while m[i]>k1 do inc(i);
        while cmp1(m[j],k1)=true do dec(j);
        if (cmp1(i,j)=true) or (i=j) then
            begin
                k:=m[i]; m[i]:=m[j]; m[j]:=k; inc(i); dec(j);
            end;
    until cmp1(i,j)=false ;
    if cmp1(l,j)=true then sort(l,j,@cmp1,m);
    if cmp1(i,r)=true then sort(i,r,@cmp1,m);
end;
begin
assign(inp, 'input.txt');
reset(inp);
readln(inp,n);
setlength(m,n+1);
for i:=1 to n do
read(inp,m[i]);
close(inp);
    sort(1,n,@cmp1,m);
assign(out, 'output.txt');
rewrite(out);
for i:=1 to n do
write(out,m[i], ' ');
close(out);
end.

```