

## **Почитувани идни студенти на ИИ,**

Во продолжение се наоѓаат задачи со РЕШЕНИЈА од приемните испити во последните 10-тина години.

Задачите во врска со MS-DOS може да ги игнорирате ☺

Уживајте!

## Задачи

**06.07.1995**

---

1. Заокружи ја буквата Т, доколку командата е точна, а буквата Н, доколку командата е неточна.

- а) DIR D:/A\*.EXE            Т    Н
- б) COPY A:TETRIS.\* B:    Т    Н
- в) TYPE PROGRAM            Т    Н (6)
- г) DEL C:\CW/????.HFT    Т    Н
- д) DIR/W/P                    Т    Н

2. а) Кои DOS команди е потребно да се зададат на основниот именик OSNOVA на дискот С за да се креира следната структура:

```
\OSNOVA ----- BASIC
          \----- FORTRAN (3)
          \----- PASCAL
```

б) На кој начин ќе го отстраниме именикот BASIC во наведената структура ако во него се наоѓа датотеката IME.BAS. (3)

3. Дадена е низата броеви 0, 1, 3, 6, 10, 15, 21,... Да се состави алгоритам и напише соодветна компјутерска програма со која ќе се одреди најголемиот елемент од низата што е помал од 1000, како и неговиот реден број. (10)

4. Дадена е низата реални броеви  $a_1, a_2, \dots, a_n$ ,  $n \geq 2$ . Да се состави компјутерска програма со која најмалиот и најголемиот елемент во низата ќе си ги заменат местата. (8)

**Забелешка:** Компјутерските програми се пишуваат во програмски јазик по избор на кандидатот.

**05.07.1996**

---

1. Да се определи точноста на следните наредби напишани во MS-DOS оперативниот систем. Да се заокружат само точните одговори.

а) `c> copy *.* c:\temp\ispit`

б) `c> dir dir.dir`

в) `a> copy podatok.txt`

г) `c> rename ime1.txt`

д) `c> format /s a:`

2. Составете програма (во програмски јазик по Ваш избор) која го пресметува најголемиот и најмалиот елемент во една низа од  $n$  природни броеви. Испитот на Вашата програма да ги содржи:

- влезната низа од броеви
- вредноста на најмалиот елемент
- вредноста на најголемиот елемент

3. Да се состави програма (во програмски јазик по Ваш избор) која ги печати сите трицифрени броеви  $\overline{abc}$  за кои важи  $\overline{abc} = a + b^2 + c^3$ .

**Упатство:**  $a, b, c$  се цифри и  $\overline{abc} = 100a + 10b + c$ .

**Напомена:** Секоја точно решена задача носи 5 поени.

**12.09.1996**

1. Со заокружување означите дали наведените наредби во оперативниот систем MS-DOS се точни (Т) или неточни (Н):

а) label a: disk 1           Т Н

б) rename a: \*.bat \*.\*   Т Н

в) copy \*.bat               Т Н

г) backup c:\rabota\mnogu a:                           Т Н

д) type b:list.txt | more                           Т Н

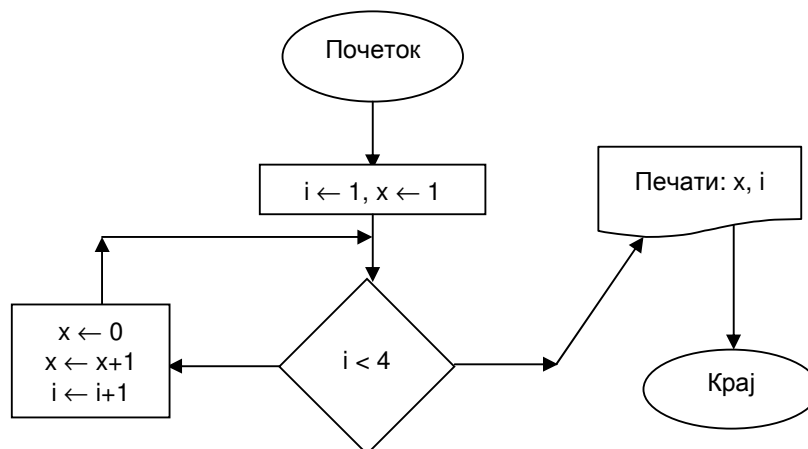
2. Заокружете ги точните вредности за  $x$  и  $i$  што ќе ги отпечати програмата дадена со следниов блок дијаграм:

а)  $x=4; i=4$

б)  $x=3; i=4$

в)  $x=1; i=4$

3. Да се



определат сите парови броеви кои собрани даваат еден двоцифрен број, а помножени го даваат бројот со обратните цифри. (На пример:  $3+24=27$  и  $3*24=72$ )

**Напомена:** Секоја точно решена задача носи 5 поени.

1. Да се определи правилноста на следните наредби напишани во MS-DOS оперативниот систем. При тоа да се заокружат само точните одговори.

а) `a>copy *.tnt c:\proba\dogovor`

б) `c>copy *,* /e d:\site`

в) `c>copy *.* /e d:\site` (5)

г) `dir/w/p`

д) `rename ispit.txt`

2. а) (6) Се наоѓате во именикот FAKULTET. Кои наредби би требало да се зададат во MS-DOS со цел да се креира следната податотечна структура:

```
FAKULTET  -----  PMF  -----  INFORMAT
           \-----  ELEKTRO  -----  SMETINFO
           \-----  FILOLOG  -----  ANGLISKI
                                           \-----  GERMANS
```

б) Напишете ја низата од наредби со која се брише датотеката REZULTAT.DOC што се наоѓа во именикот GERMANS, а моментално сме во именикот SMETINFO.

в) Со кои наредби се брише именикот INFORMAT, ако моментално се наоѓате во именикот ELEKTRO.

3. (6) Составете извршна (batch) податотека, која во именикот INFORMAT ги испишува на екран сите податотеки кои имаат наставка \*.rez со притиснување на било кој тастер, а за истото Ве информира со пораката "Сега ги испишувам сите податотеки со наставката \*.rez", а потоа истите ги отстранува (брише) од именикот со притискање на било кој тастер, повторно со известувањето "Сега ги бришам сите податотеки со наставката \*.rez". На крајот на екранот ја испишува пораката "Податотеките со наставка REZ се избришани", пак со притискање на било кој тастер.

4. (5) Одговорете на прашањата?

а) Што е бит?

б) Кој е најголемиот позитивен број што може да се претстави во сметач каде основниот збор има должина од 8 бита, а крајниот лев бит се употребува за знакот на бројот?

в) Наведете барем три вида на системска програмска опрема!

г) Кои се трите основни компоненти (делови) на централно процесната единица?

д) Која е разлика помеѓу RAM и ROM?

5.(8) Следниот дел од декларацијата во една постапка содржи грешки. Ваша задача е да ги откриете истите и накусо објасните?

```
procedure GRESKA (A:integer; var B:real);  
var A:integer; B:real;
```

6.(10) Кои се резултатите (излезните вредности) кои се добиваат со извршувањето на следната програма:

```
programa ZADACA (input, output);  
var A,B,C,D:integer;  
  procedure ZADA (C,A:integer; var D:integer)  
  var B:integer;  
  begin  
    a:=5; B:=6; C:=7; D:=8  
    writeln(A,B,C,D)  
  end;  
begin  
a:=1; B:=2; C:=3; D:=4  
writeln(A,B,C,D)  
ZADA(B,A,D);  
writeln(A,B,C,D)  
end.
```

7.(10) Составете програма (во програмски јазик по Ваш избор) со која се копира првобитно зададеното поле од 5 елементи ORIG (што секако би требало прво да се прочита), во поле OBRA, но со обратен редослед. Пример:

Ако ORIG е

2	1	3	1	4
2	0		5	9

тогаш OBRA е

49	15	3	10	22
----	----	---	----	----

**12.09.1997**

---

1. Да се определи правилноста на следните наредби напишани во MS-DOS оперативниот систем. При тоа да се заокружат само точните одговори.

а) `a>copi *.tnt c:\proba \dogovor`

б) `c>copy test.txt c:\`

в) `c>dir *.*` (5)

г) `c>type test.txt`

д) `c>compare proba.txt origano.txt`

2.(8) Одговори ги прашањата:

а) Што е бајт (слог)?

б) Кој е “најголемиот” негативен број што може да се претстави во сметач каде основниот збор има должина од 8 бита, а крајниот лев бит се употребува за знакот на бројот?

в) Наведете барем три вида на влезно/излезни направи (единици)!

г) Каде би требало да биде сместена една програма во процесот на нејзиното извршување?

3.(7) Што прави следната извршна податотека во MS-DOS?

```
a:  
cd\tempo  
copy c:\*.sys a:  
cd\postojan  
del *.txt  
novapoda
```

4.(15) Составете програма (во програмски јазик по Ваш избор) со која се:

- складираат 77 последователни фрлања на коцка (чии што страни се обележени со 1,2,3,4,5 и 6)
- пресметува колку пати во 77 фрлања се појавиле 1,...,6
- го пресметува процентот на појавувања на 1,2,...,6

**Напомена:**

Во претставувањето на проблемот да се ползува поле.

5.(15) Составете програма (во програмски јазик по Ваш избор) со која се:

- читаат податоци од некоја злезна податотека, при што секој податок е во нов ред (правилни влезни вредности се 0 или 1)
- го пресметува бројот на 0 и 1, како и вкупниот број на правилни влезни вредности
- застанува со работа ако прочита вредност различна од 0 или 1

**Пример:** 0

1

1

0

1

1

19

**Излез:** број на правилни вредности 6

број на нули: 2

број на единици: 4



1.(8) Да се определи точноста на следниве наредби напишани во MS-DOS. Да се заокружат само точните одговори.

