

1989-2014

M

ƏKTƏBLİLƏRİN  
İNFORMATİKA  
OLİMPİADALARI

Ramin MAHMUDZADƏ

İsmayıl CALALLI

Akif ƏLİYEV

B A K I



N Ə Ş R

2015

- 2015-ci il sentyabrın 1-də “İnformatika və hesablama texnikasının əsasları” kursunun ümumtəhsil məktəblərində tədrisinə başlanmasının **30 ili** tamam olacaq...
- 2015-ci ilin martında **25-ci** Respublika İnformatika Olimpiadası keçiriləcək...
- İyul ayında isə məktəblilərdən ibarət komandamız **22-ci** dəfə Beynəlxalq İnformatika Olimpiadasına qatılacaq...
- Ancaq “Beynəlxalq İnformatika Olimpiadaları” (2012) kitabı istisna olmaqla informatika olimpiadalarına aid vəsaitlər Azərbaycan dilində, demək olar ki, yoxdur. Bu kitabın yazılmasında da başlıca məqsəd həmin çatışmazlığı aradan qaldırmaq, bu sahə ilə maraqlananlara kömək etməkdir.
- Təqdim edilən kitab informatika müəllimləri, istedadlı uşaqlarla işləyən pedaqoqlar üçün nəzərdə tutulmuşdur. Kitabdakı materiallar informatika üzrə olimpiadalara hazırlaşmaq istəyən şagirdlər, tələbələr, eləcə də proqramlaşdırmaya marağı olanlar üçün də faydalı ola bilər.

Mahmudzadə R.

Calallı İ.

Əliyev A.

### **Məktəblilərin İnformatika Olimpiadaları**

“Bakınəşr”, 2015. 256 səh.

© Azərbaycan Respublikasının Təhsil Nazirliyi, 2015

© “Bakınəşr”, “TMGroup”, 2015

Çapa imzalanıb: 06.01.2015.

Formatı: 70×100<sup>1</sup>/<sub>16</sub>.

Fiziki ç.v.: 16.

Tirajı: 2000.

# ÖN SÖZ

2015-ci il sentyabrın 1-də “İnformatika və hesablama texnikasının əsasları” kursunun ümumtəhsil məktəblərində tədrisinə başlanmasının **30 ili** tamam olacaq... 2015-ci ilin martında **25-ci** Respublika İnformatika Olimpiadası keçiriləcək... İyul ayında isə məktəblilərdən ibarət komandamız **22-ci** dəfə Beynəlxalq İnformatika Olimpiadasına qatılacaq...

Ötən dövr ərzində istər ümumtəhsil məktəblərinin bütün sinifləri (1-11) üçün “İnformatika” dərsləklərinin (xüsusən milli kurikulum əsasında) hazırlanması, istər bu fənnin tədrisi, istər normativ-hüquqi sənədlərin təsdiq edilməsi, istərsə də müxtəlif yarışların, o cümlədən olimpiadaların keçirilməsi istiqamətində xeyli işlər görülmüşdür.

Azərbaycan Respublikası Prezidentinin 24 oktyabr 2013-cü il tarixli sərəncamı ilə təsdiq edilmiş “Azərbaycan Respublikasında təhsilin inkişafı üzrə Dövlət Strategiyası”nda prioritet istiqamətlərdən biri olaraq təhsilənlərin istedadının aşkar olunması və inkişafı ilə bağlı məsələlər yer tutur. Eləcə də Strategiya ilə bağlı strateji hədəflər və tədbirlərdə istedadlı uşaqlar üçün xüsusi inkişaf proqramlarının hazırlanması nəzərdə tutulur.

Yeri gəlmişkən, Azərbaycan Respublikası Prezidentinin 17 aprel 2006-cı il tarixli, 1414 nömrəli sərəncamı ilə təsdiq edilmiş “Xüsusi istedadlı malik uşaqların (gənclərin) yaradıcılıq potensialının inkişafı üzrə Dövlət Proqramı”nda da ayrı-ayrı fənlər üzrə keçirilən olimpiadaların istedadlı şagirdlərin seçilməsində əhəmiyyətinə xüsusi diqqət yetirilmişdir.

Azərbaycan Respublikası Təhsil nazirinin 04.02.2008-ci il tarixli, 171 nömrəli əmrinə əsasən, beynəlxalq fənn olimpiadalarında iştirak üçün seçilmiş istedadlı şagirdlərin hazırlıq səviyyəsinin yüksəldilməsinə daha əlverişli şərait yaratmaq məqsədilə onlar hər ilin noyabr və aprel aylarında dərslər məşğələlərindən azad olunurlar. Azərbaycan Respublikası Təhsil nazirinin 11 fevral 2009-cu il tarixli, 164 nömrəli əmri ilə təsdiq edilmiş “Ümumtəhsil məktəblərində, peşə liseylərində buraxılış imtahanları və şagirdlərin sinifdən-sinfə keçirilməsi haqqında

qaydalar”ın 75-ci bəndinə əsasən, illik qiymətləri müsbət olduğu halda, fənn olimpiadalarının respublika turunda birinci yeri tutmuş şagirdlər, eləcə də imtahanlar dövründə məktəblilərin dünya fənn olimpiadalarında və digər beynəlxalq səviyyəli idman və bilik yarışlarında iştirak edən şagirdlər Təhsil nazirinin əmri ilə buraxılış imtahanlarından azad edirlər. Tam orta təhsil kursunu bitirən şagirdlərə qızıl və gümüş medal verilərəkən də onların respublika və beynəlxalq olimpiadalarda iştirakı nəzərə alınır. Eləcə də 2010-cu ildən beynəlxalq olimpiadaların qalibləri ali məktəblərin müvafiq ixtisaslarına imtahansız qəbul olunurlar.

Artıq respublikamızda informatika üzrə olimpiadalarla bağlı müsbət təcrübə qazanılmış və bu iş davam etdirilir. Müxtəlif fənlər üzrə olimpiadalarla maraqlananlara kömək edəcək metodik vəsaitlər az da olsa var. Ancaq informatika üzrə belə vəsaitlər Azərbaycan dilində, demək olar ki, yoxdur (2012-ci ildə nəşr olunmuş “Beynəlxalq İnformatika Olimpiadaları” kitabı istisna olmaqla). Bu kitabın yazılmasında da başlıca məqsəd həmin çatışmazlığı aradan qaldırmaq, bu sahə ilə maraqlananlara kömək etməkdir. Təqdim edilən kitab informatika müəllimləri, istedadlı uşaqlarla işləyən pedaqoqlar üçün nəzərdə tutulmuşdur. Kitabdakı materiallar informatika üzrə olimpiadalara hazırlaşmaq istəyən şagirdlər, tələbələr, eləcə də proqramlaşdırmaya marağı olanlar üçün də faydalı ola bilər.

Əlbəttə, yaxşı insanların dəstəyi və yardımı olmasa idi, belə bir kitabı üzə çıxarmaq çox çətin olardı. İlk növbədə, kitabın işıq üzü görməsində göstərdikləri dəstəyə görə Təhsil Nazirliyinin Məktəbəqədər və ümumi təhsil şöbəsinin müdiri Aydın Əhmədova, Təhsilin inkişafı proqramları şöbəsinin müdiri Emin Əmrullayeva, “Azərbaycan müəllimi” qəzetinin arxivindən materialların əldə edilməsində göstərdiyi köməyə görə baş redaktor Bayram Hüseynzadəyə və qəzetin əməkdaşlarına öz dərin minnətdarlığımızı bildiririk. Eləcə də Təhsil Nazirliyinin arxivindən həmin illərə aid sənədlərin tapılmasında çəkdiyi zəhmətə görə Fəridə Vəlimətovaya, şəxsi arxivindən təqdim etdiyi sənədlərə görə Fizika, riyaziyyat və informatika təmayüllü liseyin informatika müəllimi Valid Məhərrəmovaya, kitabın üzə çıxmasında az da olsa əməyi olan hər kəsə minnətdarıq.

Bu kitabın Respublika İnformatika Olimpiadaları haqqında ilk təcrübə olduğunu, ilk olimpiadaların ölkəmiz üçün çox çətin və sabit olmayan dövrə təsadüf etdiyini, həmin dövrdən yetərinə zaman ötdüyünü nəzərə alsaq, kitabda müəyyən çatışmazlıqların və qeyri-dəqiqliyin olması qaçılmazdır. Ona görə də kitab haqqında irad və təkliflərini müəlliflərə, yaxud “Bakı” nəşriyyatına ([bn@bakineshr.az](mailto:bn@bakineshr.az) ünvanına) göndərəcək hər kəsə qabaqcadan minnətdarıq.



