



«Նոր ժամանակի կրթություն» ՀԿ

ՀԵՐԹԱԿԱՆ ԱՏԵՍՏԱՎՈՐՄԱՆ ԵՆԹԱԿԱ  
ՈՒՍՈՒՑԻՉՆԵՐԻ ՎԵՐԱՊԱՏՐԱՍՏՄԱՆ  
ԴԱՍԸՆԹԱՑ

ԱՎԱՐՏԱԿԱՆ ՀԵՏԱԶՈՏԱԿԱՆ  
ԱՇՈՒՏԱՆՔ

Հետազոտության թեման՝ «Pascal ABC NET» ծրագրի  
արդյունավետությունը ինֆորմատիկայի գործնական  
աշխատանքների դասավանդման ընթացքում՝ ծրագրավորման  
Պասկալ լեզուն դասավանդելիս

Առարկան՝ Ինֆորմատիկա

Հետազոտող ուսուցիչ՝ Աննա Ասատրյան

Ուսումնական հաստատություն՝ «Ակունքի միջնակարգ դպրոց»

Եռևան 2022

## Բովանդակություն

- Տիտղոսաթերթ.....
- Բովանդակություն ..... էջ 1
- Ներածություն ..... էջ 2-3
- Պարագրաֆ 1-ին ..... էջ 3-7
- Պարագրաֆ 2-րդ ..... էջ 8-12
- Պարագրաֆ 3-րդ ..... էջ 12-16
- Եզրակացություն ,առաջարկություններ..... էջ 16-17
- Օգտագործված գրականության ցանկ ..... էջ 18

## Ներածություն

Արդի մասնագիտություն ընտրելը ,սովորողներին մոտիվացնելը կրթության կարևորագույն նպատակն է: Քանզի մասնագիտությունը ծառայեցնել ի նպաստ հայրենիքի բարգավաճմանը և զարգացմանը յուրաքանչյուրիս պարտքն են համարում` չէ որ դա մեր բարեկրթության հիմնական գրավականն է. մեր երկիրն անառիկ պահելու ,բարեկեցիկ կյանք ապահովելու ճշմարիտ ուղիներից մեկը: Մեր երկրի կառավարությունը սկսել է կրթական գործընթաց `2023թ.-ի կրթական չափորոշչի համաձայն, նպաստելու հետևյալ ռազմավարական նշանակություն ունեցող գործընթացին:Դա ինֆորմատիկայի դասաժամերի ավելացումն է հանրակրթական դպրոցներում:

Իսկ ի՞նչ մասնագիտություն ընտրել, միանշանակ` ծրագրավորում, որի հիմքը դրվում է հանրակրթական դպրոցում : Գաղ տնիք չէ այն փաստը, երբ աշխարհում հայտնի բազմաթիվ ծրագրավորողներ պնդում են այն մասին , որ ծրագրավորման հիմքը Պասկալ լեզուն է, բացի այդ չեն կիսում որոշ ուսուցիչների կարծիքը, թե այն ուսուցանելը և սովորելը բարդ է: Բոլորովին էլ բարդ չէ, եթե ունես որոշ տարրական հմտություններ : Դրանցից մեկը <<**Pascal ABC NET**>> ծրագրի միջավայրում սովորողներին Պասկալ լեզու ուսուցանելն է: Այն ունի պարզ ինտերֆեյս, որը գրավիչ է դարձնում աշխատանքը: Կարելի է համատեղել նաև բազմաթիվ այլ բջջային հավելվածներ:

Սովորողների ոգևորությունն անսահման է , երբ մի քանի դասաժամից հետո ցուցաբերում են ակտիվություն` ցանկանալով միայնակ որևէ պարզագույն ծրագիր գրել: Դա առիթ է հանդիսանում էլ ավելի պատասխանատու կերպով մոտենալ դասին նախապատրաստվելուն և այն առավել արդյունավետ կազմակերպելուն:

Սովորելով Պասկալ լեզու, նրանք ապագայում կկարողանան հեշտությամբ ընկալել մնացած (ֆորմալ )լեզուները:

Սովորողները արագ են ադապտացվում, իմ կարծիքով բոլորովին էլ երազանք չէ ապագայում բոլորի

**Python**-ում կող գրելը :

Գործնական և լաբորատոր աշխատանքների կատարման ժամանակ միայն, ինֆորմատիկայի դասաժամի ընթացքում, աշակերտների գիտելիքները վերածվում են կարողությունների:

Այդ կարողությունները դառնում են հմտություններ և կիրառելի են դառնում կյանքում:

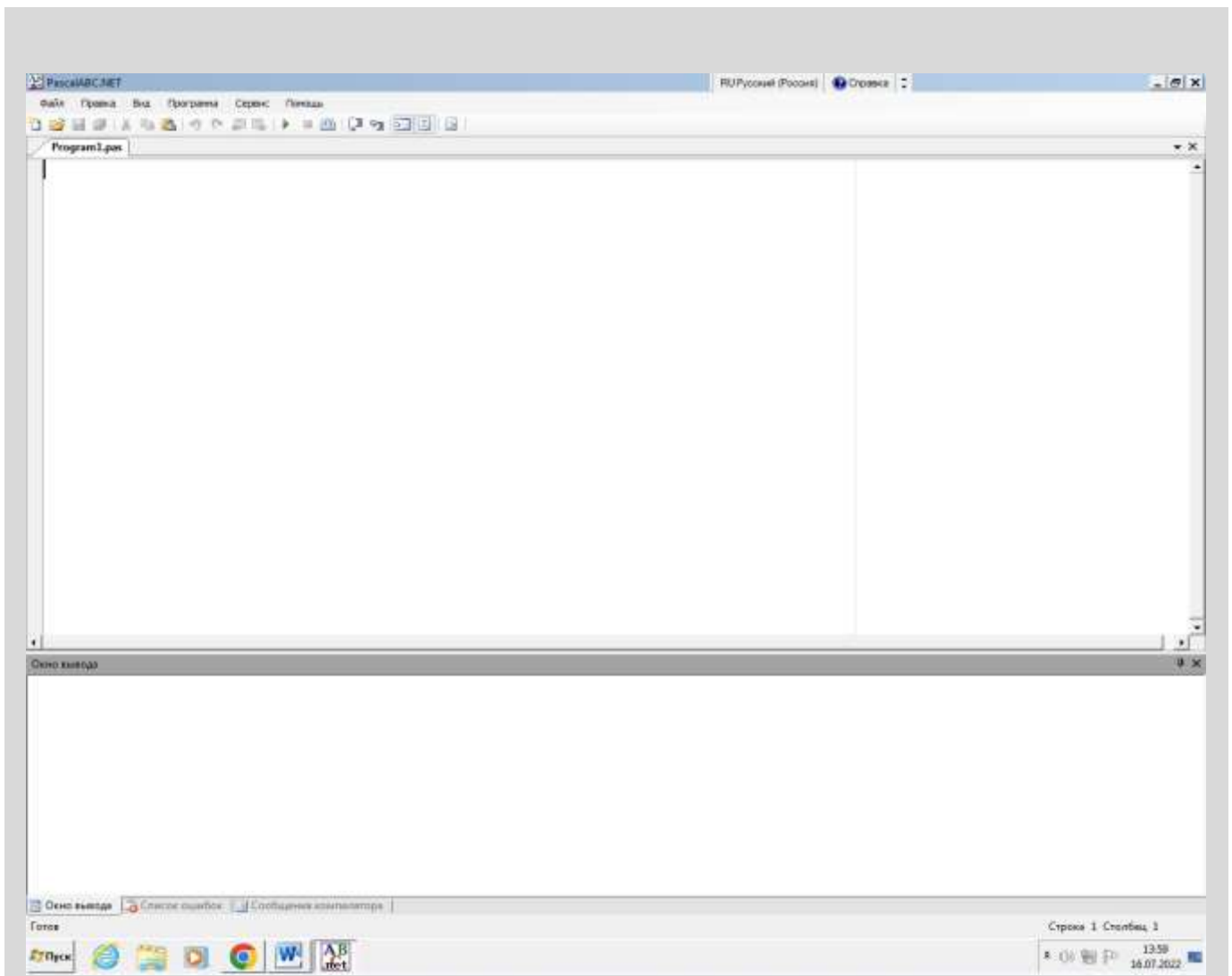
Այսպես` միայն կարևորելով յուրաքանչյուր թեմա, հետագայում կունենանք գերազանց արդյունք:

Միանշանակ իրագործելի է և նպատակային ըստ իս՝ անչափ կարևոր գոծոնի կիրառման դեպքում դրական արդյունք ունենալը: Դրան կարելի է հասնել հստակ կրթական ծրագրի օգնությամբ, հետևելով չափորոշի պահանջներին և չխախտել սահմանված գծային ալգորիթմի կարգը:

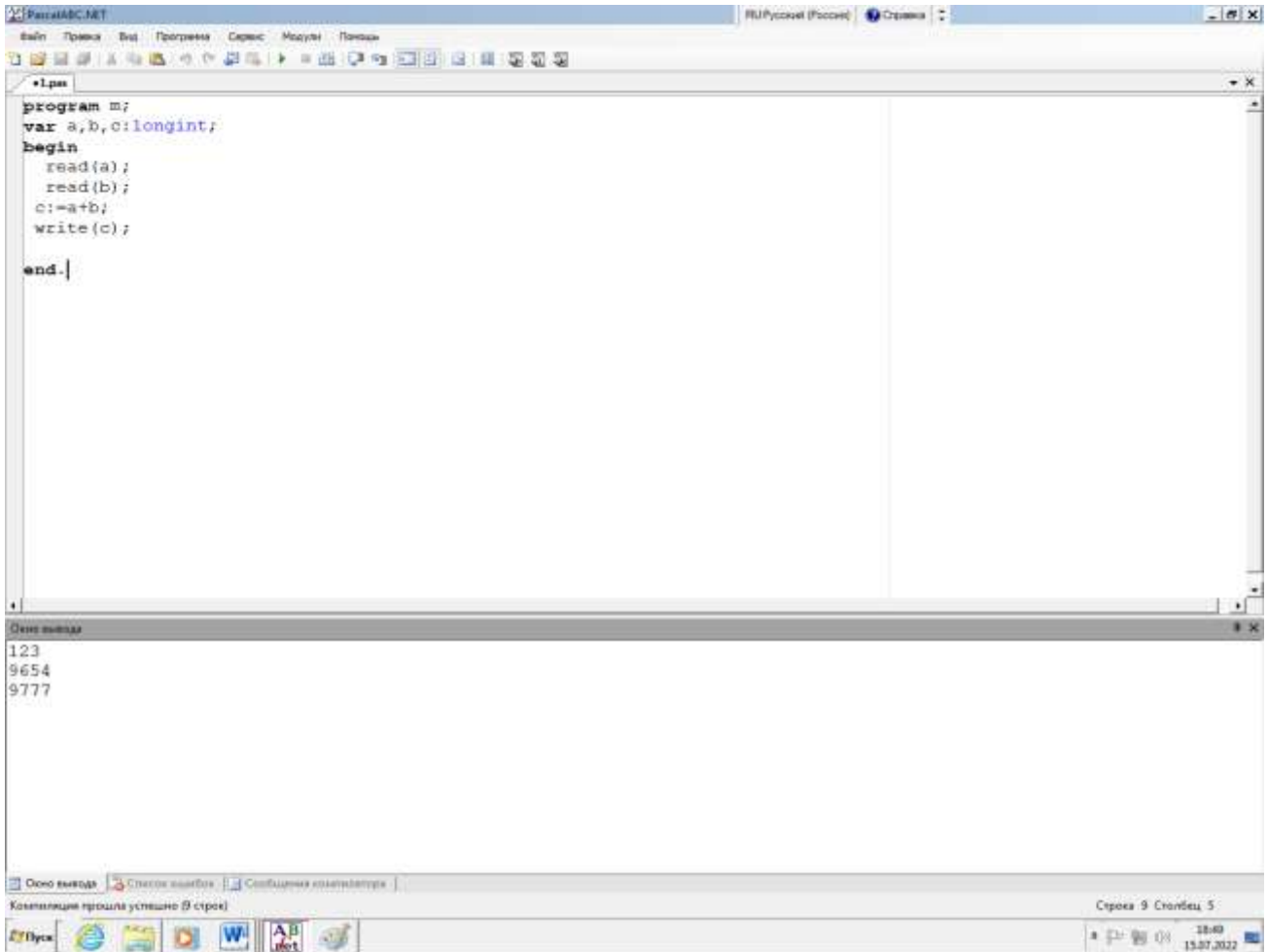
Ժամանակակից, կիրթ պահանջված մասնագիտություն ունեցող քաղաքացի դաստիարակելը մեր՝ ուսուցիչներիս, հիմնական խնդիրն է և ուրեմն ուղղենք մեր ամբողջ ջանքերն ու եռանդը այս կարևորագույն աշխատանքին:

### Պարագրաֆ 1-ին

1. Պասկալ լեզվի այբուբենը ,տարբերը թեմայից հետո անմիջապես աշակերտների ներկայությամբ սկսել ծրագրի վերբեռնման գործընթացը: Դա և՛ գիտելիք, և՛ ծանոթություն կտա սովորողներին : <<Pascal ABC NET>> խմբագրիչի միջավայրում սովորողները հեշտությամբ կկողմնորոշվեն, քանի որ ծրագիրը ունի պարզ կառուցվածք ,գրավիչ ինտերֆեյս: Համոզվելու համար դիտենք ծրագրի ներքին տիրույթի արտաքին տեսքը:



Ինչպես տեսնում ենք աշխատանքային միջավայրը կազմված է երկու պատուհանից՝ կող գրելու և տվյալներ ներմուծելու : Տվյալներ ներմուծելու պատուհանի մասում ծրագիրը գործարկելու ժամանակ առաջանում է տվյալների արտածման դաշտ,որտեղ ցանկության դեպքում երևում են և՛ ներմուծված, և՛ արտածված տվյալները: Դիտարկենք ամենապարզ մի ծրագիր, որտեղ ներկայացված է երեք փոփոխականների հայտարարմամբ որևէ մաթեմատիկական գործողություն ,օրինակ՝ գումարում:



Այս օրինակը գրավիչ է իր պարզությամբ ,այդ պատճառով ցանկացած սովորող բացի 123 և 9654 թվերը գումարելուց ցանկություն է հայտնում փորձել բացասական թվերի համար ,կամ օրինակ ՝ կոտորակների:Առաջ է գալիս միջակայքեր սովորելու անհրաժեշտությունը: Այդպես շարունակ ուսուցիչը պետք է պատրաստի հաջորդ դասի ընթացքում բացատրելու ՝ այնպիսի ծրագիր ,որը կառնչվի անմիջապես թեմային ու կհեշտացնի աշակերտի կողմից ըմբռնումը: Որքան հետաքրքիր լինի կազմված ծրագրի առաջադրանքը ,այնքան դասն արդյունավետ և օգտակար կլինի:

Երբ գծային ալգորիթմական ծրագրերից անցում ենք կատարում ցիկլայինի և ճյուղավորված ալգորիթմով խնդիրներին,ապա կարելի է նույն խնդիրը ներկայացնել մի քանի ձևովով, այսպես օրինակ՝

## Խնդրի պահանջը՝

Արտածել ներմուծված քանակով սինվոլներ(\*)

Խնդիր 1.