- а) C:\TEXT>REN A\*.\* B\*.\*
- б) C:\>RENAME A.TXT B
- в) C:\>COPY B A:\WORD
- г) C:\>FORMAT \S A:
- д) C:\PROGRAMI>COPY A:\*.BAS
- ѓ) C:\PROGRAMI>MK ZADACI
- е) C:\>DIR/P/W A:\*.\*
- ж) C:\>DIR/W A:\*.PAS

2.(7) Да се даде одговор на следниве прашања:

- а) Што е оперативен систем? Наброј неколку попознати оперативни системи.
- б) Колку различни знаци може да се претстават во компјутер кој користи 8-битен процесор?
- в) Која функција се активира со едновремено притискање на тастерот Ctrl и буквата C (Ctrl+C)?
- г) Нека е активен дискот C. Како ќе ги излистате датотеките со наставка .BAS, од именикот A:\PROGRAMI, чии имиња започнуваат со буквата P?
- д) Која е вредноста на изразот:

$(x+y>z \text{ AND } y>5) \text{ OR } (x<y \text{ AND } z<50)$

ако  $x=2$ ,  $y=5$  и  $z=7$  ?

- ѓ) Која е најмалата должина на бинарните зборови за запишување на малите и големите букви на македонската кирилица?
- е) Најди го бинарниот еквивалент на децималниот број 321.

3.(10) На квалификациониот испит по информатика учествувале  $n$  ( $n<100$ ) кандидати. Постигнатите поени на кандидатите се дадени со бројната низа  $a_1, a_2, \dots, a_n$ . Ако минималниот број поени потребни за упис е  $\min$ , составете програма со која ќе се определи бројот на кандидатите што стекнале право за упис и ќе се пре-

смета нивниот среден успех покажан на квалификациониот испит.

4.(10) Составете програма со која ќе се прочита низа на знаци – име и презиме, одвоени со празно место, а потоа ќе се отпечати прво презимето па името, одвоени со празно место.

5.(15) Дадена е дробката  $\frac{a}{b}$ ,  $a < b$ ,  $a$  и  $b$  природни броеви. Составете програма за кратење на дробката се додека именителот и броителот не станат взаемно прости броеви.

**Забелешка:** Програмите се пишуваат во програмски јазик по Ваш избор.

1. (5) Да се определи правилноста на следните наредби напишани во MS-DOS оперативниот систем. При тоа, правилните наредби да се означат со **T**, додека неправилните (неточните) со **H**.

```
a) a>copy *.tnt c:\prva\vtora
б) c>copy *,* /e d:\ostanati
в) c>copy *.
г) dir/p/s
д) delete
```

- 2.(4)

- а) Што е бајт?
- б) Што е регистер?
- в) Што е алгоритам?
- г) Кои се основните функции на работната меморија?

- 3.(6) Објасни накусо што, во еден програмски јазик, претставуваат наредбите за:

- а) декларација
- б) избор
- в) повторување

4. (8) Што прави следната програма во BASIC, односно кој алгоритам е реализиран со неа?

```
11 FOR I=1 to N-1
12 M=A(I)
13 J=I
14 FOR K=I+1 TO N
15 IF A(K) >= A(J) THEN 18
16 M=A(K)
17 J=K
18 NEXT K
19 T=A(I)
20 A(I)=A(J)
21 A(J)=T
22 NEXT I
```

каде  $A(1)$  е еднодимензионално поле со елементи од множеството на целите позитивни броеви.

- 5.(12) Зборовите од јазикот ЧУДМАК (или ЧУДен МАКедонски) се добиваат на тој начин што се земаат зборовите од македонскиот јазик, и се преобразуваат т.ш. првиот вокал од оригиналниот збор станува почеток на новиот збор во ЧУДМАК. Сите претходни

букви до првиот вокал се додаваат на крајот од остатокот од оригиналниот збор, за потоа на тие букви да се додаде наставката АУ:

- од ВРАТА се добива АТАВРАУ
- од ИСПИТ се добива ИСПИТАУ
- од ГЛАВА се добива АВАГЛАУ
- од ОКО се добива ОКОАУ

Составете програма која преведува од МАКЕДОНСКИ на ЧУДМАК во програмски јазик по Ваш избор (препорака: BASIC или Pascal).

6. (15) Една од основните теореми во аритметиката тврди дека секој позитивен цел број е производ од единствено множество прости броеви, односно од броеви кои се делливи само со себе и бројот 1. Така за бројот 12 имаме  $12 = 2 \cdot 2 \cdot 3$ , за бројот 121 =  $11 \cdot 11$ , додека за бројот  $250 = 2 \cdot 5 \cdot 5 \cdot 5$ . Составете програма, во програмски јазик по Ваш избор, која ги пресметува простите фактори на еден позитивен цел број (препорака: BASIC или Pascal).

Кралот на Гази Баба имал еден убав бел коњ, имал прекрасна принцеза и изградил нов замок. Во кралството имало 6 витези и тој се размислувал кому да му го предаде коњот, кому да му го предаде замокот, а кому принцезата. За таа цел тој ги повикал витезите на вечера. Витезите добивале редни броеви онака како што пристигнувале. Витезот број 1 бил познат по своите обиди од физиката и метафизиката. Втор бил витезот кој правел експерименти и магии со прашоци, метали и течности. Третиот витез најмногу го интересирало испитувањето на непознатите земји и мориња. Витезот број 4 бил преокупиран со одгледување на билки и животни. Следниот витез познавал многу геометриски и аритметички тајни. Последниот витез бил најмлад во кралството и го занимавале современи компјутерски мрежи.

1. (10) За да го предаде замокот кралот имал идеја да види од кој витез ќе собере најмногу пари како данок во кралството. Затоа тој побарал витезите да му кажат колкав дел од средствата што ќе ги соберат од поданиците ќе му ги дадат во форма на данок. Витезот број 1 понудил 3% од приходите на неговите поданици. Вториот витез понудил 2% од приходите на своите поданици сметајќи дека има најмногу поданици. Третиот витез понудил 5% од приходите на своите поданици сметајќи дека има малку поданици. Витезот број 4 понудил 4% од приходите на своите поданици. Следниот витез понудил за првиот поданик една паричка, а за секој следен поданик по една паричка повеќе од претходниот. Последниот витез понудил дека има можност да собере по три парички од сите поданици во кралството, без разлика дали му припаѓаат нему или не. Кралот го прашал советникот да му каже колку поданици припаѓаат на секој витез за да може да пресмета кој витез му нуди најмногу парички. Се претпоставува дека приходите на поданиците изнесуваат 200 парички. Советникот го земал својот рачен компјутер ABACUS и во него ги внел бараните податоци за абројот на поданиците предодредени на секој витез. Рачниот компјутер потоа му дал одговор кој витез нуди најмногу пари.

Направи програма во програмски јазик по желба која има намена да пресмета колку парички нуди секој витез, на основ на тоа што рачно треба да се внесат податоци за бројот на поданиците за секој од витезите!

2. (8) За да се одлучи кому ќе му го даде својот бел коњ го натерал советникот да им постави едноставни MS-DOS проблеми за својот рачен компјутер ABACUS. За таа цел препиши ги во тетратката следните MS-DOS наредби и покрај нив запиши Т и ⊥. Заокружи го знакот за кој сметаш дека одговара на точно напишаната наредба.

а) Т C:\DIR  
⊥ A\*.B?C

б) Т C:\>RAN A\*.\*  
⊥

в) Т C:\>COPYFILE  
⊥ A\* A:

г) Т C:\>RENAME  
⊥ A\*.\* B\*.\*

д) Т C:\>DIR  
⊥ A:B?.C\*

ѓ) Т B:\>DIR C:\  
⊥ /W

е) Т C:\>COPY  
⊥ A\*.\*

ж) Т C:\B\A>MK D  
⊥

3. (5) За да се одлучи кому ќе му ја даде принцезата кралот се одлучил да приреди витешки игри. Претходно поставил квалификациоинен тест за проверка на основните знаења за неговиот компјутер ABACUS. Тестот се состоел од прашања на кои витезите одговарале со Т и ⊥ во зависност дали поставената реченица е точна или не. За таа цел препиши ги во тетратката следните реченици и покрај нив запиши Т и ⊥. Заокружи го знакот за кој сметаш дека одговара на точно напишаната наредба.

а) Т RAM е процесор.  
⊥

б) Т Преведувач е програмска  
⊥ опрема.

в) Т Програмирање е постапка за  
⊥ изведување на програмите.

- г) Т Едитор е програма за пишување  
└ програми.
- д) Т Глувче е влезно-излезна  
└ единица.

4. (7) Само 3 витези го поминале квалификациониот тест. Тие учествувале на пет витешки игри. Секој од витезите имал можност да добие златен, сребрен или бронзен медал за секоја витешка игра. Бидејќи ниеден од витезите не ги освоил сите златни медали кралот имал тешка задача да избере кому ќе му ја подари принцезата. Затоа советникот го бодувал златниот медал со 5 бодови, сребрениот со 3 бодови и бронзениот медал со 2 бодови. За секоја од игрите во својот рачен компјутер ABACUS советникот внел колку бодови освоил секој од трите витези. Компјутерот потоа го соопштил редниот број на витезот што освоил најмногу бодови.