# MƏKTƏBLİLƏR ARASINDA İNFORMATİKA OLİMPİADALARININ TARİXİ

## 1.1. İNFORMATİKA ÜMUMTƏHSİL MƏKTƏBLƏRİNDƏ

1985-ci ilin 1 sentyabr tarixindən keçmiş SSRİ-nin, o cümlədən Azərbaycan SSR-in ümumtəhsil məktəblərində yeni fənnin – **“İnformatika və hesablama texnikasının əsasları”** («Основы информатики и вычислительной техники») fənninin tədrisinə başlandı. Fənnin ümumtəhsil məktəblərinin yalnız yuxarı siniflərində (IX və X siniflərdə) tədrisi nəzərdə tutulurdu. Əlbəttə, məktəblərə yeni fənnin gətirilməsi birdən-birə baş vermədi, bu yenilik bir müddət müəyyən hazırlıq işləri aparıldıqdan sonra mümkün oldu. Belə ki, 1960-cı illərin başlanğıcında şagirdlərə kibernetikanın elementlərinin öyrədilməsi istiqamətində eksperimentlər aparılırdı. Bu eksperimentlərin nəticəsi olaraq 1970-ci ildə “Kibernetikanın əsasları” kursu orta ümumtəhsil məktəblərinin fakültativ kursları siyahısına rəsmən daxil edildi. 140 saatlıq (70 saat IX, 70 saat X sinifdə) bu kurs 1985-ci ilədək əsasən fizika-riyaziyyat təmayüllü məktəblərdə tədris olundu.

**12 aprel 1984-cü il.** Məhz bu tarixdə keçmiş SSRİ-də, o cümlədən Azərbaycanda ümumtəhsil məktəblərində informatika və hesablama texnikasının əsaslarının öyrənilməsi dövrü başlandı. SSRİ Nazirlər Soveti sədrinin 1-ci müavini **Heydər Əliyevin** sədrliyi ilə keçirilmiş iclasda tarixi qərar qəbul edildi. Sov.İKP MK-nın (Sovet İttifaqı Kommunist Partiyası Mərkəzi Komitəsi) və SSRİ Nazirlər Sovetinin 1984-cü il 12 aprel tarixli 313 nömrəli qərarı ilə SSRİ Maarif Nazirliyinə, Pedaqoji Elmlər Akademiyasına, Dövlət Texniki Peşə Təhsili Komitəsinə, Ali və Orta ixtisas Təhsili Nazirliyinə tapşırılırdı:

- ümumtəhsil məktəblərinin yuxarı siniflərində, texniki peşə məktəblərində, orta ixtisas məktəblərində hesablayıcı elektron texnikasının əsaslarının öyrənilməsini təşkil etsinlər, belə ki, şagirdlərə kompüterlərdən istifadə

etmək vərdişləri aşılsın və onlar xalq təsərrüfatında bu texnikanın geniş tətbiqi haqqında biliklərlə silahlandırılınsınlar. Bundan ötrü şagirdlər üçün xüsusi kurslar hazırlansın, lazımi dərsliklər, dərs vəsaitləri, məktəb kabinetləri və məktəblərarası kabinetlər yaradılsın, habelə baza müəssisələrinin və başqa idarələrin kompüter texnikasından dərs məqsədləri ilə istifadə olunması nəzərdə tutulsun;

- ümumtəhsil məktəblərinin dərs prosesində kompüterlərin tətbiqi ilə bağlı psixoloji-pedaqoji problemlər haqqında 1986-cı ildə Sov.İKP MK-ya və SSRİ Nazirlər Sovetinə məlumat verilsin;
- 1986-1990-cı illərdə hesablayıcı elektron və mikroprosessor texnikası kabinetləri yaradılsın.

1984-cü ildə hazırlanan “Ümumtəhsil və peşə məktəblərində islahatın əsas istiqamətləri” adlı təşkilati-metodik sənəddə məktəb islahatının əsas istiqamətlərindən biri kimi gənclərin ümumi kompüter savadsızlığının aradan qaldırılmasının təmini, informatika və hesablama texnikasının əsaslarının tədris prosesinə daxil edilməsi elan olundu.

1984-cü ilin sonunda A.P.Yeršov və V.M.Monaxovun rəhbərliyi ilə “İnformatika və hesablama texnikasının əsasları” fənninin proqramı hazırlanmağa başlandı və 1985-ci ilin ortalarında həmin proqram SSRİ Maarif Nazirliyi tərəfindən bəyənilədi. “İnformatika və hesablama texnikasının əsasları” bir tədris fənni kimi ümumtəhsil məktəblərinin tədris planına 1985-ci ilin 1 sentyabrından daxil edildi.

1986-cı ildə “İnformatikanın və hesablama texnikasının əsasları” kursu üzrə ilk proqramın “maşınli variantı” çap edildi. İki yuxarı sinifdə tədrisi nəzərdə tutulan kursa 102 saat vaxt ayrıldı. Maşınli variantda nəzərdə tutulan məzmunu uyğun olaraq bir neçə dərs vəsaiti hazırlandı.

A.P.Yeršov və V.M.Monaxovun rəhbərliyi ilə hazırlanmış dərsliklər və müəllim üçün vəsaitlər 1985-1987-ci illərdə Azərbaycan dilində nəşr edildi.

1987-ci ilin sonunda işə qərara alındı ki, 1988-ci ilin yazında İnformatika üzrə ilk Ümumittifaq olimpiadası keçirilsin.

## 1.2. RESPUBLİKA İNFORMATİKA OLİMPİADALARI

Riyaziyyat, fizika, kimya kimi fənlərdən fərqli olaraq informatika üzrə olimpiada hərəkətinin tarixi nisbətən yenidir. Bu da “İnformatika və hesablama texnikasının əsasları” fənninin ümumtəhsil məktəbləri proqramına adları yuxarıda sadalanan fənlərlə müqayisədə nisbətən yeni daxil edilməsi ilə bağlıdır.

Qeyd olunduğu kimi, “İnformatika və hesablama texnikasının əsasları” fənninin ümumtəhsil məktəbləri proqramına daxil edilməsindən (1985-1986 tədris ilindən) iki il sonra – 1988-ci il aprelin 13-dən 20-dək 1-ci Ümumittifaq İnformatika Olimpiadası keçirildi. Azərbaycan məktəbliləri bu yarışmada iştirak etmirdi, çünki o zaman məktəblilər arasında respublikamızda keçirilən fənn olimpiadaları siyahısına informatika hələ daxil edilməmişdi. Bu hadisə bir qədər sonra baş verdi. Belə ki, 1988-1989 tədris ilində məktəblilərin respublika fənn olimpiadalarının müştəkkil keçirilməsi məqsədilə Azərbaycan SSR Xalq Təhsili Nazirliyində 22 dekabr 1988-ci il tarixli, 329 nömrəli əmr imzalandı. Əmrə müvafiq olaraq cari tədris ilində olimpiadaları beş fənn – rus dili, fizika, kimya, riyaziyyat və informatika üzrə üç mərhələdə keçirmək nəzərdə tutulurdu.

**1989-cu ilin 25-31 mart tarixində 1-ci Respublika İnformatika Olimpiadası keçirildi.** Yarışma iki turdan ibarət idi: birinci tur nəzəri, ikinci tur isə praktik. Çox az (cəmi 5) məktəblinin iştirak etdiyi ilk olimpiadanın qalibi Bakı şəhəri 23 saylı məktəbin 9-cu sinif şagirdi **Musa Məhərrəmov** oldu. Maraqlıdır ki, Musa nəzəri turda iştirak edə bilməmişdi, çünki o, həmin vaxt keçirilən riyaziyyat olimpiadasında da iştirak edirdi. Ona görə də yalnız praktik tura imkan taparaq bütün məsələlərin öhdəsindən layiqincə gəlmişdi. Bundan az sonra – aprelin 15-dən 22-dək Belarusiyanın Minsk şəhərində keçirilən 2-ci Ümumittifaq İnformatika Olimpiadasında da respublikamızı Musa Məhərrəmov təmsil etdi. 1989-cu ildə ümumittifaq olimpiadalarına bütün fənlər üzrə göndərilən 18 Azərbaycan məktəblisindən yalnız 1 nəfər qalib olmuşdu ki, bu da informatika fənni üzrə Musa Məhərrəmov idi (III yer).