Var

I,n:integer;

Begin

Write('Ներմուծվող նշանների քանակը:');  
Readln(n);

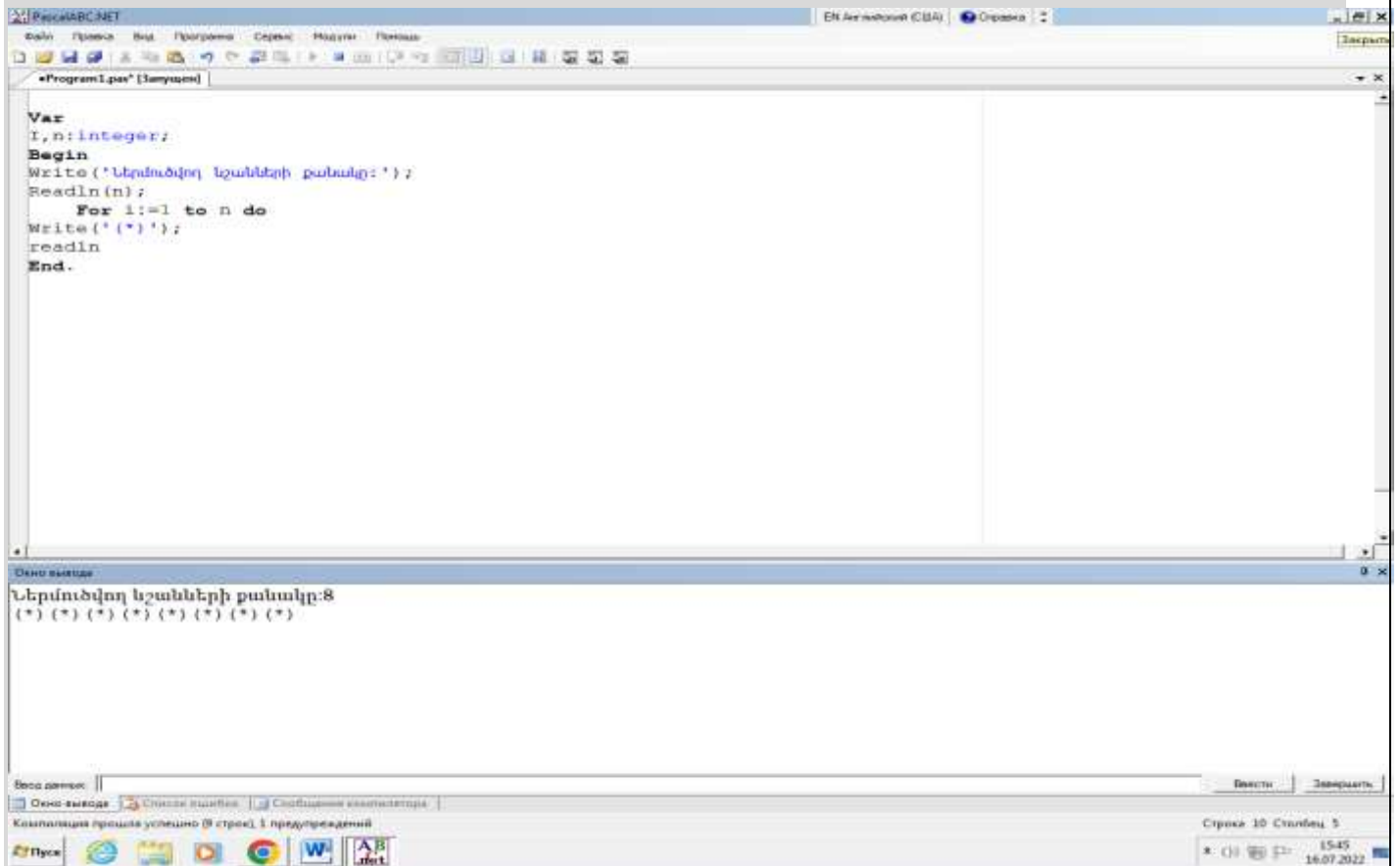
For i:=1 to n do

Write('\*');

readln

End.

Ծրագիրը թողարկելիս կունենանք հետևյալ տեսքը:



```
Var
I,n:integer;
Begin
Write('Ներմուծվող նշանների քանակը:');
Readln(n);
  For i:=1 to n do
Write('* ');
readln
End.
```

Ներմուծվող նշանների քանակը:8  
(\* ) (\* ) (\* ) (\* ) (\* ) (\* ) (\* )

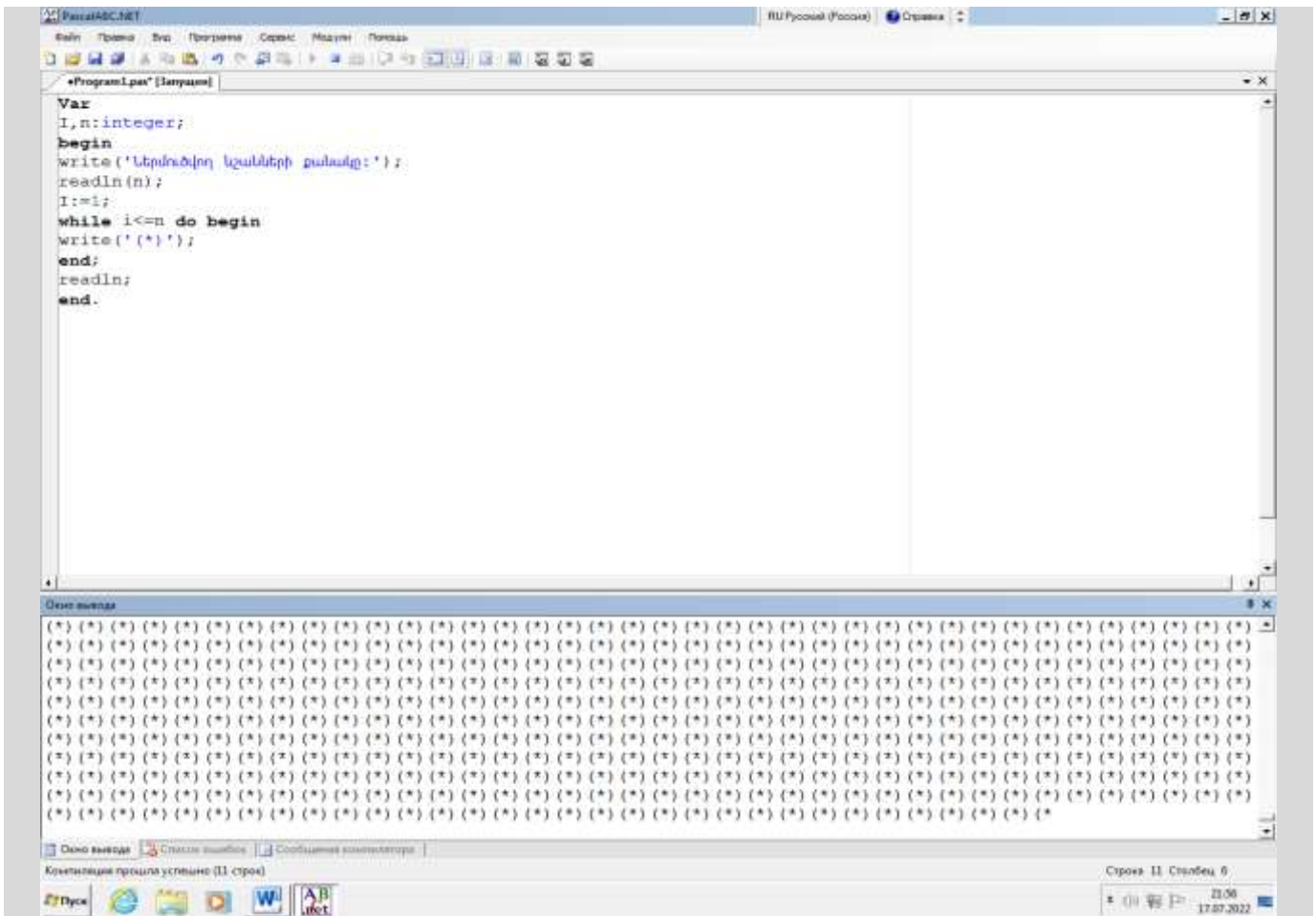
Նույն արդյունքը կստանանք արդյոք, եթե For-ի փոխարեն կիրառենք While օպերատորը:

Առաջին հայացքից այո,բայց դա միայն առաջին հայացքից:

Այսպիսով ծրագիրը գործարկելուց, ցանկացած սիմվոլ ներմուծելիս կստանանք անվերջանալի աստղանիշների տարափ ,որը կարելի է ընդհատել միայն Esc կոճակով:

## Խնդիր 2

```
Var
I,n:integer;
begin
write('Ներմուծվող նշանների քանակը:');
readln(n);
I:=1;
while i<=n do begin
write('* ');
end,
readln
end.
```

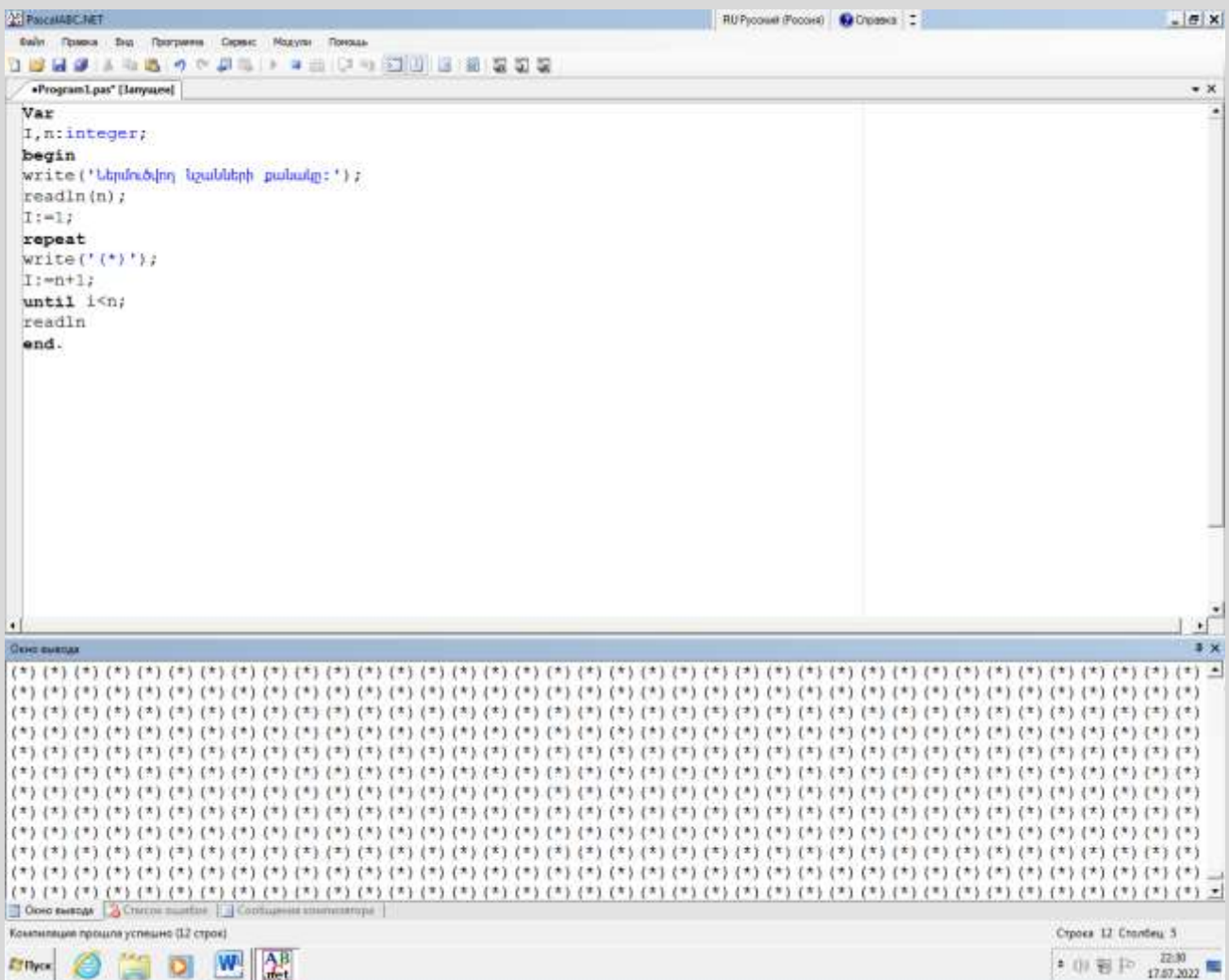


Նույն արդյունքը չենք ստանա, եթե For-ի փոխարեն կիրառենք repeat -until օպերատորը:

Բայց կստացվի Խնդիր 2 –ի նույն արդյունքը: Այսպիսով ծրագիրը գործարկելուց հետո կստացվի անվերջ աստղանիշերը ութ, կամ որևէ այլ թիվ մուտք անելու ժամանակ:

### Խնդիր 3

```
Var
I,n:integer;
begin
write('Ներմուծվող նշանների քանակը:');
readln(n);
I:=1;
repeat
write('* ');
I:=n+1;
until i>n;
readln
end.
```



Հատկանշական է, որ այս բոլոր գիտելիքները աշակերտը կստանա ժամանակին և խելամիտ ծավալներով՝ համապատասխանեցնելով չափորոշային պահանջներին:



## Պարագրաֆ 2-րդ

Հաջորդ թեման, որը նախատեսել էմ ներկայացնել, դա որոշ նոր օպերատորների կիրառությունն է՝ ինդիքների լուծման օրինակներով:

Օրինակ `DIV և MOD օպերատորների կիրառությունը ինդիքների լուծման ժամանակ :Ինչպես հայտնի է դրանք կատարում են բաժանման գործողություն մնացորդի անտեսմամբ,իսկ մյուս դեպքում ամբողջ մասի անտեսմամբ :Ինդիքները բազմաթիվ են ,որոնք կարելի է կատարել վերոնշյալ օպերատորների կիրառությամբ:

Օրինակ.

Ինդիքի պահանջը՝Ներմուծած թվի հակադարձ թվի դուրսբերում:

Ինդիք 4

```
var n,m:longit;
    a:byte;
    begin
    readln(n);
    m:=0;
    while n>0 do begin
        a:=n mod 10;
        n:=n div 10;
        m:=-m*10+a;
    end;
    writeln(m);
    readln;
    end.
```

Այս ծրագրի գործարկման ժամանակ 526319 թվի ներմուծման ժամանակ կստանանք 913625 թիվը:

Հաջորդ օրինակները ավելի հետաքրքիր և գրավիչ է դարձնում դասաժամը, հաստատելով մշտական միջառարկայական կապը մաթեմատիկայի և ինֆորմատիկայի միջև:

Թվերի հաջորդականություն թեմայի խնդիրներ՝ ֆակտորիալ և ֆիբոնաչի:

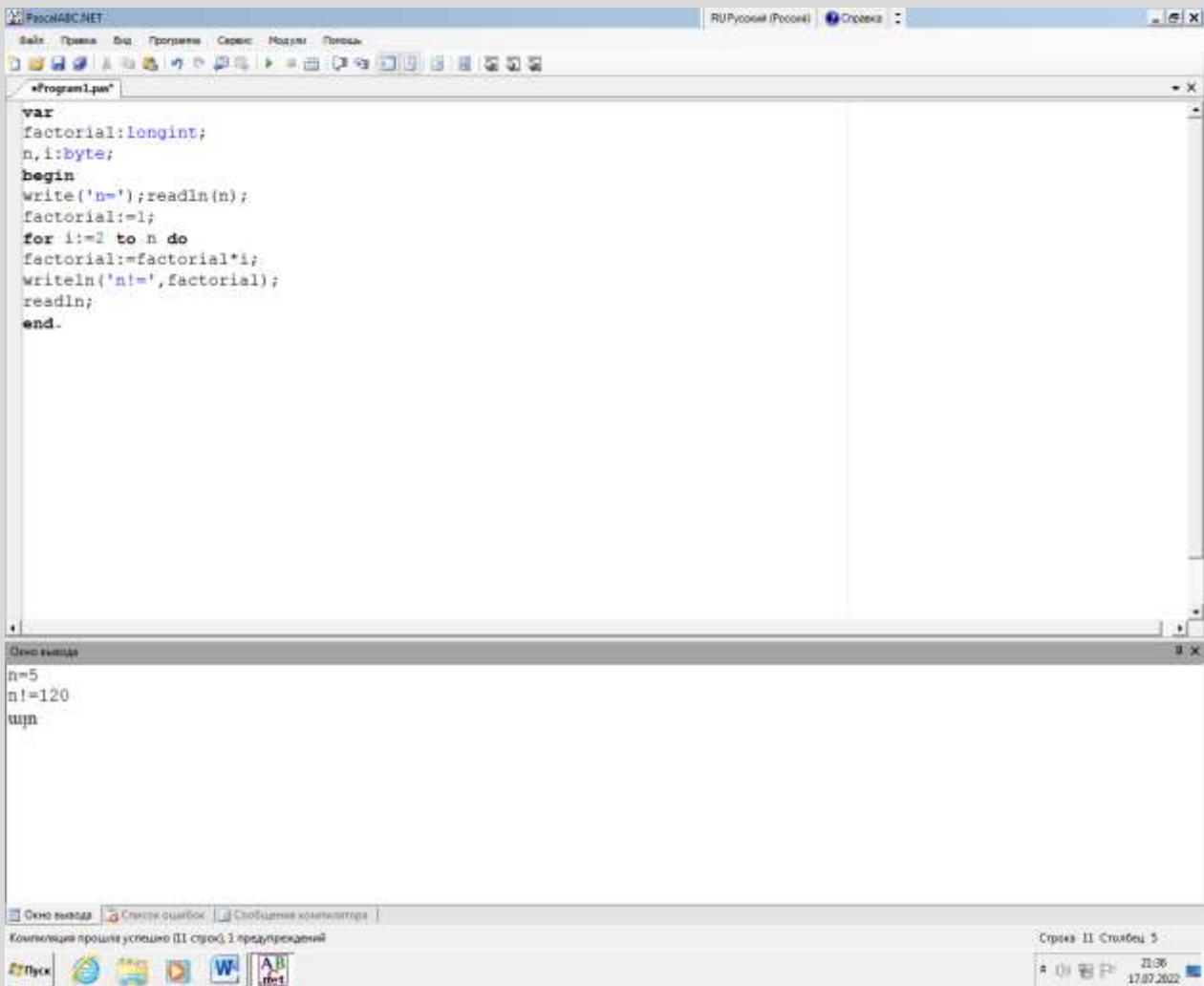
Օրինակ.

Խնդրի պահանջը՝ թվի ֆակտորիալի գտնելը:

Խնդիր 5

```
var
factorial:longint;
n,i:byte;
begin
write('n=');readln(n);
factorial:=1;
for i:=2 to n do
factorial:=factorial*i;
writeln('n!=',factorial);
readln;
end.
```

Ինչպես գիտենք, ֆակտորիալի բանաձևն է՝  $n! = n \cdot (n-1) \cdot (n-2) \cdot \dots \cdot 1$ , հետևաբար ծրագիրը գործարկելիս կստացվի հետևյալ տեսքը:



```
var
factorial:longint;
n,i:byte;
begin
write('n=');readln(n);
factorial:=1;
for i:=2 to n do
factorial:=factorial*i;
writeln('n!=',factorial);
readln;
end.
```

Արդյունք  
n=5  
n!=120  
աղբ

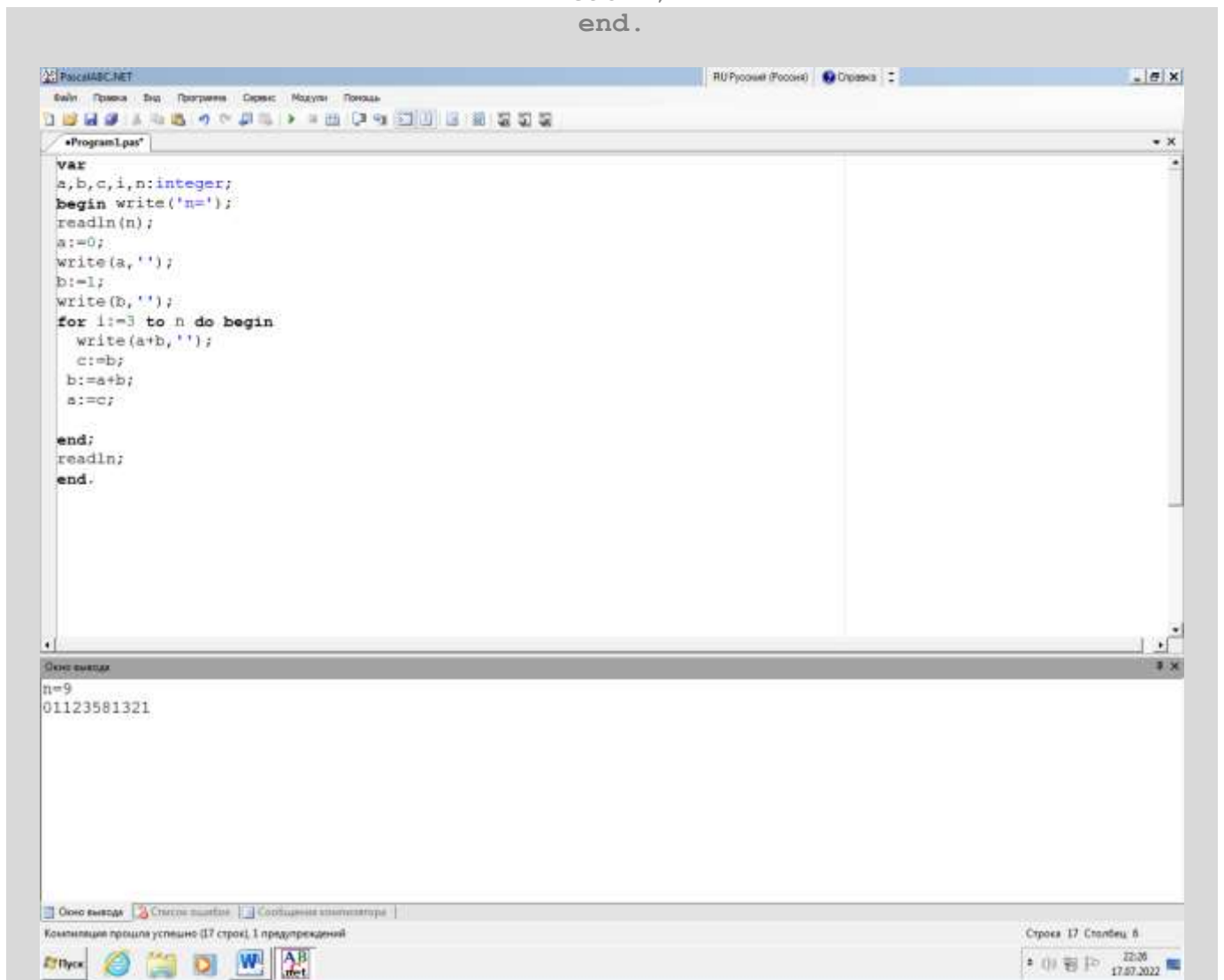
## Օրինակ.