Направи програма во програмски јазик по желба со намена да пресмета кој витез добил најмногу бодови, на основ на тоа што рачно треба да се внесат бараните податоци за бројот на бодови за секој витез во секоја од игрите.

**05.07.1999**

---

На кралот на Гази Баба му поминал мандатот да го одржува кралството па морал да го напушти тронот. Се пријавиле повеќе кандидати-принцови да раководат со кралството. Затоа на главниот мудрец во кралството му поставиле задача како да избере вистински престолонаследник. Тој поставил повеќе загатки и тестови за снаодливост на принцовите што се пријавиле.

- a) `C:\>EXECUTE EXE.EXE`  
Т  
⊥
- б) `C:\>DIR A:B*C.D?E`  
Т  
⊥
- в) `C:\>RENAME A*.* B*.*`  
Т  
⊥
- г) `C:\>CALL A`  
Т  
⊥
- д) `A:\>DIR /P`  
Т  
⊥
- ѓ) `C:\>DIRECTORY ABC`  
Т  
⊥
- е) `C:\B\A>MK D`  
Т  
⊥
- ж) Алгоритам е програма за решавање проблеми.  
Т  
⊥
- з) ROM е меморија.  
Т  
⊥
- с) Интернет е програмски јазик.  
Т  
⊥
- и) Текст процесор е програма за уредување на текстови.  
Т



⊥

1. (11) Првата загатка што принцовите требало да ја решат се сведувала на тестови на знаење за рачниот компјутер ABACUS на мудрецот и во тогашно време современите MS-DOS наредби. За таа цел во тетратката препиши ги следните MS-DOS наредби, како и реченици што се однесувале на компјутерот ABACUS. По крај нив запиши T и ⊥. Заокружи го знакот за кој сметаш дека одговара на точно напишаната наредба или реченица.
2. (9) Секој принц што можел да одговори на прашањата поминувал на следното ниво на тестови за снаодливост. Тестот се сведувал на тоа да го поминат лавиринтот и кога ќе ги пречека мудрецот да одговорат колку вкупно мечеви виделе во собите низ кои поминувале. Лавиринтот се состоел во тоа што во даден момент принцовите ги пуштале да отворат врата, да поминат низ собата и на крајот од собата да отворат следна врата. По пат треба да ги бројат мечевите што се во собата. Она што всушност тие го знаеле однапред е дека во секоја следна соба имало два меча повеќе од претходната соба. Од друга страна времето на поминување им било ограничено, па тие едноставно трчале низ собите и немале доволно време да бројат и додаваат. Нивниот проблем се сведувал на тоа што можеле да ги избројат мечевите во првата соба додека им се дозволило да поминат во следната соба. Потоа морале да трчаат се до мудрецот во некоја од собите и можеле да бројат само колку соби ќе претрчаат. Мудрецот имал соодветна програма за својот рачен компјутер ABACUS и ги внесувал податоците за бројот на мечеви во првата соба и бројот на соби што принцовите ги поминувале.

Направи програма во програмски јазик по желба со намена да пресмета колку мечеви имало во собите на основа на двата податоци што мудрецот ги внесувал.

3. (10) Последниот тест бил тест на снаодливост и борба. Затоа принцот се пуштал кај многуглавниот змеј. За принцот постоела загатка во тоа што никогаш не се знаело во каква форма ќе се појавува змејот. Тој можел да се појавува со повеше големи, средни и мали глави. Вештината на принцот се состоела во тоа да пресмета колку вкупно јаже ќе му треба да ги заврзе сите глави на змејот додека истиот спиел, но така да не му остане вишок на јаже. Ако пак земел премалку јаже, тогаш змејот ќе го напад-

нел со слободната глава и принцот ќе бил погубен. За секоја голема глава на змејот било потребно по пет метри јаже, за средна по три метри јаже и за секоја мала глава по два метри јаже. Мудрецот дозволувал на принцот да земе помошници според правилото ако змејот има до 5 глави се зема еден помошник, ако има повеќе од 5 глави може да се земат два помошника, ако има повеќе од 10 глави може да се земат три помошника, ако има повеќе од 15 глави може да се земат четири помошника итн. Меѓутоа, сето тоа се случувало неверојатно брзо бидејќи змејот ја менувал формата на секоја минута. Затоа принцот брзо ги броел главите на змејот и на рачниот компјутер ABACUS на мудрецот го внесувал бројот на големи, бројот на средни и бројот на мали глави на змејот. Компјутерот издавал два броја како резултат, прво колку метри јаже е потребно, а второ колку помошници треба да се земат за да може во дадената минута да се совлада змејот.

Направи програма во програмски јазик по желба со намена да се пресметаат бараните броеви на основ на трите податока што мудрецот ги внесувал.

1. (10) Да се определи вистинитоста на следните изјави (со заокружување на соодветниот атрибут покрај изјавата, имено В - за вистината, и Н - за неvistинитата изјава.
- а) В Н Аритметичко-логичката единица (АЛЕ) е дел од ЦПЕ (централно-процесната единица) во кој се извршуваат сите аритметички и логички операции.
  - б) В Н Механизмот за адресирање се користи за именување на променливите кои се појавуваат во рамките на една промена.
  - в) В Н Еден бајт (слог) означува количество на меморија потребно за складирање на два знака или симболи.
  - г) В Н Централно-процесната единица може да ги чита наредбите во делот од меморијата ROM, но не може да складира податоци во истиот.
  - д) В Н Под запис се подразбира збирка на податочни елементи помеѓу кои постои некоја логичка врска.
  - ѓ) В Н Под основа или база на еден позиционен броен систем се подразбира бројот на важечки цифри кои не може да се појават на една позиција
  - е) В Н Поле претставува збирка од податоци кои би требало да имаат најмалку еден заеднички атрибут.
  - ж) В Н Под кодирање се подразбира процесот на преведување на алгоритам во програмски јазик.
  - з) В Н Објектна програма е секоја програма која претставува влез за еден преведувач.
  - и) В Н Основната разлика помеѓу сметач и калкулатор е особината на првиот да ги складира наредбите кои може да ги извршува.
2. (10) Составете програма во Pascal која на почетокот создава пакувано поле (еднодимензионално) од десет алфамерички знаци (на пример: PROVER POL). Потоа, Вашата програма би требало да ги помести кружно знаците во полето за три места на десно. Имено, ако влезот е:

P	R	O	V	E	R		P	O	L
---	---	---	---	---	---	--	---	---	---

по извршувањето на Вашата програма, излезот е:

P	O	L	P	R	O	V	E	R	
---	---	---	---	---	---	---	---	---	--

3. (10) Составете програма во Pascal која ја содржи потпрограмата MAXODTRI, за читање на три цели броеви и определување на најголемиот од нив (можете слободно да претпоставите дека броевите се различни).

Главната програма чита шест цели броеви (кои се внесуваат преку тастатурата). Потоа помош на потпрограмата MAXODTRI го наоѓа максимумот од дадените броеви.

- 1.(4) Да се определеи вистинитоста на следните изјави (со заокружување на соодветниот атрибут покрај изјавата, имено В – за вистинита, и Н – за неvistинита изјава).
- а) В Н Акумулаторот е регистер кој не се употребува и во процесот на пресметување на реалните адреси.
  - б) В Н Адресирањето е механизам за пресметување на адресата на местото во меморијата каде е складиран еден податок/наредба.
  - в) В Н Со еден бит се означува најголемото количество на информации што може да се складира во меморијата на еден компјутер.
  - г) В Н ROM е вид на меморија во која не може да се запишуваат нови податоци/наредби.
  - д) В Н Природата на полупроводничките елементи кои се вградени во секој компјутер го прави бинарниот броен систем соодветен за претставување на податоците и наредбите.
  - ѓ) В Н Правилната наредба за повторување мора да содржи и услов за излегување од структурата.
  - е) В Н Екранот на еден компјутер претставува излезна направа.
  - ж) В Н Алгоритам е постапка за пресметување која секогаш завршува без разлика на вредностите кои го определуваат влезот.
- 2.(6) Составете програма (постапка, функција) во Pascal/BASIC со која се сумираат елементите на едно-димензионално поле од реални броеви. Може да претпоставите дека бројот на елементите во полето е  $n$ .
- 3.(9) Составете програма во Pascal/BASIC која како влез зема еден позитивен цел број, а како излез дава позитивен цел број во кој редот на цифрите е обратен од редот на цифрите определен со влезот. Пример: влез  $\leftarrow 37891$  , излез  $\leftarrow 19873$ .
- 4.(11) Составете програма (постапка, функција) во Pascal/BASIC со која се пресметува најмалиот делител (различен од единица) на даден позитивен цел број.

1.(5) Да се определи вистинитоста на следните изјави (со заокружување на соодветниот атрибут покрај изјавата, имено В – за вистинита, и Н – за неvistинита изјава).

а) В Н `A>COPY *.TNT C:\PROBA\DOGOVOR`

б) В Н `C>COPY *,* /E D:\SITE`

в) В Н `C>COPY *.* /E D:\SITE`

г) В Н `DIR/W/P`

д) В Н `RENAME ISPIT.TXT`

2.(5) Одговори ги прашањата:

а) Што е бит?

б) Кој е најголемиот позитивен број што може да се претстави во компјутер каде основниот збор има должина од 8 бита, а крајниот лев бит се употребува за знакот на бројот?

в) Наведете барем три вида на системска програмска опрема?

г) Кои се трите основни компоненти (делови) на централно процесната единица?

д) Што е разликата меѓу RAM и ROM?

3.(10) Дадени се два искази

A: Мојата тетка живее во Скопје.