Bu dövrdə daha bir mühüm hadisə baş verdi. Azərbaycan SSR Xalq Təhsili Nazirliyinin 12 iyul 1989-cu il tarixli, 819 nömrəli əmrinə əsasən, məktəblilərin respublika fənn olimpiadalarının qaliblərinin ali məktəblərə imtahansız qəbul edilməsi haqqında qərar qəbul olundu. Elə həmin il respublikamızda ilk dəfə müxtəlif fənlər üzrə olimpiadaların yekun turunun I dərəcəli diplomuna layiq görülmüş 6 nəfər X sinif məzunu müvafiq sahələr üzrə ali məktəblərə imtahansız qəbul edildi və bu da sonrakı illərdə olimpiada qaliblərinin stimullaşdırılmasına müsbət təsir göstərdi.

**1989-1990-cı** dərs ilində əvvəlki illərdən fərqli olaraq məktəblilərin fənn olimpiadalarının təşkili və keçirilməsinə xüsusi diqqət və əhəmiyyət verildi, bir sıra əməli tədbirlər görüldü. Azərbaycan SSR Xalq Təhsili Nazirliyinin 10 oktyabr 1989-cu il tarixli, 1100 nömrəli əmri ilə yenidən işlənmiş “Məktəblilərin fənn olimpiadaları haqqında Əsasnamə” təsdiq edildi, olimpiadaların coğrafiyası daha da genişləndirilərək olimpiada keçirilən fənlərin sayı 5-dən 9-a çatdırıldı, olimpiadaların məktəb və rayon (şəhər) turlarında şagirdlərin VI sinifdən başlayaraq iştirak etməsi nəzərdə tutuldu.

Ancaq həmin il regionda yaranmış gərgin vəziyyətlə (Dağlıq Qarabağ məsələsi, Bakıda 20 Yanvar faciəsindən sonrakı komendant saati) əlaqədar olaraq Respublika İnformatika Olimpiadasının yekun turunu nəzərdə tutulduğu müddətdə (məktəblilərin yaz tətili dövründə – 24-31 martda) və həm də Bakı şəhərində keçirmək mümkün olmadı və bu, olimpiada qaliblərinin Ümumittifaq Olimpiadalarında iştirak etmələrinə imkan vermədi. Buna baxmayaraq, həmin il respublika olimpiadalarının yekun turunu **1990-cı il aprelin 20-23-də Sumqayıt** şəhərində keçirmək mümkün oldu. Fənnin məktəb kursuna daxil edildiyi ilk illər olduğuna görə “İnformatika və hesablama texnikasının əsasları” fənni üzrə olimpiadalarda respublikanın 80 faiz rayon və şəhərindən bir nəfər belə şagird iştirak etməmiş, iştirakçıların sayı az olmuş və respublika turunda 1 nəfər qalib olmuşdur. Belə bir çətin dövrdə fənn olimpiadalarının yüksək səviyyədə təşkili və istedadlı uşaqların seçilməsinə böyük qayğı və gərgin əmək sərf etdikləri üçün fənlər üzrə münisflər heyətinin sədrləri, o cümlədən “İnformatika və hesablama texnikasının əsasları” fənni üzrə münisflər heyətinin sədri, Bakı Dövlət Universitetinin dosenti Ramin Mahmudzadə Təhsil Nazirliyinin fəxri fərmanı ilə təltif edilmişdi.

Ötən dövr ərzində respublikamızda informatika üzrə olimpiadaların miqyası arzuolunan səviyyədə olmasa da, ilbəl genişlənmişdir. İndi informasiya və kommunikasiya texnologiyalarına aid başqa müsabiqələr də həm şagirdlər arasında, həm də müəllimlər arasında keçirilir. Belə ki, 2008-ci ildən şagirdlər arasında müxtəlif nominasiyalar üzrə “Gənc kompüter istedadları” müsabiqəsi keçirilmişdir. Hər il müəllimlər arasında “İnnovativ müəllimlər” müsabiqəsi keçirilir. Cari ildə Azərbaycan məktəbliləri ilk dəfə olaraq informatika üzrə beynəlxalq müsabiqə olan “Bebras” (“Qunduz”) yarışmasına qatılmışlar. Ancaq məktəblilərin Respublika İnformatika Olimpiadası həm əhəmiyyətinə, həm də dövlət dəstəyinə görə əsas yer tutur, çünki bu olimpiadanın qalibləri respublikamızın şərəfini Beynəlxalq İnformatika Olimpiadalarında qoruyurlar.



### **İnformatika olimpiadalarının sonrakı tarixinə ötəri baxış:**

- **1991-ci ildə olimpiadanın respublika turu 25-30 martda Bakı şəhəri fizika-riyaziyyat təmayüllü 1 nömrəli internat məktəbdə təşkil edilmişdir.** Respublika turunun nəticələrinə görə 2 nəfər 1-ci, 1 nəfər isə 3-cü yeri tutmuşdur.
- **1992-ci ildə Respublika İnformatika Olimpiadasının yekun turunun 25-29 martda keçirilməsi nəzərdə tutulmuşdu.** Ancaq ölkədə yaranmış ictimai-siyasi vəziyyətlə əlaqədar bu tur keçirilməmişdir.

<b>Olimpiada iştirakçılarının iş rejimi</b>	
<b>25 mart</b>	
Olimpiada iştirakçılarının qəbulu və yerləşdirilməsi	08.00 – 21.00
Səhər yeməyi	08.00 – 11.00
Nahar	14.00 – 15.00
Kinofilmə baxış (yerləşdikləri internat məktəblərində)	16.00 – 18.00
Axşam yeməyi	19.00 – 20.00
Mədəni-kütləvi tədbirlər	20.00 – 22.00
Yuxu	22.00
<b>26 mart</b>	
Yataqdan qalxma	07.00 – 08.00
Səhər yeməyi	08.00 – 09.00
Olimpiadanın açılışı	11.00 – 13.00
Nahar	14.00 – 15.00
Komanda rəhbərlərinin iclası	16.00 – 18.00
Olimpiada iştirakçıları üçün məsləhət (yaşadıqları internatlarda)	18.00 – 19.00
Axşam yeməyi	19.00 – 20.00
Olimpiadaya müstəqil hazırlaşmaq	20.00 – 22.00
Yuxu	22.00
<b>27 mart</b>	
Yataqdan qalxma	07.00 – 08.00
Səhər yeməyi	08.00 – 09.00
Nəzəri yarış (komandaların yerləşdikləri internat məktəblərində)	10.00 – 15.00
Nahar	16.00 – 17.00
Sərbəst vaxt	17.00 – 19.00

Axşam yeməyi	19.00 – 20.00
Turun nəticələrinin elan olunması	21.00-dan
Yuxu	22.00
<b>28 mart</b>	
Yataqdan qalxma	07.00 – 08.00
Səhər yeməyi	08.00 – 09.00
Praktik iş. Təcrübə işlərinin yerinə yetirilməsi: Kimyadan – Respublika 5 №-li internat məktəbində Fizikadan – 1 №-li internat məktəbində İnformatikadan – Bakı Müəllimləri Təkmilləşdirmə İnstitutunda Riyaziyyat olimpiadasının iştirakçıları üçün ekskursiya	10.00 – 12.00
Münaqişə məsələlərinə komissiya tərəfindən baxılması	12.00 – 20.00
Nahar	14.00 – 15.00
Mədəni-kütləvi tədbirlər	16.00 – 18.00
Axşam yeməyi	19.00 – 20.00
Kinofilmə baxış	20.00 – 22.00
Yuxu	22.00
<b>29 mart</b>	
Yataqdan qalxma	07.00 – 08.00
Səhər yeməyi	08.00 – 09.00
Sərbəst vaxt	09.00 – 11.00
Olimpiadaların yekunlaşdırılması	11.00 – 13.00
Nahar	13.00 – 14.00
İştirakçıların yola salınması	15.00-dan
<b>Qeyd.</b> Gedə bilməyən iştirakçıların 30 mart tarixdə yola salınması davam etdiriləcək.	

1992-ci ildə məktəblilərin respublika fənn olimpiadaları üçün  
hazırlanmış nümunəvi proqram

- **1993-cü ildə olimpiadanın respublika turu 25-29 mart tarixində Fizika, riyaziyyat və informatika təmayüllü liseydə (keçmiş Fizika-riyaziyyat təmayüllü 1 nömrəli internat məktəbdə) təşkil edilmişdir. Respublika turunun nəticələrinə görə 1 nəfər 1-ci yerə layiq görülmüşdür.**

- Təhsil nazirinin 11.01.1994-cü il tarixli 41 nömrəli əmri ilə “Məktəblilərin fənn olimpiadaları haqqında Əsasnamə” təsdiq edilmişdir.
- **1994-cü ilin martında** keçirilmiş sayca 5-ci olan Respublika İnformatika Olimpiadasında 8 iştirakçı arasında 1 nəfər 1-ci, 1 nəfər 2-ci, 1 nəfər isə 3-cü yeri tutmuşdur. Həmin ilin 3-10 iyul tarixində İsveçdə keçirilən 6-cı Beynəlxalq İnformatika Olimpiadasında ilk dəfə olaraq Azərbaycan komandası da iştirak etmişdir.