Խնդրի պահանջը՝ Ֆիբենաչի թվեր:

### Խնդիր 6

```
var
a,b,c,i,n:integer;
begin write('n=');
      readln(n);
      a:=0;
      write(a,' ');
      b:=1;
      write(b,' ');
for i:=3 to n do begin
write(a+b,' ');
  c:=b;
  b:=a+b;
  a:=c;

end;
readln;
end.
```



The screenshot shows the PascalABC.NET IDE with the following code in the editor:

```
var
a,b,c,i,n:integer;
begin write('n=');
      readln(n);
      a:=0;
      write(a,' ');
      b:=1;
      write(b,' ');
for i:=3 to n do begin
write(a+b,' ');
  c:=b;
  b:=a+b;
  a:=c;

end;
readln;
end.
```

The output window displays the result of the program execution:

```
n=9
01123581321
```

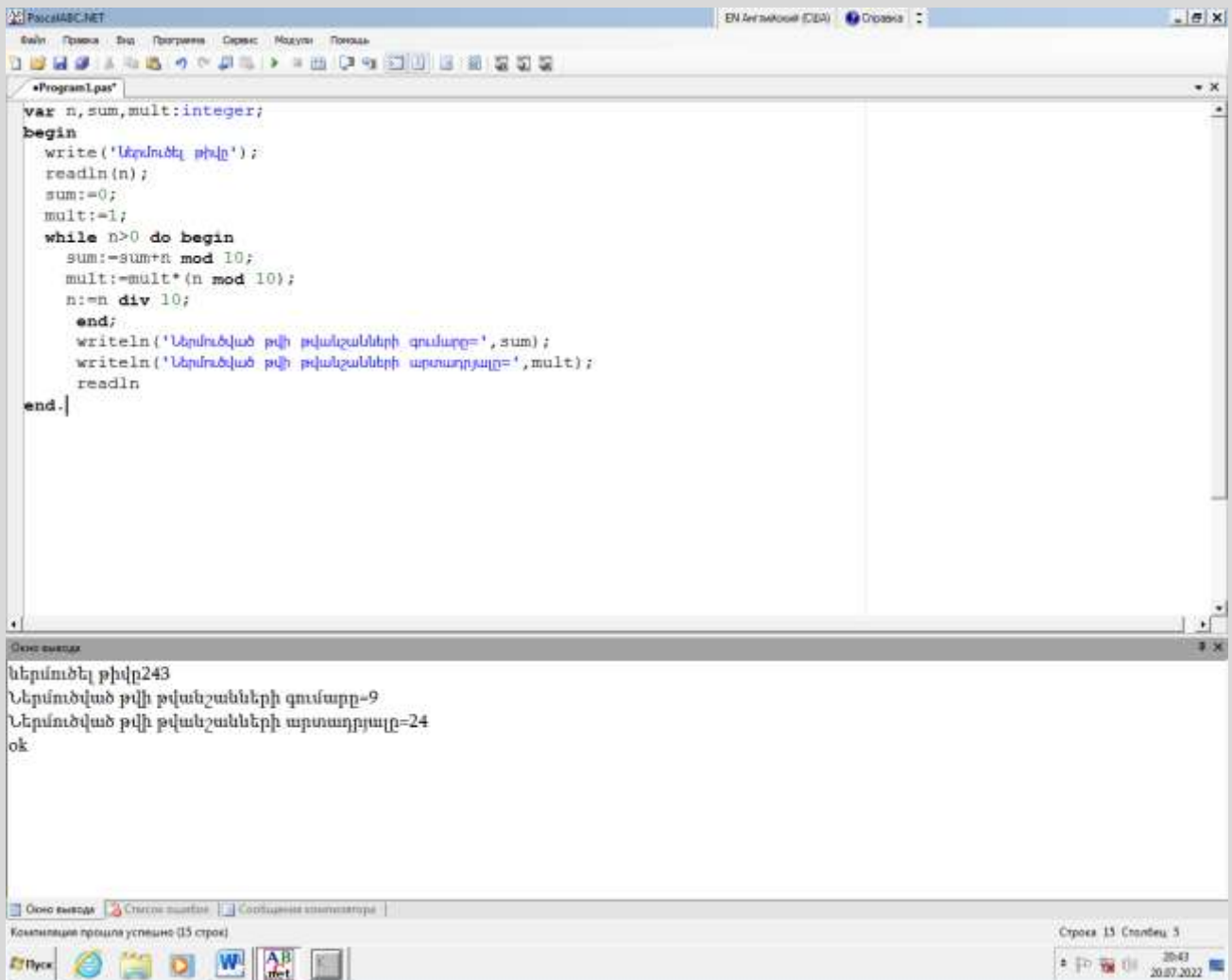
The status bar at the bottom indicates that the compilation was successful (17 lines, 1 warning) and the current cursor position is at line 17, column 6. The system tray shows the date and time as 17.07.2022, 22:06.

Վերոնշյալ խնդրում ներկայացված է ֆիբոնաչի թվային շարք  $/9/$ -ը թվանշանի ներմուծման դեպքում օրինակ՝ ծրագիրը գործարկելիս, դուրս կբերվի հետևյալ սովյալները՝  $/9+2/$  թվանշան, որտեղ ամեն հաջորդ թիվը հավասար է երկու նախորդ թվերի արտադրյալին:

Օրինակ.

Խնդրի պահանջ՝ գտնել տրված թվի թվանշանների գումարը և արտադրյալը:

Խնդիր 7



Ինչպես տեսանք այս ծրագիրը ներմուծած 243 թվի թվանշանները առանձնացրեց DIV և MOD օպերատորներով, իսկ հետո կատարվեց բազմապատկում և գումարում գործողություններ:

DIV և MOD կատարում են բաժանում մի դեպքում մնացորդի անտեսմամբ, իսկ մյուս դեպքում ամբողջ մասի անտեսմամբ: Դա հնարավորություն է տալիս թվի թվանշանները առանձնացնել և կիրառել ծրագրի ընթացքում: Իրականում, կարելի է բազմզթիվ այլ, առօրյա խնդիրներ կազմել՝ տարբեր առարկաներին առընչվող և խթանել սովորողների հետաքրքրասիրությունը:

:

Դա ամեն դեպքում կօգնի զարգացնել խնդիրների իմացությունը:

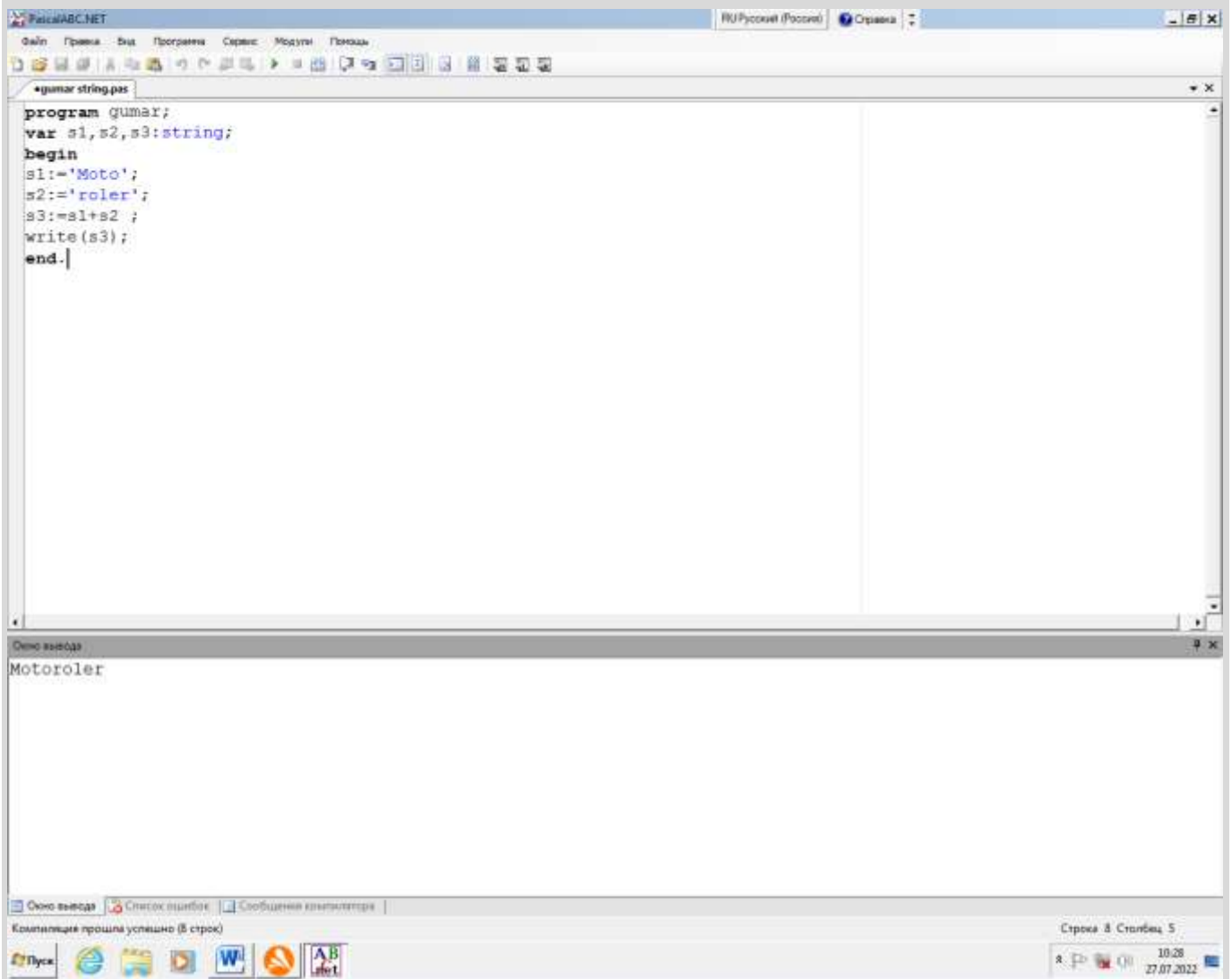
### Պարագրաֆ 3-րդ

Այս պարագրաֆը զանգվածների մասին է,կամ ինչը նույն է `մասսիվներ մասին:

Նաև, ներառված է սիմվոլային տվյալներ: Սիմվոլային տվյալների ամենատարածված միջակայքերը Char և String միջակայքերն են: Մաքսիմալ տողի երկարությունը մինչև 255 սիմվոլ է: Այդպես է ,որովհետևայդ Char տիպը պարունակում է հենց այդքան սիմվոլ: Պասկալում ամեն մի տողը ունի հենց այդքան ինֆորմացիա: Իսկ String տիպի միջակայքը տարբերվում է նրանով , որ կարող է պարունակել մեկից ավելի բառեր, այսինքն ամբողջական տողեր: Առաջինը բերենք օրինակներ սիմվոլային տվյալներ պարունակող խնդիրներից:

### Խնդիր 8

Խնդրի պահանջը` միացնել նախադասության մասերը :



Ինչպես տեսանք, այս օրինակում միացվեցին բառի երկու մասերը ,բայց կարելի է կիրառել նույնը նախադասության համար:

### Խնդիր 9

Խնդրի պայմանն՝ առանձնացնել ներմուծված տողից թվանշաններ ,կազմել դրանցով զույգ և կենտ թվեր:

Ինչպես տեսանելի է այս խնդրում կիրառվել է նաև String տիպի միջակայք:

```
S1, S2, S3: string;
l, i: byte;
begin
  readln(S1);
  S2:= '';
  S3:= '';
  l:= length(S1);
  for i:=1 to l do
    if odd(i) then
      S3:=S3+S1[i]
    else
      S2:=S2+S1[i];
  Writeln(S2);
  Writeln(S3);
end.
```

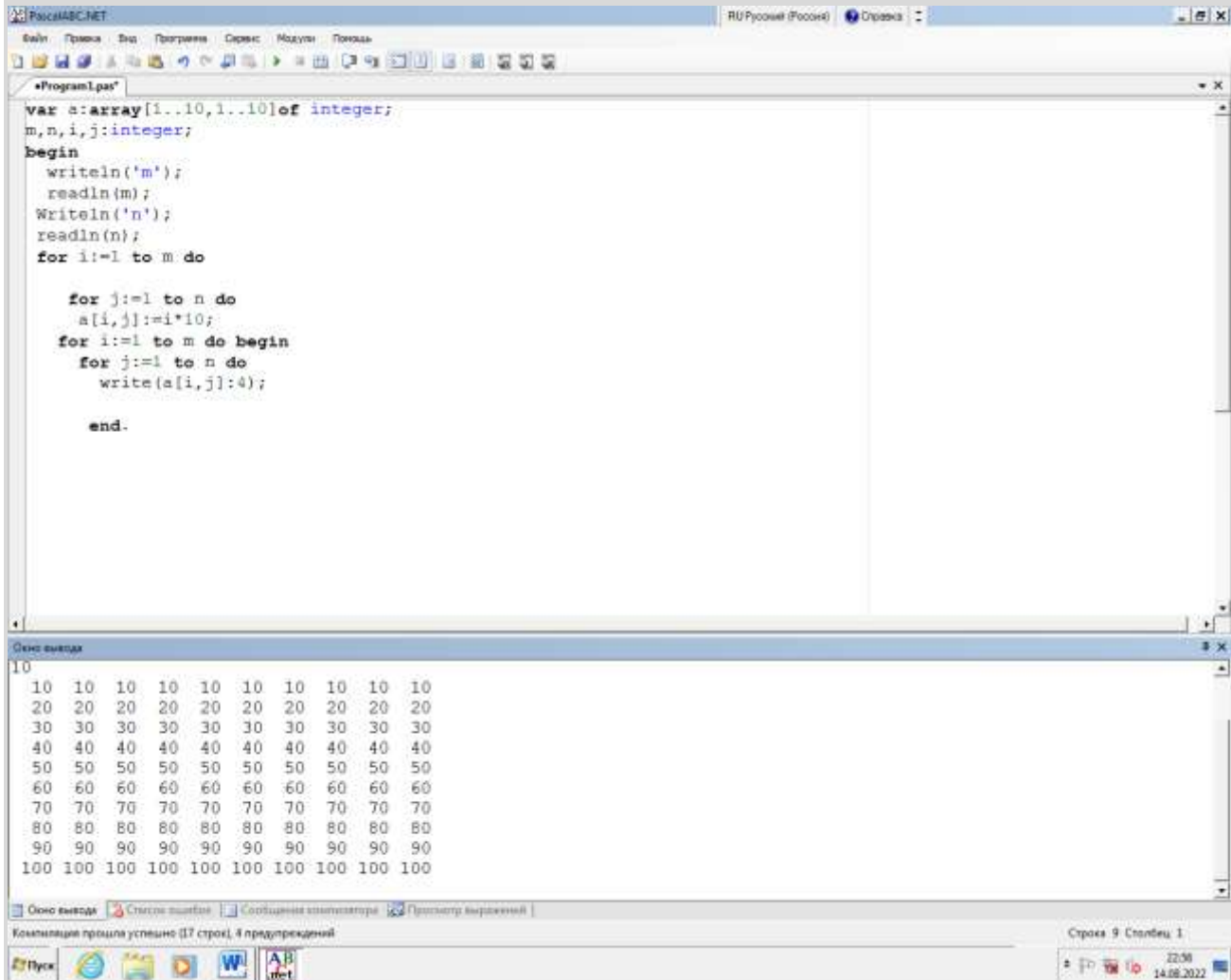
3657894052481  
679028  
3584541

Այսպիսով անցում կատարենք զանգվածների: Առաջին խնդիրը ,որը որպես այլընտրանք, կարելի է ներկայացնել՝ դա բազմապատկման աղյուսակի մասին ծրագիրն է:

## Խնդիր 10

Խնդրի պայման՝ Տրված են ամբողջ դրական թվեր  $M$  և  $N$  :

Ստանալ ամբողջական մատրիցա  $M * N$  չափի ,որի բոլոր էլեմենտները առաջին տողում փոխում են իրենց նշանակությունը ,10 –առաջին/առաջինից-1,...,  $M$ /:

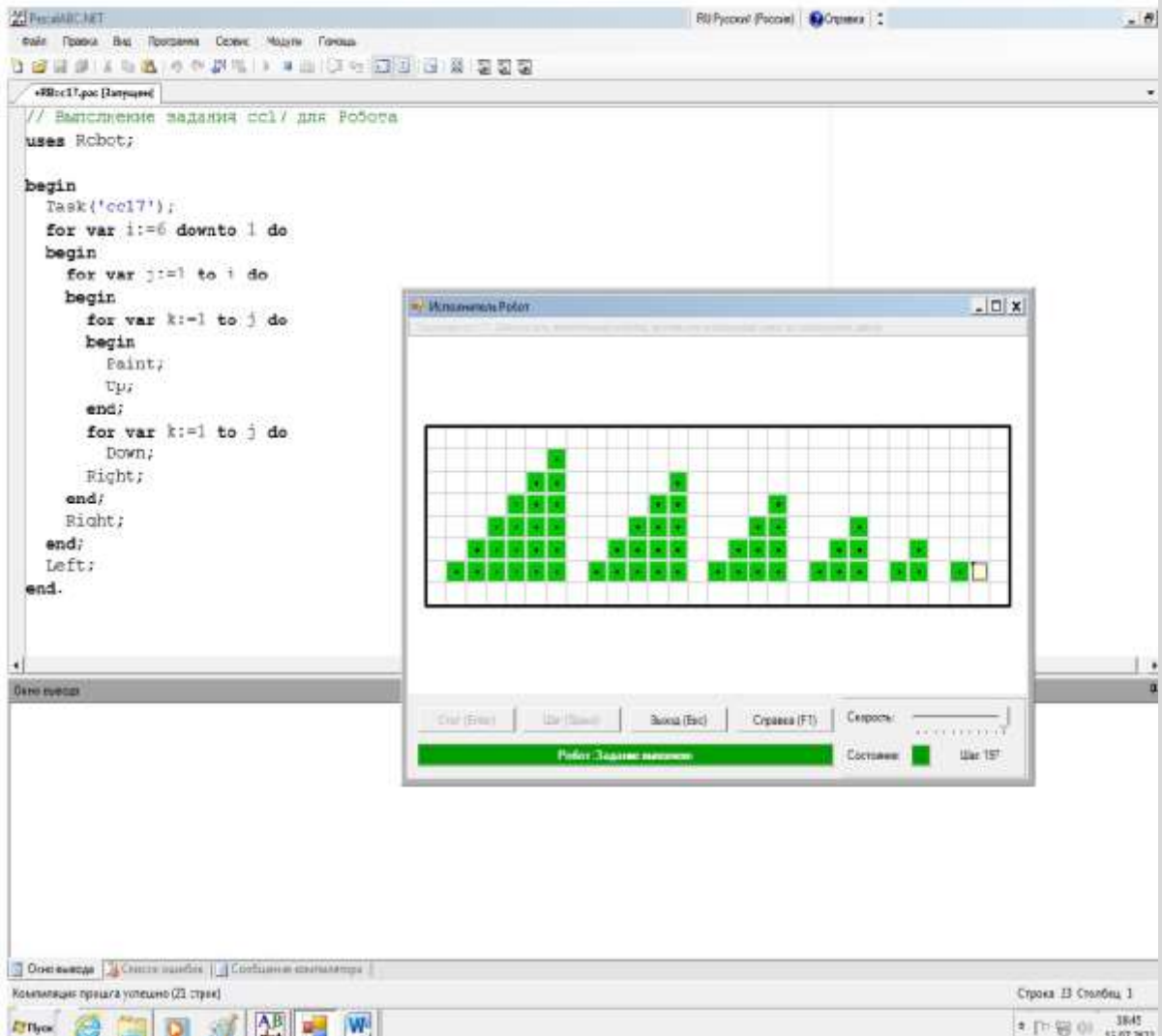


```
var a:array[1..10,1..10]of integer;
m,n,i,j:integer;
begin
  writeln('m');
  readln(m);
  writeln('n');
  readln(n);
  for i:=1 to m do
    for j:=1 to n do
      a[i,j]:=i*10;
    for i:=1 to m do begin
      for j:=1 to n do
        write(a[i,j]:4);
      end.
end.
```

10  
10 10 10 10 10 10 10 10 10 10  
20 20 20 20 20 20 20 20 20 20  
30 30 30 30 30 30 30 30 30 30  
40 40 40 40 40 40 40 40 40 40  
50 50 50 50 50 50 50 50 50 50  
60 60 60 60 60 60 60 60 60 60  
70 70 70 70 70 70 70 70 70 70  
80 80 80 80 80 80 80 80 80 80  
90 90 90 90 90 90 90 90 90 90  
100 100 100 100 100 100 100 100 100 100

Նմանօրինակ խնդիրները հետաքրքիր են և սովորողները մեծ ոգևորությամբ են կատարում:

Հաջորդ տեսակի խնդիրները նույնպես կարելի է առաջնահերթ ներկայացնել կապիլյատորի ինտերֆեյսի օրինակներից՝ շարժողական խաղի օրինակի տեսքով: Իսկ հետո սովորողը մեծ հաճույքով կփորձի սովորել այն ամենը, ինչ կբացատրի և կցուցադրի ուսուցիչը: Ահա և օրինակները՝



Այսպիսի օրինակներ կարելի է ավելի շատ բերել և քննարկել բոլոր օգտագործված օպերատորները, ֆունկցիաները ծրագրի ընթացքում օգտագործված ալգորիթմի տեսակը և մնացած բաղադրիչները:



## Եզրակացություն, առաջարկություններ

Այսպիսով կարող ենք նշել <<PascalABCNET>> ծրագրի արդյունավետությունը ,այս ծրագրի բազմաթիվ առավելությունները: Դրանցից են առաջին հերթին <<Pascal++>> ,

<< Pascal++>> և մնացած այլ տարբերակների համեմատ պարզ աշխատանքային միջավայր /ինտերֆեյս/ արագագործություն:

Ծրագիրը բեռնել կարելի է ամենազանազան ծրագրային ապահովման փաթեթների միջավայրում: Այն անվճար և անժամկետ տարբերակ է առաջարկում, որը հարմար է տարբեր սոցիալական հնարավորությունների տեր սովորողների և ուսուցիչների համար:

Սովորողները շատ արագ են հարմարվում ծրագրի միջավայրին , կարևորելով գործնական աշխատանքների դերը ուսումնառության ընթացքում կարող ենք եզրակացնել ,որ այս տեսակ լաբորատոր աշխատանքներ կազմակերպելը կնպաստի արագ և արդյունավետ գիտելիքների ձեռքբերմանը , խթան կհանդիսանա նրանց մասնակցությունն ապահովել զանազան մրցույթներին և դպրոցական օլիմպիադաներին` <<Ինֆորմատիկա>> առարկայից :

Իսկ խթանել սովորողների գիտելիքները պարտադիր պայման եմ համարում ,քանի որ միայն այդպես է առանում սեր և հետաքրքրություն առարկայի նկատմամբ: Չէ որ գիտելիքը այն զենքն է , որը կարող է դիմակայել դարերի անարդարությանը ու արհավիրքներին որոնց հանդիպում է մեր ազգը, իսկ <<Ինֆորմատիկա>> առարկայի խորացումը հատկապես ծրագրավորման բնագավառում համարում եմ ռազմավարական նշանակություն:

Այնպես որ ,որպես առաջարկ ` սովորենք և սովորեցնենք անընդմեջ և անդադար....

## Օգտագործված գրականության ցանկ

- ❖ Հանրակրթական ավագ դպրոցի ընդհանուր և հումանիտար հոսքի համար  
Ինֆորմատիկա 12 Ս.Ս.Ավետիսյան Ս.Վ. Դանիելյան -1
- ❖ В.В. Фаронов<<Основы Турбо-Паскаля>>
- ❖ Е.А.Зуев <<Язык программирования<<Pascal >>
- ❖ Д.Б.Поляков, И.Ю. Круглов <<Програмирование в среде Турбо Паскаль>>