B: Мојата баба живее во Скопје.

Составете програма (во програмски јазик по Ваш избор) со кој се проверува дали два искази X и Y се еквивалентни (два искази се еквивалентни или ако и двата искази се вистинити или ако и двата искази се лажни). На пример, X и Y се сложени изјави од типот:

X: Не е вистина дека мојата тетка или мојата баба живеат во Скопје.

Y: Ниту мојата тетка ниту мојата баба живеат во Скопје.

4.(10) Составете програма (во програмски јазик по Ваш избор) со која се копира првобитно зададеното поле од 5 елементи ORIG (што секако би требало прво да се прочита), во поле OBRA, но со обратен редослед на елементите. На пример:

Ако ORIG е

22	10	3	15	49
----	----	---	----	----

Тогаш, ОБРА е

49	15	3	10	22
----	----	---	----	----

**Група 1**

- 1.(7) Пред точниот исказ да се стави знакот ✓, а пред неточниот ×:
- а) Внатрешната меморија на компјутерот е многу поголема од надворешната.
  - б) Microsoft Word, Lotus 1-2-3, CorelDraw се апликативни програми.
  - в) Со командата Save As се запишува документот под друго име.
  - г) Збирот на броевите 101101 и 10111 е 1110000.
  - д) Бајт е група од 4 бита.
  - ѓ) FORTRAN, C, BASIC се програмски јазици.
  - е) Преведувач е програмски јазик кој ја преведува изворната програма.
- 2.(5) Да се напише алгоритам или програма за подредување на три броја a, b и c по големина.
- 3.(8) Да се напише алгоритам или програма за пресметување на збирот:
- $$1+(1+2)+(1+2+3)+(1+2+3+4)+\dots+(1+2+3+4+\dots+n).$$
- 4.(10) Да се напише подалгоритам или потпрограма за претварање на децимален декаден број во бинарен до 6-децимали. Потоа да се напише алгоритам или програма во кој ќе се повикува подалгоритмот односно потпрограмата.

**Група 2**

- 1.(7) Пред точниот исказ да се стави знакот ✓, а пред неточниот ×:
- а) Брзината на компјутерот се изразува преку висината на фреквенцијата на процесорот.
  - б) Капацитетот на магнетните дискови се изразува во бајти (B) и херци (Hz).
  - в) Windows и DOS се оперативни системи.
  - г) Copy и Paste се команди за пренесување дел од документот од едно на друго место..
  - д) Производот на броевите 10011101 и  $2^3$  е 10011101000.
  - ѓ) Преведувач е програмски јазик кој ја преведува изворната програма.



е) Internet е апликативна програма за пренесување на пораки меѓу два компјутери.

2.(5) Да се напише алгоритам или програма за наоѓање на спротивниот број на природниот број  $n$  ( $n < 1000$ ).

Пример:  $n=274$  спротивен е 472

$n=73$  спротивен е 37

$n=9$  спротивен е 9

3.(8) Да се напише алгоритам или програма за пресметување на збирот:

$(1+2+3+4+\dots+n) + (2+3+4+\dots+n) + (3+4+\dots+n) + \dots + ((n-1)+n)+n$ .

4.(10) Да се напише подалгоритам или потпрограма за претварање на цел декаден број во бинарен. Потоа да се напише алгоритам или програма во кој ќе се повикува подалгоритмот односно потпрограмата.

- 1.(8) Пред точниот исказ да се стави знакот ✓, а пред неточниот ×:
- а) Внатрешната меморија на компјутерот е многу помала од надворешната.
  - б) Брзината на процесорот се изразува во број на извршени операции во секунда.
  - в) Тастатурата, глумчето и печатачот се влезни уреди.
  - г) Најпознат оперативен систем денес е Windows.
  - д) Компјутерските вируси се најмалите програми во компјутерот.
  - ѓ) Microsoft Word е апликативна програма за цртање.
  - е) Со командата Open се отвора нов документ во апликациите.
  - ж) Internet е апликативна програма за пренесување на пораки меѓу два компјутери.
- 2.(8) Пред точниот исказ да се стави знакот ✓, а пред неточниот ×:
- а) 0 и 1 се битови.
  - б) Бајт е група од 8 бита.
  - в) PASCAL, C и BASIC се програмски јазици.
  - г) Збирот на бинарните броеви  $1001_2$  и  $1101_2$  е  $2102_2$ .
  - д) Преведувач е програмски јазик кој ја преведува изворната програма.
  - ѓ) Контролните структури служат за контрола на дејството во алгоритмите.
  - е) IF-THEN-ELSE е наредба за избор (гранање).
  - ж) Логичките променливи можат да ја имаат која било вредност.
- 3.(6) Да се напише алгоритам или програма за одредување дали природниот број  $n$  е парен или непарен.
- 4.(8) Да се напише алгоритам или програма за наоѓање на аритметичката средина на првите  $n$  природни броеви.

- 1.(7) Пред точниот исказ да се стави знакот ✓, а пред неточниот ×:
- а) Меморијата на компјутерите се изразува во бајти и поголеми единици од него.
  - б) Брзината на процесорот се изразува во мегахерци (MHz).
  - в) Со командата Save As се запишува документот под друго име.
  - г) Декадниот број 10 има бинарен запис 110.
  - д) Компјутерските вируси се всушност компјутерски програми.
  - ѓ) Microsoft Excel е апликативна програма за уредување на слики.
  - е) ana@freemail.com.mk е Интернет страница.

- 2.(5) Што пресметува следниот алгоритам?

**алгоритам А;**

**почеток**

**печати** 'Внеси два природни броја m и n:';

**читај** m,n;

a ← m;

b ← n;

**повторувај**

**ако** m < n

**тогаш**

n ← n - m;

**инаку**

m ← m - n;

**крај\_ако**{m < n}

**до** m = n;

**крај\_повторувај**{m = n}

**печати** '\_\_\_\_\_ на броевите ', a, ' и ', b, ' е ', m;

**крај** {A}

- 3.(7) Да се напише алгоритам или програма за одредување на бројот на појавувања на буквата А во зборот АБРАКАДАБРА.
- 4.(11) Природните броеви кои се еднакви на збирот на своите делители (без самиот број) се наречени совршени броеви. На пример, совршени броеви се:  $6 = 1 + 2 + 3$ ,  $28 = 1 + 2 + 4 + 7 + 14$

и други. Да се напише алгоритам или програма за наоѓање на совршените броеви до 1000.

- 1.(7) Пред точниот исказ да се стави знакот ✓, а пред неточниот ×:
- а) Множеството на програми кои се користат во компјутерот се нарекува софтвер.
  - б) Капацитетот на меморијата се изразува во бајти.
  - в) Најпознат оперативен систем е Microsoft.
  - г) Декадниот број 10 има запис 1010.
  - д) Pascal, Basic, C се програмски јазици.
  - ѓ) Microsoft Word е апликативна програма за уредување на текст.
  - е) [www.pmf.ukim.edu.mk](http://www.pmf.ukim.edu.mk) е Интернет страница.

2.(5) Што пресметува следниот алгоритам?

**алгоритам А;**

**почеток**

**печати** 'Внеси еден природен број m:';

**читај** m;

s ← 0;

p ← 1;

**за** i ← 1 **зголемувај до** m

s ← s + i;

p ← p \* i;

**крај\_за**{i}

**печати** '\_\_\_\_\_ =', s;

**печати** '\_\_\_\_\_ =', p;

**крај** {A}

3. (7) Што пресметува следниот алгоритам?

**алгоритам Б;**

**почеток**

**печати** 'Внеси еден природен број n:';

**читај** n;

bcif ← 0;

**додека** n > 0 **извршувај**

n ←  $\left[ \frac{n}{10} \right]$ ;

bcif ← bcif + 1;

```
крај_додека{n > 0}
печати 'Бројот ', n, ' има ', bcif, ' _____';
крај {Б}
```

4.(11) Да се напише алгоритам или програма за наоѓање на парните елементи во бројната низа  $a_1, a_2, a_3, \dots, a_n$ .

- 1.(7) Пред точниот исказ да се стави знакот ✓, а пред неточниот ×:
- а) Најраспространет (најкористен) оперативен систем за персонални компјутери е MS Word.
  - б) Фреквенцијата на денешните процесори за персонални компјутери е неколку MHz.
  - в) Со командата Save As се запишува документот под друго име.
  - г) Бинарните број 1010 има декаден запис 10.
  - д) Microsoft Excel е апликативна програма за уредување на слики.
  - ѓ) Internet Explorer е најпознат разгледувач на Web-страници.
  - е) www.pmf.ukim.edu.mk е Интернет страница.

2.(5) Што пресметува следниот алгоритам?

**алгоритам Н;**

**почеток**

**читај** a,b,c;

**ако** a>b

**тогаш**

            p←a

**инаку**

            p←b

**крај\_ако**{a>b}

**ако** p>c

**тогаш**

            n←p

**инаку**

            n←c

**крај\_ако**{p>c}

**печати** n;

**крај** {Н}

3. (7) Да се напише алгоритам или програма за наоѓање на збирот и производот на елементите на низата  $[a_i]_n$ .

4. (11) Да се напише алгоритам или програма за кретење на дробката  $\frac{a}{b}$ , каде a и b се цели броеви.

Напомена: Програмите може да се пишуваат во програмски јазик по избор на кандидатот.

Време за работа: 120 минути.



1.(10) Во квадратчето пред точниот исказ да се стави знакот ✓ (☑), а пред неточниот знакот × (☒):

а) Компјутерот може да обработува податоци изразени во форма на:

- број       текст       слика       звук       мирис  
 вкус

б)  Модемот е уред кој служи за поврзување на компјутерот на телефонската мрежа.