Təhsil naziri Lidiya Rəsulova respublika fənn olimpiadalarının qalibləri ilə (1994)

- **1995-ci ildə** obyektiv səbəbdən (13-17 mart hadisələri ilə əlaqədar) informatika üzrə olimpiadanın yekun turu keçirilməmişdir.
- Təhsil nazirinin 25 noyabr 1996-cı il tarixli 613 nömrəli əmri ilə yenidən işlənmiş “Məktəblilərin fənn olimpiadaları haqqında Əsasnamə” təsdiq edilmişdir.
- **1996-cı ildə** ilk dəfə olaraq olimpiadanın yekun (respublika) turu Tələbə Qəbulu üzrə Dövlət Komissiyasının hazırladığı testlər əsasında keçirilmişdir. Respublika turunun nəticələrinə görə 1 nəfər 1-ci yeri tutmuşdur. Həmin ilin 25 iyul – 2 avqust tarixində Macarıstanda keçirilən 8-ci Beynəlxalq İnformatika Olimpiadasında Azərbaycan komandasının üzvü **Elnur Seyidli** ölkəmizə ilk medalı (bürünc) qazandırmışdır.

- Təhsil nazirinin 04.11.1997-ci il tarixli 777 nömrəli əmri ilə “İnformatika və hesablama texnikasının əsasları” fənni üzrə münasiflər heyətinin sədri Azərbaycan Dövlət Neft Akademiyasının kafedra müdiri Sabit Qəhrəman oğlu Kərimov təsdiq edilmişdir.
- **1997-ci ildə respublika fənn olimpiadaları** 5 fənn üzrə (riyaziyyat, informatika, fizika, kimya və biologiya) keçirilmiş, informatikadan 10 nəfər iştirakçıdan 1-ci yerə layiq görülən olmamış, 1 nəfər 2-ci, 1 nəfər isə 3-cü yer tutmuşdur. Təhsil nazirinin 1996-1997-ci dərslər ilində məktəblilərin fənn olimpiadalarının nəticəsi haqqında verdiyi əmrə “İnformatika və hesablama texnikasının əsasları” fənni üzrə ümumtəhsil məktəbləri şagirdlərinin daha dərin bilik əldə etmələri, onların bu fənnə marağının artırılması, bu fənn üzrə növbəti olimpiadalarda iştirakçıların sayının çoxalması və yüksək nəticələr əldə edilməsi üçün müvafiq tədbirlərin müəyyənləşdirilməsi təhsil şöbə müdirlərinə tapşırılmışdır.
- **1998-ci ildə informatika fənni üzrə olimpiadanın respublika turu** mart ayında Fizika, riyaziyyat və informatika təmayüllü liseydə təşkil edilmişdir. Nəticələrə əsasən 1 nəfər 1-ci, 2 nəfər isə 3-cü yerə layiq görülmüşdür.
- **1999-cu ildə fənn olimpiadalarının respublika turu** 22-25 mart 1999-cu il tarixində keçirilmişdir. Açıliş mərasimi 22 martda saat 15-də Ü.Hacıbəyov adına Musiqi Akademiyasında, bağlanması isə 25 martda saat 10-da Azərbaycan Dövlət Gənc Tamaşaçılar Teatrında olmuşdur. Fizika, riyaziyyat və informatika təmayüllü liseydə keçirilmiş İnformatika Olimpiadasının nəticələrinə görə 2 nəfər 1-ci, 2 nəfər 2-ci, 2 nəfər isə 3-cü yeri tutmuşdur.
- **2000-ci ildə respublika fənn olimpiadalarının yekun turu** 22-25 mart tarixində keçirilmişdir. Qapanış mərasimi 25 martda Ü.Hacıbəyov adına Musiqi Akademiyasında təşkil olunmuşdur. Məktəblilərin fənn olimpiadalarının rayon (şəhər) mərhələsinin keçirilməsi qaydalarında dəyişiklik edilməsi haqqında Təhsil nazirinin 06.12.1999-cu il tarixli 873 nömrəli əmrinə əsasən, olimpiadaların keçirilməsi işini təkmilləşdirmək və respublika turuna daha layiqli şagirdlərin göndərilməsini təmin etmək məqsədilə fənn olimpiadalarının zona mərhələsi ləğv edilmiş, rayon (şəhər) mərhələsi 2 turdan ibarət olmuşdur. 2-ci rayon turunun məsələləri respublika münasiflər heyətinin hazırladığı materiallar əsasında aparılmış və onlar tərəfindən də təkrar yoxlanmışdır. Fizika, riyaziyyat və informatika təmayüllü liseydə təşkil edilmiş informatika olimpiadasının nəticəsinə görə 20 nəfər iştirakçıdan 2 nəfər 1-ci, 1 nəfər 2-ci, 1 nəfər isə 3-cü yerə layiq görülmüşdür.
- **2001-ci ildə Respublika İnformatika Olimpiadasının yekun turu** 26-29 mart tarixində Fizika, riyaziyyat və informatika təmayüllü liseydə təşkil edilmiş, yarış 27 martda keçirilmişdir. Yarışlara qatılan 19 nəfər iştirakçıdan 2 nəfər

1-ci, 1 nəfər 2-ci, 2 nəfər isə 3-cü yeri tutmuşdur. Fənn olimpiadalarının qapanış mərasimi 29 mart saat 12-də Azərbaycan Dövlət Pedaqoji Universitetində keçirilmişdir. Olimpiadanın yüksək səviyyədə keçirilməsi, materialların hazırlanması, işlərin yoxlanılması və yerlərin müəyyənləşdirilməsində daha səmərəli fəaliyyət göstərdiklərinə görə ilk dəfə olaraq ayrı-ayrı fənlər üzrə münisiflər heyətinin sədrləri, məsul katiblər və üzvlərin hər biri Təhsil nazirinin müvafiq əmri ilə pulla mükafatlandırılmışdır.

- **2002-ci ildə informatika olimpiadasının respublika turu** 25-28 mart tarixində Bakı şəhərində təşkil olunmuşdur. Yarış 20.03.2002-ci il tarixində ilk dəfə olaraq Bakı Dövlət Universitetində keçirilmiş, bağlanması isə Fizika, riyaziyyat və informatika təmayüllü liseydə olmuşdur. Olimpiadada 23 nəfər iştirakçıdan 3 nəfər 1-ci, 2 nəfər 2-ci, 2 nəfər isə 3-cü yerə layiq görülmüşdür.

Təhsil nazirinin 07.11.2002-ci il tarixli, 1041 nömrəli əmri ilə “Məktəblilərin fənn olimpiadaları haqqında Qaydalar” təsdiq edilmişdir.

- **2003-cü ildə 13-cü Respublika İnformatika Olimpiadası** 25-28 martda keçirilmiş, yarış Bakı Dövlət Universitetində təşkil olunmuş, olimpiadanın bağlanması isə Fizika, riyaziyyat və informatika təmayüllü liseydə olmuşdur. Yekun turda 33 şagird iştirak etmiş, onlardan 8 nəfəri mükafata layiq yerlər tutmuşdur.
- **2004-cü ildə informatika olimpiadasının respublika turunda** 36 şagird iştirak etmiş, onlardan 3 nəfər 1-ci, 3 nəfər 2-ci, 5 nəfər isə 3-cü yerə layiq görülmüşdür. İlk dəfə olaraq şagirdləri I yer tutmuş fənn müəllimləri də pulla mükafatlandırılmışdır.
- **2005-ci ildə informatika olimpiadasının respublika turu** 16-19 martda Bakı Dövlət Universitetində keçirilmiş, olimpiadanın bağlanması isə Fizika, riyaziyyat və informatika təmayüllü liseydə təşkil edilmişdir. Olimpiadanın nəticəsinə görə 33 yarış iştirakçısından 3 nəfər 1-ci, 6 nəfər 2-ci, 6 nəfər isə 3-cü yerə layiq görülmüşdür.
- **2006-cı ildə 16-cı Respublika İnformatika Olimpiadası** 16-19 martda Bakı Dövlət Universitetində təşkil edilmiş, qapanış mərasimi isə 19 martda Fizika, riyaziyyat və informatika təmayüllü liseydə keçirilmişdir. Bu turda 37 şagird iştirak etmiş, onlardan 3 nəfər 1-ci, 4 nəfər 2-ci, 5 nəfər isə 3-cü yeri tutmuşdur.
- **2007-ci ildə informatika olimpiadasının respublika turu** 16-18 mart tarixində Bakı Dövlət Universitetində keçirilmiş, olimpiadanın bağlanması isə Fizika, riyaziyyat və informatika təmayüllü liseydə təşkil edilmişdir. Bu

turda 33 şagird iştirak etmiş, onlardan 2 nəfər 1-ci, 2 nəfər 2-ci, 2 nəfər isə 3-cü yerə layiq görülmüşdür.