в) Печатачите можат да бидат:

- ласерски       звучни       млазни       линиски        
матрични

г) Оперативни системи се:

- Windows       Word       Microsoft       Pascal

д)  Датотека е логичка целина од податоци запамтени под едно име.

ѓ)  Microsoft Word е апликативна програма за цртање.

е)  Со компандата Save As се запамтува документ во апликациите.

ж)  Internet е апликативна програма за пренесување на пораки меѓу два компјутери.

2.(10) Во квадратчето пред точниот исказ да се стави знакот ✓ (☑), а пред неточниот знакот × (☒):

а) Меморијата на компјутерот може да се изрази во:

- KB       MB       KG       GB       MHz

б) Програмски јазици се:

- Microsoft       Pascal       Internet       Basic       C

в)  Фонт е алатка за пишување на компјутер.

г)  Вие полагате приемен во  $11111010101_2$  година.

д)  Преведувач е програмски јазик кој ја преведува изворната програма во извршна.

ѓ) Контролни структури се:

Windows       Word       Microsoft       Pascal

е) Логичките променливи можат да имаат вредност:

True       Yes       End       False

**3.(4)** Да се напише алгоритам или програма за одредување на геометриската средина на  $n$ -броеви.

**4.(6)** Да се напише алгоритам или програма за подредување на три броја по големина.

- 1.(7) Пред точниот исказ да се стави знакот ✓, а пред неточниот ×:
- а) Надворешната меморија на компјутерот е многу помала од внатрешната.
  - б) Microsoft Word, Excel, CorelDraw се апликативни програми.
  - в) Со командата Save се запишува документот под друго име.
  - г) Збирот на броевите 101001 и 10111 е 1101000.
  - д) Бајт е група од 4 бита.
  - ѓ) Pascal, C, Basic се програмски јазици.
  - е) Преведувач е програмски јазик.
  - ж) Internet е апликативна програма за пренесување на пораки меѓу два компјутери.
- 2.(4) Да се напише алгоритам или програма за подредување на три броја a, b и c по големина.
- 3.(7) Да се напише алгоритам или програма за наоѓање за наоѓање на сите Питагорини броеви до 100 (Питагорини броеви се тројките (x, y, z) за кои важи  $x^2+y^2=z^2$ ).
- 4.(12) Да се напише алгоритам или програма за пресметување на збирот  $1+(1+2)+(1+2+3)+(1+2+3+4)+\dots+(1+2+3+\dots+n)$ .

# Решенија

**06.07.1995**

---

1. а) Н б) Н в) Т г) Н д) Т

2.

```
a)
c:
cd \osnova
md basic
md fortran
md pascal
```

```
б)
c:
cd \osnova
del basic
rd basic
```

3.

```
program NajgolemPomal1000;
var n,c:integer;
begin
  n:=0;
  c:=1;
  while n<1000 do
  begin
    n:=n+c;
    c:=c+1;
  end;
  c:=c-1;
  n:=n-c;
  writeln('Najgolemiot broj vo nizata pomal od 1000 e ',
          n);
  writeln('Toj broj e na pozicija ',c,' vo nizata');
end.
```

4.

```
program Zamena;
var n,i,min,max:integer;
    t:real;
    a:array[1..200] of real;
begin
  write('Vnesi ja dolzhinata na nizata: ');
  readln(n);
  for i:=1 to n do
  begin
    write ('Vnesi go a['',i,'']: '); readln (a[i]);
  end;
  min:=1;
  max:=1;
  for i:=1 to n do
  begin
    if a[min]>a[i] then min:=i;
    if a[max]<a[i] then max:=i;
  end;
end.
```

```
end;  
t:=a[min];  
a[min]:=a[max];  
a[max]:=t;  
for i:=1 to n do  
    writeln ('a[' ,i, ']=' ,a[i]);  
end.
```

1.

а) Т б) Т в) Н г) Н д) Т

2.

```
program NajgolemNajmal;
var n,i,min,max:integer;
    a:array[1..200] of integer;
begin
    write('Vnesi ja dolzhinata na nizata: ');
    readln(n);
    for i:=1 to n do
    begin
        write('Vnesi go a['i,']: '); readln (a[i]);
    end;
    min:=1; max:=1;
    for i:=1 to n do
    begin
        if a[min]>a[i] then min:=i;
        if a[max]<a[i] then max:=i;
    end;
    for i:=1 to n do
        writeln ('a['i,']='a[i]);
    writeln ('Najmal element e ',a[min]);
    writeln ('Najgolem element e ',a[max]);
end.
```

3.

```
program Tricifreni;
var a,b,c:integer;
    i:integer;
begin
    for a:=1 to 9 do
        for b:=0 to 9 do
            for c:=0 to 9 do
                if (100*a+10*b+c = a+b*b+c*c*c)
                    then writeln (100*a+10*b+c);
            end;
        end;
    end;
end.
```

1. a) H

б) H

в) H

г) T

д) T

2. в)  $x=1; i=4$

3.

```
program Broevi;
var a,b,c:integer;
    i,x,y:integer;
begin
    for a:=1 to 99 do
        for b:=1 to a do
            begin
                c:=a+b;
                if c>=10 then
                    begin
                        x:=c mod 10;
                        y:=(c div 10) mod 10;
                        if a*b=10*x+y then writeln(a,' ',b);
                    end;
            end;
        end;
    end;
end.
```

1. а) Т б) Н в) Т г) Т д) Н

2.

а)

```
md pmf
md elektro
md filolog
cd pmf
md informat
cd ..\elektro
md smetinfo
cd ..\filolog
md angliski
md germans
cd ..
```

б)

```
cd ..\..\elektro
del rezultat.doc
```

в)

```
cd ..\pmf
del informat
rd informat
```

3.

```
echo Sega gi ispishuvam site podatoteki so nastavkata *.rez
pause
dir c:\fakultet\pmf\informat\*.rez
echo Sega gi brisham site podatoteki so nastavkata *.rez
pause
del c:\fakultet\pmf\informat\*.rez
echo Podatotekite so nastavka REZ se izbrizhani
pause
```

4. а) Бит е една бинарна цифра.

б) 127

в) Оперативен систем, програми за преведување, услужни програми

г) Централна меморија, аритметичко-логичка единица, управувачка единица

д) Најважна разлика е можноста за запишување/измена на содржината во RAM-от; можност која ја нема кај ROM-от.

5. Во заглавието на процедурата и во делот за декларации на локални променливи постојат дупликати променливи (променливите А и В).

6. Ако се игнорира фактот што програмата има голем број синтаксни грешки и освен пораките за грешки не би давала никакви резултати, по исправувањето на истите, резултатите ќе бидат:

```
1234
5678
1238
```

7.

```
program OrigObra;
var orig, obra: array [1..5] of integer;
    i: integer;
begin
```



```
for i:=1 to 5 do begin
    write ('Vnesete go ',i,'-tiot element na ORIG: ');
    readln (orig[i]);
end;
for i:=1 to 5 do obra[5-i+1]:=orig[i];
for i:=1 to 5 do writeln('OBRA[' ,i, ']=' ,obra[i]);
end.
```

1. а) Н б) Т в) Т г) Т д) Н
2. а) Регистар од (вообичаено) 8 единечни мемориски ќелии (битови).  
б) -128  
в) Тастатура, екран, печатач, глумче  
г) Во централната меморија.
3. На дискетата во дискетната единица А:, во подименикот Tempo на главниот именик ги копира сите датотеки со наставка SYS од главниот именик на дискот С:. Потоа во подименикот Postojan на главниот именик на дискетата ги брише сите датотеки со наставка TXT. На крај, ја стартува програмата NOVAPODA од истиот именик, или пак од некој од имениците наведени во PATH системската променлива (во соодветен редослед).

4.

```
Program Kocka;  
Type  kocka = 1..6;  
const n=77;  
var  
    frlanje: array[1..n] of kocka;  
    pati: array[kocka] of integer;  
    i:integer;  
begin  
    randomize;  
    for i:=1 to 6 do pati[i]:=0;  
    for i:=1 to n do frlanje[i]:=random(6)+1;  
    for i:=1 to n do pati[frlanje[i]]:=pati[frlanje[i]]+1;  
    for i:=1 to n do writeln ('Frlanje ',i,': ',frlanje[i]);  
    for i:=1 to 6 do  
        begin  
            write  (i,' se pojavi ',pati[i],' pati,');  
            writeln (' shto e ',(pati[i]*100/n):0:2,' procenti.');        end;  
end.
```

5.

```
program datoteka;  
var input:text;  
    br0,br1,brpr:integer;  
    c:string;  
begin  
    assign(input,'in.txt');  
    reset(input);  
    c:=''; br0:=0; br1:=0;  
    while (not eof(input)) and (c='') do  
        begin  
            readln(input,c);
```

```
        if c='0' then begin inc(br0); c:=''; end
        else if c='1' then begin inc(br1); c:=''; end
    end;
    writeln ('Broj pravilni vrednosti: ',br0+br1);
    writeln ('Broj na nuli: ',br0);
    writeln ('Broj na edinici: ',br1);
end.
```

1. а) Т б) Т в) Т г) Н д) а Т ё) Н е) Т ж) Т
2. а) Оперативен систем е делот од системската програмска опрема на еден сметач задолжен за управување со работата на сметачот, контрола над хардверските единици, управување со податоците и др. Оперативниот систем е врската меѓу програмите обработка на податоците и машинскиот дел од сметачот.

MSDOS, Unix, VMS, OS/2 ...

б) Не постои директна корелација.

в) Под MSDOS се повикува функцијата за прекин. Под Windows се активира копирање на означен објект на Clipboard.

г) `dir a:\programi\p*.bas`

д) TRUE

ё) 6 бита

е) 101000001

3.

```
Program upis;
const
  min=60;
var
  a:array[1..100] of integer;
  n,i,sum,count:integer;

begin
  write('Vnesi go brojot kandidati shto polagale: ');
  readln(n);

  for i:=1 to n do
  begin
    write ('Kandidatot ',i,' osvoil poeni: ');
    readln(a[i]);
  end;

  sum:=0;
  count:=0;
  for i:=1 to n do
    if a[i]>min then
    begin
      sum:=sum+a[i]; inc(count);
    end;

  writeln('Kvalifikacioniot ispit go polozhile ',
    count,' kandidati.');
```

```

        writeln('Sredniot postignat uspeh kaj niv e ',
                sum/count:0:2);
end.

```

4.

```

program ImePrezime;
var
    s:string;
    ime,prezime:string;
    i:integer;
begin
    write ('Vnesi Ime i Prezime: ');
    readln (s);

    i:=1;
    while (s[i]<>' ') and (i<=length(s)) do
    begin
        ime:=ime+s[i];
        inc(i);
    end;
    inc(i);
    while (i<=length(s)) do
    begin
        prezime:=prezime+s[i];
        inc(i);
    end;
    writeln(prezime,' ',ime);
end.

```

5.

```

program kratenje;
var
    a,b,cut:integer;
begin
    writeln('Vnesi go broitelot');
    readln(a);
    writeln('Vnesi go imenitelot');
    readln(b);

    cut:=2;
    repeat
    while (a mod cut=0) and (b mod cut=0) do
    begin
        a:=a div cut;
        b:=b div cut;
    end;
    inc(cut);
    until cut>a;
    writeln(a,'/',b);
end.

```

1. а) Т б) Н в) Н г) Т д) Н
2. а) Низа од битови, најчесто осум, со која се оперира како целина.
  - б) Активна мемориска единица во рамките на процесорот. Во регистарот се извршуваат аритметички-логичките операции (акумулатор), се чуваат наредби во текот на обработката (регистар на наредби), се држи адресата на податокот кој ќе биде обработен (адресен регистар) и др.
  - в) Множество упатства кој во конечен број чекори, одделни операции, решаваат поставен проблем.
  - г) Помнење на програмите и податоците потребни за тековната обработка. Се користат и имињата централна меморија, главна меморија и внатрешна меморија.
3. а) Наредбите за декларација претставуваат дел од програмата каде се определува доменот на вредности на променливите и се резервира соодветен мемориски простор за сместување на тие вредности.
  - б) Наредбите за избор претставуваат форма на разгранета линиска структура и овозможуваат условна контрола на правецот на извршување на една програма, но строго линиски.
  - в) Наредбите за повторување претставуваат форма на циклична структура каде се овозможува повторување на извршувањето на дел од програмата и тоа претходно одреден конечен број пати или пак повторување на извршувањето со или до исполување на некој услов.
4. Програмата извршува подредување по големина на низата  $A(i)$  во растечки редослед со алгоритмот минимакс.

5.

```
Program ChudMak;
var
  si, so, s1, s2: string;
  i, j: integer;
begin
  write ('Makedonski: ');
  readln(si);

  i:=1;
  s1:='';

  while (i<=length(si)) and
    not(si[i] in ['a','e','i','o','u']) do
  begin
    s1:=s1+si[i];
    inc(i);
  end;
```

```

    for j:=i to length(si) do
        s2:=s2+si[j];
    so:=s2+s1+'au';
    write ('ChudMak:      ');
    writeln(so);
end.

```

**6.**

```

Program ProstiFaktori;
var  n,n0,d,i,j:integer;
begin
    write ('Vnesete n = '); readln(n);
    n0:=n; d:=2;
    while (d<=n0) do
        begin
            while (n0 mod d = 0) do
                begin
                    if d=2 then writeln('Prosti faktori na brojot ',n,'
se:');
                    n0:=n0 div d;
                    writeln(d);
                end;
                inc(d);
            end;
        end;
    end.

```

1.

```
Program Danok;
const prihod=200;
var podanici:array[1..6] of integer;
    paricki: array[1..6] of real;
    i,j,suma: integer;
begin
    for i:=1 to 6 do begin
        write ('Kolku podanici ima vitezot ',i,': ');
        readln(podanici[i]);
    end;
    paricki[1]:=0.03*podanici[1]*prihod;
    paricki[2]:=0.02*podanici[2]*prihod;
    paricki[3]:=0.05*podanici[3]*prihod;
    paricki[4]:=0.04*podanici[4]*prihod;
    paricki[5]:=0.5*podanici[5]*(podanici[5]+1);
    suma:=0;
    for i:=1 to 6 do
        suma:=suma+podanici[i];
    paricki[6]:=3*(suma);
    j:=1;
    for i:=2 to 6 do
        if paricki[i]>paricki[j]
        then j:=i;
    writeln ('Najmnogu paricki kje daje vitezot ',i,'.');
    writeln ('Toj kje daje ',paricki[i]:0:2,' paricki.');
```

2. a) T б) ⊥ в) ⊥ г) T д) T ё) T e) ⊥ ж) ⊥

3. a) ⊥ б) T в) ⊥ г) ⊥ д) ⊥

4.

```
Program ViteshkiIgri;
var bodovi: array[1..3,1..5] of integer;
    suma: array[1..3] of integer;
    i,j: integer;
begin
    for i:=1 to 5 do begin
        writeln ('Vnesovanje podatoci za igrata ',i);
        for j:=1 to 3 do begin
            write ('Kolku bodovi osvoil vitezot ',
                j,'? ');
            readln(bodovi[j,i]);
        end;
    end;
    for i:=1 to 3 do begin
        suma[i]:=0;
        for j:=1 to 5 do
```



```
        suma[i]:=suma[i]+bodovi[i,j];
end;
j:=1;
for i:=2 to 3 do
    if suma[i]>suma[j] then j:=i;
writeln('Najmnogu bodovi osvoil vitezot ',j);
writeln('Toj osvoil vkupno ',suma[j],' bodovi.');
```

```
end.
```

1. а) ⊥ б) Т в) ⊥ г) Т д) Т ё) ⊥ е) Т ж) ⊥ з) Т с) ⊥ и) Т

2.

```
Program Mechevi;
var
  n,i,s0,s:integer;
begin
  write('Kolku mechevi imalo vo prvata soba? ');
  readln(s0);
  write('Kolku vkupno sobi pominale princovite? ');
  readln(n);

  s:=n*(s0+n-1);
  writeln('Princovite videle vkupno ',s,' mechevi.');
```

end.

3.

```
Program Zmej;
var M, G, S: integer;
    VkupnoJazhe, VkupnoGlavi, Pomoshni: integer;
begin
  write ('Kolku golemi glavi? ');    readln (G);
  write ('Kolku sredni glavi? ');    readln (S);
  write ('Kolku mali glavi? ');      readln (M);

  VkupnoJazhe:=5*G + 3*S + 2*M;
  VkupnoGlavi:=G + M + S;

  Pomoshni:=((VkupnoGlavi-1) div 5)+1;

  writeln ('Potrebni se ',VkupnoJazhe,'m jazhe.');
```

writeln ('Potrebni se ',Pomoshni,' pomoshnici');

end.

1. а) В б) Н в) Н г) Н д) Н ё) Н е) В ж) В з) Н и) В

2.

```
Program Rotacija;
uses Crt;
var
  a: packed array[1..10] of char;
  i, j: integer;
  t: char;
begin
  write('Vnesi 10 znaci od tastaturata >');
  for i:=1 to 10 do
  begin
    a[i]:=ReadKey;
    write(a[i]);
  end;
  writeln('<');
  for i:=1 to 3 do
  begin
    t:=a[10];
    for j:=9 downto 1 do
      a[j+1]:=a[j];
    a[1]:=t;
  end;
  writeln('Po 3 rotacii e dobiena nizata: ');
  writeln('>', a, '<');
end.
```

3.

```
Program MaksOdTri;
var a, b, c, d, e, f, m1, m2, max: integer;

function maxodtri(a, b, c: integer): integer;
var m: integer;
begin
  m:=a;
  if m<b then m:=b;
  if m<c then m:=c;
  maxodtri:=m;
end;

begin
  writeln('Vnesete 6 broevi oddeleni so prazni karakteri');
  readln(a, b, c, d, e, f);
  m1:=maxodtri(a, b, c);
  m2:=maxodtri(d, e, f);
  max:=maxodtri(m1, m2, m2);
  writeln('Maksimumot e ', max);
end.
```

1. а) Н б) В в) Н г) В д) В ё) В е) В ж) В

2.

```
program SumaN;
var N,i: integer;
    A: array [1..100] of real;
    Suma: real;
begin
    write('Vnesete ja goleminata na poletoto: ');
    readln(N);
    for i:=1 to N do
    begin
        write ('Elementot A[' ,i, ' ] := ');
        readln(A[i]);
    end;
    Suma:=0;
    for i:=1 to N do Suma:=Suma+A[i];
    writeln('Sumata na elementite od poletoto e: ', Suma:0:3);
end.
```

3.

```
program ObratenRed;
var N: integer;
    i,j: integer;
begin
    write('Vnesi eden cel pozitiven broj: ');
    readln(N);
    i:=N;
    j:=0;
    while i>0 do
    begin
        j:=j*10+(i mod 10);
        i:=i div 10;
    end;

    writeln('Brojot so obraten redosled cifri e: ', j);
end.
```

4.

```
program NajmalDelitel;
var N,i,j: integer;
    najden: boolean;
begin
    write('Vnesi eden cel pozitiven broj: '); readln(N);
    i:=2; najden:=false;
    repeat
        if n mod i=0
        then najden:=true
        else inc(i);
    until najden or (i>n div 2);
```

```
if najden
then writeln('Najmaliot delitel pogolem od 1 e ',i)
else writeln('Brojot ima samo dva deliteli',
             'brojot 1 i samiot broj ', N);
end.
```

1. а) В б) Н в) В г) В д) Н

2. а) Бит е една бинарна цифра.