- **2008-ci ildə** 18-ci Respublika İnformatika Olimpiadası 16-18 martda Bakı Dövlət Universitetində keçirilmiş, olimpiadanın bağlanması isə Fizika, riyaziyyat və informatika təmayüllü liseydə təşkil edilmişdir. Yarışlara qatılmış 38 məktəblidən 14-ü mükafata layiq yerlər tutmuşdur.

*Azərbaycan Respublikası Prezidentinin 17 aprel 2006-cı il tarixli 1414 nömrəli Sərəncamı ilə təsdiq edilmiş "Xüsusi istedadla malik olan uşaqların (gənclərin) yaradıcılıq potensialının inkişafı üzrə Dövlət Proqramı"nın 5.14-cü bəndinin icrası ilə bağlı Təhsil nazirinin 04.02.2008-ci il tarixli, 171 nömrəli əmri ilə dünya olimpiadalarında iştirak üçün seçilmiş istedadlı şagirdlərin hazırlıq səviyyəsinin yüksəldilməsinə daha əlverişli şərait yaratmaq məqsədilə onların hər ilin noyabr və aprel aylarında dərş məşğələlərindən azad olunması məqsəda uyğun hesab edilmişdir (bax: səh. 244).*

- **2009-cu ildə** Respublika İnformatika Olimpiadasının yekun turunun 18-20 aprel tarixində keçirilməsi nəzərdə tutulmuş, ancaq obyektiv səbəbdən 2-3 mayda Bakı Dövlət Universitetində keçirilmişdir. Bu turda 43 şagird iştirak etmiş, onlardan 3 nəfər 1-ci, 2 nəfər 2-ci, 7 nəfər isə 3-cü yeri tutmuşdur.
- **2010-cu ildə** 20-ci Respublika İnformatika Olimpiadasının yekun turu 18 martda Bakı Dövlət Universitetində keçirilmişdir. Bu turda 41 şagird iştirak etmiş, onlardan 4 nəfər 1-ci, 4 nəfər 2-ci, 4 nəfər isə 3-cü yerə layiq görülmüşdür.
- **2011-ci ildə** İnformatika üzrə olimpiadanın respublika turu 18 martda Bakı Dövlət Universitetində keçirilmişdir. Qapanış mərasimi isə, son illər olduğu kimi, Fizika, riyaziyyat və informatika təmayüllü liseydə təşkil edilmişdir. Bu turda 42 şagird iştirak etmiş, onlardan 4 nəfər 1-ci, 3 nəfər 2-ci, 5 nəfər isə 3-cü yerə layiq görülmüşdür. Respublika fənn olimpiadalarının qaliblərinin mükafatlandırılması mərasimi 5 may tarixində Təhsil Nazirliyində təşkil edilmişdir.
- **2012-ci ildə** İnformatika üzrə respublika olimpiadasının yekun turu 18 martda Bakı Dövlət Universitetində keçirilmişdir. Bu turda Respublika İnformatika Olimpiadalarının tarixində ən çox sayda – 51 nəfər şagird iştirak etmiş, onlardan 5 nəfər 1-ci yer, 8 nəfər 3-cü yerə layiq görülmüş, 2-ci yer isə olmamışdır.
- **2013-cü ildə** Respublika İnformatika Olimpiadası 17-18 martda Bakı Dövlət Universitetində təşkil olunmuşdur. Yekun turda iştirak etmiş 46 şagirdədən 3 nəfəri 1-ci, 3 nəfəri 2-ci yerə layiq görülmüş, 3-cü yer isə olmamışdır.

- **2014-cü ildə XXIV Respublika İnformatika Olimpiadasının yekun turu 16-17 martda Bakı Dövlət Universitetində keçirilmişdir.** Bu turda 38 şagird iştirak etmiş, onlardan 3 nəfər 1-ci, 3 nəfər 2-ci və 3 nəfər 3-cü yerə layiq görülmüşdür.



Təhsil Nazirliyində respublika fənn olimpiadalarının qaliblərinin mükafatlandırılması mərasimi (17.05.2014)

## Respublika İnformatika Olimpiadalarının xronologiyası

№	İl	Keçirildiyi yer	Tarix
1	1989	Bakı şəhəri. Fizika-riyaziyyat təmayüllü 1 nömrəli internat məktəb	25-31 mart
2	1990	Sumqayıt şəhəri. Fizika-riyaziyyat təmayüllü 1 nömrəli internat məktəb	20-23 aprel
3	1991	Bakı şəhəri. Fizika-riyaziyyat təmayüllü 1 nömrəli internat məktəb	25-30 mart
	1992	<i>İnformatika üzrə olimpiadanın respublika turu obyektiv səbəblərdən keçirilməmişdir.</i>	
4	1993	Bakı şəhəri. Fizika, riyaziyyat və informatika təmayüllü lisey	25-29 mart
5	1994	Bakı şəhəri. Fizika, riyaziyyat və informatika təmayüllü lisey	mart
	1995	<i>İnformatika üzrə olimpiadanın respublika turu obyektiv səbəblərdən keçirilməmişdir.</i>	
6	1996	Bakı şəhəri. Fizika, riyaziyyat və informatika təmayüllü lisey	mart
7	1997	Bakı şəhəri. Fizika, riyaziyyat və informatika təmayüllü lisey	mart
8	1998	Bakı şəhəri. Fizika, riyaziyyat və informatika təmayüllü lisey	mart
9	1999	Bakı şəhəri. Fizika, riyaziyyat və informatika təmayüllü lisey	22-25 mart
10	2000	Bakı şəhəri. Fizika, riyaziyyat və informatika təmayüllü lisey	22-25 mart
11	2001	Bakı şəhəri. Fizika, riyaziyyat və informatika təmayüllü lisey	26-29 mart
12	2002	Bakı şəhəri. Bakı Dövlət Universiteti	25-28 mart
13	2003	Bakı şəhəri. Bakı Dövlət Universiteti	25-28 mart
14	2004	Bakı şəhəri. Bakı Dövlət Universiteti	22-25 mart
15	2005	Bakı şəhəri. Bakı Dövlət Universiteti	16-19 mart
16	2006	Bakı şəhəri. Bakı Dövlət Universiteti	16-19 mart
17	2007	Bakı şəhəri. Bakı Dövlət Universiteti	16-18 mart
18	2008	Bakı şəhəri. Bakı Dövlət Universiteti	16-18 mart
19	2009	Bakı şəhəri. Bakı Dövlət Universiteti	2-4 may
20	2010	Bakı şəhəri. Bakı Dövlət Universiteti	17-18 mart
21	2011	Bakı şəhəri. Bakı Dövlət Universiteti	17-18 mart
22	2012	Bakı şəhəri. Bakı Dövlət Universiteti	17-18 mart
23	2013	Bakı şəhəri. Bakı Dövlət Universiteti	17-18 mart
24	2014	Bakı şəhəri. Bakı Dövlət Universiteti	16-17 mart



## Respublika İnformatika Olimpiadalarının nəticələri

N	İllər	İştirakçıların sayı	Mükafatçılar			
			I yer	II yer	III yer	Cəmi
I	1989	5	1	-	-	1
II	1990	10	1	-	-	1
III	1991	6	2	-	1	3
	1992 *					
IV	1993	7	1	-	-	1
V	1994	8	1	1	1	3
	1995 *					
VI	1996 **	6	1	-	-	1
VII	1997	10	-	1	1	2
VIII	1998	23	1	-	2	3
IX	1999	29	2	2	2	6
X	2000	20	2	1	1	4
XI	2001	19	2	1	2	5
XII	2002	23	3	2	2	7
XIII	2003	33	3	3	2	8
XIV	2004	36	3	3	5	11
XV	2005	33	3	6	6	15
XVI	2006	37	3	4	5	12
XVII	2007	33	2	2	2	6
XVIII	2008	38	3	4	7	14
XIX	2009	43	3	2	7	12
XX	2010	41	4	4	4	12
XXI	2011	42	4	3	5	12
XXII	2012	51	5	-	8	13
XXIII	2013	46	3	3	-	6
XXIV	2014	38	3	3	3	9
Yekun		637	56	45	66	167

\* 1992 və 1995-ci illərdə İnformatika üzrə olimpiadanın respublika turu obyektiv səbəblərdən keçirilməmişdir.