б) 127

в) Оперативен систем, програми за преведување, услужни програми

г) Централна меморија, аритметичко-логичка единица, управувачка единица

д) Најважна разлика е можноста за запишување/измена на содржината во RAM-от; можност која ја нема кај ROM-от.

3.

```
program EkvivalentniIskazi;
var
  TetkaZiveeVoSkopje,
  BabaZiveeVoSkopje: boolean;
  X, Y: boolean;
  SeEkvivalentni: boolean;

begin
  SeEkvivalentni:=true;
  for TetkaZiveeVoSkopje:=TRUE downto FALSE do
    for BabaZiveeVoSkopje:=TRUE downto FALSE do
      begin
        X:=not(TetkaZiveeVoSkopje or BabaZiveeVoSkopje);
        Y:=not(TetkaZiveeVoSkopje) or not(BabaZiveeVoSkopje);
        if X<>Y
          then SeEkvivalentni:=false;
        end;

        if SeEkvivalentni
          then writeln('Izrazite X i Y se ekvivalentni')
          else writeln('Izrazite X i Y ne se ekvivalentni');
      end.
end.
```

4.

```
program OrigObra;
var orig,obra:array [1..5] of integer;
    i:integer;
begin
  for i:=1 to 5 do
    begin
      write ('Vnesete go ',i,'-tiot element: ');
      readln (orig[i]);
    end;
    for i:=1 to 5 do obra[5-i+1]:=orig[i];
    for i:=1 to 5 do writeln('OBRA[' ,i, ']=' ,obra[i]);
  end.
end.
```

1. а) × б) ✓ в) ✓ г) ✓ д) × е) ✓ ф) ✓ з) ×

2.

**алгоритам** ПодредувањеНаТриБроја;

**почеток**

**печати** 'Внесете три броја: ';

**читај** a, b, c;

**ако** a > b

**тогаш**

a ↔ b;

**крај\_ако** {a > b}

**ако** a > c

**тогаш**

a ↔ c;

**крај\_ако** {a > c}

**ако** b > c

**тогаш**

b ↔ c;

**крај\_ако** {b > c}

**печати** a, b, c;

**крај** {ПодредувањеНаТриБроја}

```
program PodreduvanjeNaTriBroja;
```

```
var A, B, C, Pom: integer;
```

```
begin
```

```
    Write('Vnesete tri broja: ');
```

```
    ReadLn(A, B, C);
```

```
    if A > B
```

```
    then
```

```
        begin
```

```
            Pom := A;
```

```
            A := B;
```

```
            B := Pom;
```

```
        end;
```

```
    if A > C
```

```
    then
```

```
        begin
```

```
            Pom := A;
```

```
            A := C;
```

```
            C := Pom;
```

```
        end;
```

```
    if B > C
```

```
    then
```

```
        begin
```

```
            Pom := B;
```

```
            B := C;
```

```
            C := Pom;
```

```

        end;
        WriteLn(A, ' ', B, ' ', C);
end. {PodreduvanjeNaTriBroja}

```

3.

**алгоритам** *ЗбирОдПодзбирови*;

**почеток**

**печати** 'Внеси до кој природен број n=' ;

**читај** n;

zbir←0;

brojac←1;

podzbir←1;

**повторувај**

zbir←zbir+podzbir;

brojac←brojac+1;

podzbir←podzbir+brojac;

**до** brojac>n;

**крај\_повторувај**

**печати** 'Збирот изнесува',zbir;

**крај**{*ЗбирОдПодзбирови*}

```

program ZbirOdPodzbirovi;

```

```

var N,Zbir,Podzbir,Brojac:integer;

```

```

begin

```

```

    Write('Vnesi do koj prirodan broj N=');

```

```

    ReadLn(N);

```

```

        Zbir:=0;

```

```

        Brojac:=1;

```

```

        Podzbir:=1;

```

```

        repeat

```

```

            Zbir:=Zbir+Podzbir;

```

```

            Brojac:=Brojac+1;

```

```

            Podzbir:=Podzbir+Brojac;

```

```

        until Brojac>N;

```

```

        WriteLn('Zbirot iznesuva ',Zbir);

```

```

end. {ZbirOdPodzbirovi}

```

4.

**подалгоритам** *ДецималенВоБинарен*(decimalen, a, j) ;

**почеток**

j←0;

**повторувај**

j←j-1;

decimalen←2·decimalen;

a<sub>j</sub>←[decimalen];

decimalen←decimalen-a<sub>j</sub> ;

**до** j=-6;

**крај\_повторувај**

**крај** {*ДецималенВоБинарен*}



```

procedure DecimalenVoBinaren(Decimalen:real;VAR A:niza;VAR
J:byte);
begin
    J:=0;
    repeat
        Dec(J);
        Decimalen:=2*Decimalen;
        A[J]:=Trunc(Decimalen);
        Decimalen:=Decimalen-A[J];
    until J=-6;
end;{DecimalenVoBinaren}

program PretvoranjeDecimalenDekadenBrojVoBinaren;
uses Crt;
type niza=array[1..100] of 0..1;
var    I,J,K:byte;
        DecimalenBroj:real;
        A:niza;
begin
    ClrScr;
    Write('Vnesete decimalen dekadен број: ');
    ReadLn(DecimalenBroj);
    DecimalenVoBinaren(DecimalenBroj,A,J);
    for K:=-1 downto J DO
        Write(A[K]);
    repeat
    until KeyPressed;
end.{PretvoranjeDecimalenDekadenBrojVoBinaren}

```

---

## Група 2

1. а) × б) × в) ✓ г) × д) ✓ е) ×

2.

```

алгоритам СпротивенБрој;
почеток
    печати 'Внеси природен број помал од 1000, n= ';
    читај n;
    ако n<10
    тогаш
        sprotiven←n
    инаку
        ако n<100
        тогаш
            sprotiven←10·(n мод 10)+ $\frac{n}{10}$ 
        инаку
            sprotiven←100·(n мод 10)+10·( $\frac{n}{10}$  мод 10)+ $\frac{n}{100}$ ;

```

```

        крај_ако{ n<100}
        крај_ако{ n<10}
        печати 'Спротивен број на бројот ',n,' е бројот
',sprotiven;
крај{СпротивенБрој}

program SprotivenBroj;
var N,Sprotiven:integer;
begin
    Write('Vnesi prirodan broj pomal od 1000, N= ');
    ReadLn(N);
    if N<10
then
    Sprotiven:=N
else
    if N<100
then
        Sprotiven:=10*(N MOD 10)+(N DIV 10)
    else
        Sprotiven:=100*(N MOD 10)+
        10*((N DIV 10) MOD 10)+(N DIV 100);
    WriteLn('Sprotiven broj na brojot ',N,' e brojot ',
Sprotiven, '.');
end.{SprotivenBroj}

```

### 3.

```

алгоритам ЗбирОдПодзбирови;
почеток
    печати 'Внеси до кој природен број n=';
    читај n;
    zbir←0;
    brojac←1;
    podzbir←0;
    повторувај
        podzbir←podzbir+brojac;
        brojac←brojac+1;
    до brojac>n;
    крај_повторувај
    zbir←0;
    brojac←1;
    повторувај
        zbir←zbir+podzbir;
        podzbir←podzbir-brojac;
        brojac←brojac+1;
    до brojac>n;
    крај_повторувај
    печати 'Збирот изнесува', zbir;
крај{ЗбирОдПодзбирови}

```

```

program ZbirOdPodzbirovi;
var N, Zbir, Podzbir, Brojac:integer;
begin
    Write('Vnesi do koj prirodan broj N=');
    ReadLn(N);
    Podzbir:=0;
    Brojac:=1;
    repeat
        Podzbir:=Podzbir+Brojac;
        Brojac:=Brojac+1;
    until Brojac>N;
    Zbir:=0;
    Brojac:=1;
    repeat
        Zbir:=Zbir+Podzbir;
        Podzbir:=Podzbir-Brojac;
        Brojac:=Brojac+1;
    until Brojac>N;
    WriteLn('Zbirot iznesuva ', Zbir);
end. {ZbirOdPodzbirovi}

```

4.

```

подалгоритам ЦелВоБинарен(сел, а, i);
почеток
    i←-1;
повторувај
    i←i+1;
    ai←сел mod 2;
    сел← $\lfloor \frac{сел}{2} \rfloor$ ;
до сел=0;
крај_повторувај
крај {ЦелВоБинарен}
procedure CelVoBinaren(Cel:longint;VAR A:niza; VAR I:byte);
begin
    I:=-1;
    repeat
        Inc(I);
        A[I]:=Cel MOD 2;
        Cel:=Cel DIV 2;
    until Cel=0;
end; {CelVoBinaren}

program PretvoranjeNaCelDekadenBrojVoBinaren;
uses Crt;
type niza=ARRAY[1..100] of 0..1;
var    I, J, K:byte;
        CelBroj:longint;
        A:niza;
begin
    ClrScr;

```

```
Write('Vnesete pozitiven realen dekaden broj: ');
ReadLn(CelBroj);
Write(CelBroj:12:4,' = ');
CelVoBinaren(CelBroj,A,I);
for K:=I downto 0 do
    Write(A[K]);
repeat
until KeyPressed;
end.{PretvoranjeNaCelDekadenBrojVoBinaren}
```

1.

а) ✓ б) ✓ в) × г) ✓ д) × е) × ж) ×

2.

а) ✓ б) ✓ в) ✓ г) × д) ✓ е) ✓ ж) ×

3.

```
алгоритам ПаренИлиНепарен;  
почеток  
    печати 'Внеси природен број n=';  
    читај n;  
    ако  $n - 2 \cdot \left\lfloor \frac{n}{2} \right\rfloor = 0$   
        тогаш  
            печати 'Бројот ', n, ' е парен'  
        инаку  
            печати 'Бројот', n, ' е непарен';  
крај_ако {n mod 2=0}  
крај {ПаренИлиНепарен}
```

4.

```
алгоритам АритметичкаСредина;  
почеток  
    печати 'Внеси еден природен број n=';  
    читај n;  
    zbir ← 0;  
    brojac ← 1;  
    додека brojac ≤ n извршувај  
        zbir ← zbir + brojac;  
        brojac ← brojac + 1;  
    крај_додека {brojac ≤ n}  
    arisredina ←  $\frac{zbir}{n}$ ;  
    печати arisredina;  
крај {АритметичкаСредина}
```

1.

a) ✓ б) ✓ в) ✓ г) × д) ✓ ё) × е) ×

2.

НЗД на двата внесени броја.

3.

```
program bukva_vo_zbor;

var
  s : string;
  b, i : integer;

begin
  s := 'ABRAKADABRA';
  b := 0;
  for i := 1 to length (s) do
  begin
    if s[i] = 'A' then b := b + 1;
  end;
  writeln ('Vo zborot ABRAKADABRA bukvata A se naoga ', b, '
pati.');
```

4.

```
program SovrseniBroevi;

var
  i, j, zbir : integer;

begin
  writeln ('Sovrseni broevi se:');
  for i := 1 to 1000 do
  begin
    zbir := 0;
    for j := 1 to i-1 do
    begin
      if i mod j = 0 then zbir := zbir + j;
    end;
    if zbir = i then writeln (i);
  end;
end.
```

1.

a) ✓ б) ✓ в) × г) ✓ д) ✓ ё) × е) ✓

2.

s е збирот на природните броеви од 1 до m.

p е производот на природните броеви од 1 до m.

3.

bcif е бројот на цифри.

4.

```
program Parni_vo_niza;

var
  i, n : integer;
  a : array [1..100] of integer;

begin
  writeln ('Kolku elementi ima nizata:');
  readln (n);
  writeln ('Vnesete ja nizata:');
  for i := 1 to n do
  begin
    write ('a[' , i , ']=');
    readln (a[i]);
  end;
  writeln ('Parnite elementi vo nizata se:');
  for i := 1 to n do
  begin
    if a[i] mod 2 = 0 then writeln (a[i]);
  end;
end.
```

## 30.06.2003

---

1. a) × б) × в) ✓ г) ✓ д) × е) ✓

2. Ja печати најголемата вредност од три внесени вредности.

3.

```
program SumaIProizvodN;
var N,i: integer;
    A: array [1..100] of real;
    Suma, Proizvod: real;
begin
    write('Vnesete ja goleminata na nizata: ');
    readln(N);
    for i:=1 to N do
    begin
        write ('Elementot A[' ,i, ' ] := ');
        readln(A[i]);
    end;
    Suma:=0;
    Proizvod:=1;
    for i:=1 to N do
    begin
        Suma:=Suma+A[i];
        Proizvod:=Proizvod*A[i];
    end;
    writeln('Sumata na elementite od nizata e: ', Suma:0:3);
    writeln('Proizvodot na elementite od nizata e: ',
Proizvod:0:3);
end.
```

4.

```
program kratenje;
var
    a,b,cut:integer;
begin
    writeln('Vnesi go broitelot');
    readln(a);
    writeln('Vnesi go imenitelot');
    readln(b);

    cut:=2;
    repeat
    while (a mod cut=0) and (b mod cut=0) do
    begin
        a:=a div cut;
        b:=b div cut;
    end;
    inc(cut);
    until cut>a;
    writeln(a, '/', b);
end.
```



**05.09.2003**

1. а) број, текст, слика, звук      б) ×      в) ласерски, линиски, матрични  
г) Windows      д) ✓      ф) ×      е) ✓      ж) ×
2. а) KB, MB, GB      б) Pascal, Basic, C      в) ×      г) ×      д) ×  
ѓ) ако-тогаш-инаку, повторувај-до      ж) True, False

3.

```
program GeometriskaSredina;
var N,i: integer;
    A: array [1..100] of real;
    Proizvod, Geom: real;
begin
    write('Vnesete ja goleminata na nizata: ');
    readln(N);
    for i:=1 to N do
    begin
        write ('Elementot A[' ,i, ' ] := ');
        readln(A[i]);
    end;
    Proizvod:=1;
    for i:=1 to N do
    begin
        Proizvod:=Proizvod*A[i];
    end;
    Geom:= Exp (1/N*Ln (Proizvod))
    writeln('Sumata na elementite od nizata e: ', Suma:0:3);
end.
```

4.

```
алгоритам ПодредувањеНаТриБроја;
почеток
    печати 'Внесете три броја: ';
    читај a,b,c;
    ако a>b
        тогаш
            a↔b;
    крај_ако{a>b}
    ако a>c
        тогаш
            a↔c;
    крај_ако{a>c}
    ако b>c
        тогаш
            b↔c;
    крај_ако{b>c}
    печати a,b,c;
крај{ПодредувањеНаТриБроја}
```

## 09.2004

1. а) ✓×    б) ✓    в) ×    г) ×    д) ×    е) ✓    ж) ×

2.

**алгоритам** ПодредувањеНаТриБроја;

**почеток**

**печати** 'Внесете три броја: ';

**читај** a,b,c;

**ако** a>b

**тогаш**

a↔b;

**крај\_ако**{a>b}

**ако** a>c

**тогаш**

a↔c;

**крај\_ако**{a>c}

**ако** b>c

**тогаш**

b↔c;

**крај\_ако**{b>c}

**печати** a,b,c;

**крај**{ПодредувањеНаТриБроја}

```
program PodreduvanjeNaTriBroja;
```

```
var A,B,C,Pom:integer;
```

```
begin
```

```
    Write('Vnesete tri broja: ');
```

```
    ReadLn(A,B,C);
```

```
    if A>B
```

```
    then
```

```
        begin
```

```
            Pom:=A;
```

```
            A:=B;
```

```
            B:=Pom;
```

```
        end;
```

```
    if A>C
```

```
    then
```

```
        begin
```

```
            Pom:=A;
```

```
            A:=C;
```

```
            C:=Pom;
```

```
        end;
```

```
    if B>C
```

```
    then
```

```
        begin
```

```
            Pom:=B;
```

```
            B:=C;
```

```
            C:=Pom;
```

```
        end;
```

```
WriteLn(A, ' ', B, ' ', C);
end. {PodreduvanjeNaTriBroja}
```

3.

```
program PitagoriniBroevido 100;
var a, b, c:integer;
begin
  for TetkaZiveeVoSkopje:=TRUE downto FALSE do
    for BabaZiveeVoSkopje:=TRUE downto FALSE do
      begin
        Write('Pitagorini trojki se:');
        for a:=1 to trunc(100/sqrt(2))do
          for b:=1 to 100 do
            begin
              c:=trunc(sqrt(a*a+b*b));
              if (c<=100) and (c=a*a+b*b)
                then write (a,',',b,',',c,' ');
              {end_if_(c<=100) and (c=a*a+b*b)}
            end;
          {end_if_b}
        {end_if_a}
      end. {PitagoriniBroevido 100}.
```

4.

```
алгоритам ЗбирОдПодзбирови;
почеток
  печати 'Внеси до кој природен број n=';
  читај n;
  zbir←0;
  brojac←1;
  podzbir←0;
  повторувај
    podzbir←podzbir+brojac;
    brojac←brojac+1;
  до brojac>n;
  крај_повторувај
  zbir←0;
  brojac←1;
  повторувај
    zbir←zbir+podzbir;
    podzbir←podzbir-brojac;
    brojac←brojac+1;
  до brojac>n;
  крај_повторувај
  печати 'Збирот изнесува', zbir;
крај {ЗбирОдПодзбирови}

program ZbirOdPodzbirovi;
var N, Zbir, Podzbir, Brojac:integer;
begin
```

```
Write('Vnesi do koj prirodan broj N=');
ReadLn(N);
Podzbir:=0;
Brojac:=1;
repeat
    Podzbir:=Podzbir+Brojac;
    Brojac:=Brojac+1;
until Brojac>N;
Zbir:=0;
Brojac:=1;
repeat
    Zbir:=Zbir+Podzbir;
    Podzbir:=Podzbir-Brojac;
    Brojac:=Brojac+1;
until Brojac>N;
WriteLn('Zbirot iznesuva ',Zbir);
end. {ZbirOdPodzbirovi}
```