\*\* 1996-cı ildə Respublika İnformatika Olimpiadası ilk və sonuncu dəfə Tələbə Qəbulu üzrə Dövlət Komissiyasının hazırladığı testlər əsasında aparılıb.

### 1.3. BEYNƏLXALQ İNFORMATİKA OLİMPIADALARI

İnformatika üzrə Beynəlxalq Olimpiadanın keçirilməsi ideyası ilk dəfə 1987-ci ilin oktyabrında Paris şəhərində keçirilən YUNESKO-nun 24-cü konfransında bolqarıstanlı nümayəndə, professor Blaqovest Sendov tərəfindən irəli sürülmüşdür. İlk Beynəlxalq İnformatika Olimpiadası (International Olympiad in Informatics, IOI) isə 1989-cu ilin may ayında YUNESKO-nun təşkilatçılığı və sponsorluğu ilə Bolqarıstanın Pravets şəhərində keçirilmişdir. Bu olimpiadada 13 ölkənin komandası iştirak etmişdir. Həmin vaxtdan hər il müxtəlif ölkələrdə keçirilən Beynəlxalq İnformatika Olimpiadalarında iştirakçı ölkələrin sayı ilbəil artır və artıq 80-i ötüb.

Azərbaycan bu olimpiadaya 1993-cü ildə dəvət alıb, ancaq həmin il viza problemi səbəbindən Argentinada keçirilən yarışlara qatıla bilməyib. Məktəblilərimiz ilk dəfə 1994-cü ildə İsveçdə keçirilən 6-cı Beynəlxalq İnformatika Olimpiadasında iştirak edib.

Növbəti olimpiadanın keçiriləcəyi ölkə və yer Olimpiadanın Beynəlxalq Komitəsinin qərarı ilə 3 il əvvəldən müəyyənləşdirilir. Beynəlxalq İnformatika Olimpiadasında iştirak etmək üçün namizəd ölkələr kifayət qədər mürəkkəb müsabiqə prosedurundan keçməlidir. Ona görə də hər il iştirak etmək istəyənlərin sayının çox olmasına baxmayaraq iştirakçıların sayı bir o qədər də artmır.

Yarışda hər bir iştirakçıya 5 saat müddətində həll etmək üçün 3, yaxud 4 məsələ verilir. Hər bir iştirakçıya fərdi kompüter ayrılır. İştirakçılar həll etdikləri məsələləri şəbəkə vasitəsilə yoxlayıcı sistemə göndəirlər. Yoxlayıcı sistem iştirakçının proqramını öz verilənləri ilə yoxlayır və yoxlamanın nəticəsini iştirakçıya bildirir. Beləliklə, bütün yoxlamalar avtomatik aparılır və nəticələr yarış bitən kimi bəlli olur.

*(Daha ətraflı bax: İsmayıl Calallı. Beynəlxalq İnformatika Olimpiadaları, "Bakınəşr", 2012)*

## Beynəlxalq İnformatika Olimpiadalarının təxmini gündəliyi

1-ci gün	Yarış iştirakçılarının qarşılanması
2-ci gün	Kompüterdə praktik işlər (sistemlə tanışlıq). Olimpiadanın təntənəli açılışı
3-cü gün	Yarışın 1-ci turu
4-cü gün	İstirahət. İdman, ekskursiya
5-ci gün	Yarışın 2-ci turu
6-cı gün	İstirahət. Böyük ekskursiya
7-ci gün	Olimpiadanın təntənəli bağlanması. Qaliblərin mükafatlandırılması
8-ci gün	Yarış iştirakçılarının yola salınması



25-ci Beynəlxalq İnformatika Olimpiadasında yarış zamanı (Avstraliya, Brisben şəhəri, 08.07.2013)

## II BÖLÜM

# RESPUBLİKA İNFORMATİKA OLİMPIADALARININ TƏŞKİLİ VƏ KEÇİRİLMƏ QAYDALARI

### 2.1. OLİMPIADANIN KEÇİRİLMƏ MƏRHƏLƏLƏRİ

Respublika İnformatika Olimpiadası Azərbaycan Respublikası Təhsil Nazirliyi tərəfindən təsdiq edilmiş “Məktəblilərin fənn olimpiadaları haqqında Qaydalar”a uyğun olaraq keçirilir. Hazırda qüvvədə olan Qaydalara əsasən, Respublika İnformatika Olimpiadası üç mərhələdən ibarətdir.

**Birinci mərhələ** məktəb olimpiadasıdır. Bu mərhələ rayon (şəhər) üzrə münisiflər heyəti tərəfindən hazırlanıb təsdiq edilmiş məsələlər əsasında tədris ilinin əvvəlində, adətən, oktyabr-noyabr aylarında keçirilir.

**İkinci mərhələ** rayon və şəhərlərdə keçirilir. Bu mərhələ iki turdan ibarət olur. Birinci turun məsələləri rayon (şəhər) üzrə münisiflər heyəti tərəfindən hazırlanıb təsdiq edilmiş məsələlər əsasında dekabr ayında keçirilir və bu turda məktəb olimpiadasının qalibləri iştirak edirlər. İkinci rayon (şəhər) turuna 1-ci turun qalibləri vəsiqə qazanır. Olimpiada məsələləri və qiymətləndirmə meyarları respublika münisiflər heyəti tərəfindən hazırlanıb yerlərə göndərilir. Həmin tur, adətən, məktəblilərin qış tətili dövründə keçirilir. İkinci turda məsələlərin həlləri rayon (şəhər) üzrə münisiflər heyəti tərəfindən yoxlanıb qiymətləndirilir, birinci yeri tutmuş şagirdlərin işləri növbəti respublika mərhələsinə buraxılmaq üçün Təhsil Nazirliyinə göndərilir. Burada isə həmin işlər respublika münisiflər heyəti tərəfindən təkrar yoxlanılaraq qiymətləndirilir və müvafiq bal toplayanlar növbəti mərhələyə vəsiqə qazanırlar.

Qeyd edək ki, əvvəlki illərdə 2-ci rayon turu əvəzinə zona turları keçirilirdi. Belə ki, hər zonaya daxil edilən rayonların qalibləri həmin turda iştirak edərək respublika turuna vəsiqə qazanırdılar.

**Üçüncü mərhələ** respublika olimpiadasıdır və Təhsil Nazirliyi tərəfindən məktəblilərin yaz tətili dövründə keçirilir. Olimpiada məsələləri, qiymətləndirmə meyarları, iştirakçılar üçün yaddaş respublika münisflər heyətinin sədri tərəfindən hazırlanır və olimpiada başlananda iştirakçılara təqdim edilir.

Yeri gəlmişkən, Azərbaycan SSR Xalq Təhsili Nazirliyinin 10 oktyabr 1989-cu il tarixli, 1100 nömrəli əmri ilə təsdiq edilmiş “Məktəblilərin fənn olimpiadaları haqqında Əsasnamə”nin IV bölməsinin 2-ci bəndinə əsasən, 1990-cı ildən başlayaraq fənn olimpiadalarının respublika mərhələsində 1-ci yeri tutan şagirdlər respublikanın ali və orta ixtisas məktəblərinə müəyyənləşdirilmiş qaydada imtahansız qəbul olunurdular. Ancaq 1993-cü ildə bu qayda (bu güzəşt) ali məktəblərə qəbulda yeni sistemə (imtahanların mərkəzləşmiş qaydada TQDK tərəfindən aparılmasına) keçildikdən sonra ləğv edildi.

**Yekun mərhələ** Beynəlxalq İnformatika Olimpiadasıdır. Bu mərhələyə respublika mərhələsinin qalibləri arasından müəyyən olunmuş sayda – dörd şagird seçilir.

Azərbaycan Respublikası Təhsil nazirinin 04.02.2008-ci il tarixli, 171 nömrəli əmrinə əsasən, Beynəlxalq fənn olimpiadalarında iştirak üçün seçilmiş istedadlı şagirdlərin hazırlıq səviyyəsinin yüksəldilməsinə daha əlverişli şərait yaratmaq məqsədilə onlar hər ilin noyabr və aprel aylarında dərş məşğələlərindən azad olunur. Orta təhsil kursunu bitirən şagirdlərə qızıl və gümüş medal verilərkən də onların respublika və beynəlxalq olimpiadalarda iştirakı nəzərə alınır. Eləcə də beynəlxalq olimpiadaların qalibləri ali məktəblərimizin müvafiq ixtisaslarına imtahansız qəbul olunurlar.

## **2.2. OLİMPIADANIN RƏHBƏR ORQANLARI**

Müxtəlif fənlər üzrə respublika olimpiadalarının keçirilməsinin təşkilati-metodik təminatı üçün Təhsil Nazirliyi tərəfindən daimi fəaliyyət göstərən Təşkilat Komitəsi yaradılır. Təşkilat Komitəsinin tərkibinə elm və təhsilin idarəetmə orqanlarının, ictimai təşkilatların nümayəndələri daxil edilir və Təhsil nazirinin əmri ilə bu tərkib təsdiq edilir.

Fənlər üzrə olimpiadaların keçirilməsinin metodik təminatı üçün, hər fənn üzrə olduğu kimi, informatika fənni üzrə də metodik komissiya və münisflər heyəti yaradılır. Münisflər heyəti respublikanın sahə üzrə tanınmış alim, metodist, müəllim, aspirantlarından seçilməklə müəyyənləşdirilir və bu tərkib də Təhsil Nazirliyi tərəfindən təsdiq edilir. Metodik komissiyaya Təşkilat Komitəsinin üzvü olan sədr rəhbərlik edir. Olimpiadanın ilkin mərhələlərinin təşkili üçün yerlərdə də təşkilat komitələri və müvafiq münisflər heyəti yaradılır.

Başqa fənlərdən fərqli olaraq informatika üzrə olimpiadanın spesifik xüsusiyyətləri nəzərə alınaraq metodik komissiya tərəfindən respublika turunun keçirilməsi üçün Texniki qrup təşkil edilir. Bu qrup olimpiada keçirilən zallarda lokal şəbəkədə birləşdirilmiş çoxlu sayda kompüterin işlək vəziyyətdə olmasını təmin edir, hər bir kompüterə müvafiq proqram təminatını yükləyib işə salır, olimpiadanın gedişində hər hansı texniki nasazlığı aradan qaldırmağa kömək edir və olimpiada işlərinin şəbəkə vasitəsilə mərkəzi serverə əvvəlcədən hazırlanmış müvafiq test proqramı vasitəsilə qiymətləndirilməsi üçün göndərilməsinə kömək edir. Belə qrupun Respublika İnformatika Olimpiadasının keçirilməsində rolu danılmazdır. Çoxillik təcrübə göstərir ki, Texniki qrupun təcrübəsi, proqram təminatı ilə bağlı, eləcə də texniki problemləri qısa müddətdə və lazımi anda aradan qaldırması olimpiadanın gedişinə, hər bir iştirakçının taleyinə çox müsbət təsir göstərir.



Respublika İnformatika Olimpiadasının münisflər heyəti.  
Soldan sağa: Mətanət Əhmədova, Nəbi Mahmudov, Ramin Mahmudzadə,  
Valid Məhərrəmov, Aygün Əzizova, Akif Əliyev (18.03.2013)

### 2.3. OLİMPIADA İŞTİRAKÇILARININ TƏRKİBİ

Ümumtəhsil məktəblərinin istənilən VI-XI sinif şagirdi Respublika İnformatika Olimpiadasının iştirakçısı ola bilər. Hər bir sonrakı mərhələyə əvvəlki mərhələnin qalıbları buraxılır və onların siyahısı təhsil orqanlarının əmri ilə təsdiq edilir. İkinci mərhələdən başlayaraq hər sonrakı mərhələdə həm də əvvəlki ildə müvafiq mərhələnin qalıbları da iştirak edə bilər; məsələn, respublika turunda 2014-cü ildə qalib olmuş şagirdlər əvvəlki mərhələlərdə iştirak etmədən 2015-ci ildə keçiriləcək olimpiadanın respublika turunda iştirak etmək hüququna malikdir. Hər mərhələdə iştirakçıların sayı müvafiq mərhələnin təşkilat komitəsi tərəfindən müəyyənləşdirilir.

İştirakçıların yaş səviyyəsi ilə bağlı bir məsələni də qeyd etmək istəyirik: informatika üzrə fənn olimpiadalarının keçirildiyi ilk illərdə iştirakçılar yuxarı sinif şagirdləri sırasından seçilirdi, çünki həmin illərdə bu fənn yalnız yuxarı siniflərdə tədris edilirdi. Son illərdə fənnin aşağı siniflərdən başlayaraq tədris edilməsi iştirakçıların yaş səviyyəsini aşağı salmağa imkan verdi. Digər tərəfdən təcrübə göstərir ki, olimpiadada ilk və son dəfə iştirak edən şagird yüksək nəticə göstərə bilmir və ona görə də istedadlı şagirdlərə yaradıcılıq atmosferi yaratmaq üçün onlara daha aşağı yaşdan şərait yaratmaq lazımdır.

Başqa fənlər üzrə olan olimpiadalardan fərqli olaraq informatika olimpiadalarında bütün iştirakçılara oxuduqları sinifdən asılı olmayaraq eyni məsələlər verilir. Məsələlər təqdim edildikdən sonra ilk bir saat ərzində iştirakçılar münisiflər heyətinin üzvlərinə məsələlərin şərtləri ilə bağlı suallar verə bilərlər. Ancaq suallar elə olmalıdır ki, onlara konkret olaraq ya “hə”, ya da “yox” cavabı vermək mümkün olsun.

Olimpiadanın keçirildiyi müddətdə şəxsi kompüterdən, kalkulyatordan, elektron qeyd kitabçasından, əlaqə vasitələrindən, elektron daşıyıcılardan, eləcə də tədris ədəbiyyatından istifadəyə icazə verilmir. Tur ərəfəsində kompüterlərdə və proqram təminatında nasazlıq olarsa, münisiflər heyətinin qərarı ilə nasazlıqların aradan qaldırılmasına sərf edilən müddət kompensasiya edilir və əlavə vaxt verilir.

Burada sual doğuran məsələlərdən biri də müxtəlif yaşda olan şagirdlərə eyni sualların təqdim edilməsidir. Bu sual münisiflər heyətinin iclaslarında da çox müzakirə edilmişdir. Əvvəlcə məsələləri iki kateqoriyaya bölməklə problemin həllinə nail olmağa çalışılmışdır. Ancaq Beynəlxalq İnformatika Olimpiadalarının təcrübəsi nəzərə alınaraq bu variantdan imtina edilmişdir. Kompromis olaraq hazırda belə bir variant tətbiq edilir: hər turda mütləq bir məsələ çətinlik dərəcəsinə görə aşağı siniflərin informatika proqramına və bu şagirdlərin yaş səviyyəsinə uyğun seçilir. Bu metodika son illərdə daha da təkmilləşdirilir; məsələn, praktik olaraq hər bir məsələnin müxtəlif çətinlikli bir neçə həllinin

olması nəzərdə tutulur. Bu halda ən sadə həll variantı aşağı sinif şagirdlərinin səviyyəsinə uyğun hesab edilir.

İndiyədək bu metodika özünü doğrultmuşdur. Belə ki, aşağı siniflərdən olimpiadalarda iştirak edən şagirdlər yaşca böyük yoldaşları ilə müqayisədə psixoloji cəhətdən mühitə daha tez uyğunlaşmış və onlardan daha yüksək nəticə göstərmişlər.

## **2.4. OLİMPIADANIN KEÇİRİLMƏ QAYDALARI VƏ QALIBLƏRİN MÜƏYYƏNLƏŞDİRİLMƏSİ**

İnformatika olimpiadaları proqramlaşdırma dillərindən istifadə etməklə alqoritmik məsələlərin kompüterdə həlli üzrə müsabiqədir. Mahiyyət etibarilə bu olimpiadalar 1989-cu ildən başlayaraq hər il keçirilən Beynəlxalq İnformatika Olimpiadalarına şagird seçimi mərhələlərindən biridir. Beynəlxalq olimpiada iştirakçılarının hazırlığına verilən tələblər çox yüksəkdir və ona görə də bizim olimpiadaların bütün mərhələləri (məktəb, rayon, respublika turları) respublika olimpiada komandasına layiqli namizədlərin seçilməsi məqsədinə xidmət edir.

### **1. Məktəb olimpiadası**

İnformatika üzrə məktəb olimpiadalarının keçirilməsi məktəblilərin bu fənnə marağının artırılması üçün güclü vasitə ola bilər və olmalıdır. Məktəb olimpiadasının keçirilməsi zamanı başlıca məqsəd şagirdlərin kütləvi şəkildə bu tədbirə cəlb edilməsidir. Bu, şagirdlərə özünü göstərməyə, informatika sahəsində daha məqsədyönlü inkişaf etməyə imkan verir. Bu, eyni zamanda, informatika üzrə olimpiadanın rayon turuna iştirakçılar seçməyə imkan verir. Olimpiadanın mövzusu və nümunəvi məsələlər öncədən elan olunmalı və bütün şagirdlərin nəzərinə çatdırılmalıdır. Məktəb olimpiadasında iştirak üçün yaş məhdudiyyəti qoyulmamalıdır. Dünya təcrübəsi göstərir ki, bu olimpiadanın qalibi ən gənc iştirakçı ola bilər.

Məktəblilərin hazırlıq səviyyəsini müəyyənləşdirmək üçün çətinliyi artırmaqla 6-8 məsələ təklif etmək olar. Birinci məsələ ən sadə, sonuncu iki məsələ isə rayon olimpiadasının səviyyəsində olmalıdır.

İştirak etmək istəyənlərin sayı çox olduqda iki tur keçirmək olar. Olimpiadanın kompüterdən istifadə edilməklə keçirilməsi məqsəduyğundur. İkinci rayon turuna seçilmiş şagirdlər isə mütləq proqram tərtib edib onu kompüterdə həll etməyi bacarmalıdırlar.



## **2. Rayon olimpiadaları**

Rayon olimpiadaları (I və II tur) daha istedadlı şagirdlərin yarışın növbəti – respublika mərhələsinə seçilməsinə xidmət edir. İlk illərdə, daha doğrusu, 1991-1996-cı illərdə olimpiadanın bir rayon mərhələsi olurdu. Sonrakı mərhələ zona mərhələsi idi və bu mərhələdə Təhsil Nazirliyinin təsdiq etdiyi zonalar üzrə həmin zonaya daxil olan rayonların rayon mərhələsinin qalibləri iştirak edirdilər.

Rayon olimpiadasının II turu, adətən, hər ilin yanvar ayının əvvəllərində informatika üzrə respublika münisflər heyətinin hazırladığı məsələlər əsasında keçirilir. Mürəkkəbliyinə görə rayon turunun məsələləri respublika turunun məsələlərinə yaxın olmalıdır. Bu olimpiada mütləq kompüterdən istifadə ilə keçirilməlidir. Respublika olimpiadasına yalnız o şagirdlər keçməlidir ki, onlar proqramı tərtib edib kompüterdə icra edə bilirlər.

## **3. Respublika olimpiadası**

Başqa fənlərdə olduğu kimi, Respublika İnformatika Olimpiadasının da yekun turu, adətən, məktəblilərin yaz tətili günlərində keçirilir. Respublika turunun məsələləri rayon turunun məsələlərindən çətin, ancaq Beynəlxalq İnformatika Olimpiadasının məsələlərinin səviyyəsindən bir qədər aşağı olur.

Olimpiada kompüterdən istifadə edilməklə aparılır. Hər bir şagird üçün bir kompüter ayrılır. Olimpiada tapşırıqları, adətən, 5 məsələdən ibarət olur və onların həllinə 4 saat vaxt verilir. 4 saatdan sonra bütün məsələlərin həlli toplanılır və münisflər heyəti hər bir iştirakçı üçün məsələlərin həllini əvvəlcədən hazırlanmış testlərlə yoxlayır. Hər məsələ üzrə düzgün testlərin sayı həmin məsələ üzrə qazanılan balın miqdarını müəyyən edir. Bütün həll edilmiş məsələlər üzrə balların cəmi iştirakçının yekun balını müəyyənləşdirir.

Respublika olimpiadasının yekun turunun qalibləri və mükafat alanların sayı ümumi iştirakçıların 50%-dən artıq olmamalıdır. I, II və III dərəcəli diplomların sayının isə hər sinif üzrə təqribən 1:2:3 nisbətində (1 ədəd 1-ci dərəcəli, 2 ədəd 2-ci dərəcəli, 3 ədəd 3-cü dərəcəli diplom) olması nəzərdə tutulur. Ancaq bəzən bu nisbətə riayət etmək mümkün olmur, yəni ya yüksək bal toplayanların sayı az olur, ya da həddindən artıq çox olur. Bu halda münisflər heyətinə yaranmış vəziyyətə uyğun hərəkət etməyə icazə verilir.



22-ci Respublika İnformatika Olimpiadasının yekun turu (18.03.2012)

### **Olimpiadanın proqram mühiti**

Hər bir iştirakçının istifadəsinə aşağıdakı proqram təminatı yüklənmiş bir kompüter verilir: OC Microsoft Windows, translyatorlar – Borland Pascal 7.0, Free Pascal, C, C++, Delphi 7.0, QBasic 4.5.



### **P.S.**

Azərbaycan Respublikası Təhsil nazirinin 12 dekabr 2014-cü il tarixli 1256 nömrəli əmri ilə məktəblilərin fənn olimpiadalarının təşkili və keçirilməsinin yeni qaydaları təsdiq edilmişdir (*bax*: 245).

## III BÖLÜM

# OLİMPIADA MƏSƏLƏLƏRİNİN MƏZMUNU VƏ OLİMPIADAYA HAZIRLIQ ÜÇÜN METODİK TÖVSIYƏLƏR

### 3.1. OLİMPIADA MƏSƏLƏLƏRİNƏ QOYULAN TƏLƏBLƏR

İnformatika üzrə olimpiadada şagirdlərin nailiyyətləri olimpiada üçün seçilmiş və hazırlanmış məsələlərdən çox asılıdır. Ona görə də olimpiada məsələləri elə verilməlidir ki, onun sayəsində ən istedadlı şagirdləri seçmək mümkün olsun. Bu cəhətdən informatika üzrə olimpiada məsələlərinə müəyyən tələblər qoyulur. Olimpiadanın hər mərhələsinin məsələləri, yuxarıda qeyd edildiyi kimi, müvafiq metodik komissiyalar tərəfindən hazırlanır. Bu komissiyaların işinə kömək məqsədilə bəzi məqamlara diqqət yetirməyi zəruri hesab edirik.

Beləliklə, informatika üzrə olimpiadalar, xüsusilə rayon və respublika turu üçün məsələlərin hazırlanması o qədər də sadə iş deyil. Belə ki, olimpiada məsələlərinin hazırlanması prosesi bir neçə mərhələdən ibarətdir:

- ideyanın işlənilməsi və məsələnin şərtinin hazırlanması;
- məsələnin həllinin müxtəlif alqoritmlərinin işlənilib hazırlanması;
- həll proqramlarının olimpiadada istifadə olunan proqramlaşdırma dillərinin hamısında yazılması və onların sazlanması;
- qiymətləndirmə sisteminin hazırlanması;
- məsələnin həllinin yoxlanılması üçün testlər sisteminin işlənilib hazırlanması;
- avtomatlaşdırılmış test sisteminin tərkibinə daxil etmək üçün yoxlayıcı proqramın işlənilib hazırlanması.

İlkin mərhələ ən vacib mərhələdir. Burada əsas diqqət hazırlanacaq məsələlərə qoyulan tələblərə yönəlməlidir. Respublika olimpiadasının yekun turlarının təcrübəsi göstərir ki, bu tələbləri aşağıdakı kimi qruplaşdırmaq olar:

1. Hər bir məsələnin həllinin nəticəsi icraya hazır proqram, yaxud həmin proqramın icrasından sonra alınan çıxış verilənləri faylı olmalıdır. Burada iştirakçıdan nəyin isə isbatı tələb olunmur.
2. Məsələnin əsasında orijinal formallaşdırma və ya həll üçün orijinal ideya durmalıdır. Bu, əsas tələblərdən biridir.
3. Məsələ birqiymətli müəyyən olunmalıdır. Məsələnin şərtində qeyri-müəyyənlik olmamalıdır.
4. Məsələnin həlli üçün xüsusi biliklərə ehtiyac olmamalıdır. Ancaq bu, nəzəri hazırlığı olmayan bir şagirdin həmin olimpiada məsələsini həll edə bilməsi demək deyildir.
5. Məsələ elə hazırlanmalıdır ki, onun şərti formallaşdırma mərhələsindən mütləq keçə bilsin.
6. İnformatika üzrə olimpiada məsələsi informatika kursunu harmonik olaraq əhatə etməlidir və hər bir məsələ kompüterdə məsələ həllinin bütün mərhələlərindən (məsələnin formallaşdırılması, alqoritmin qurulması, proqramın tərtibi və sazlanması, hazırlanmış proqramın icrası və testlənməsi) istifadə etməklə yerinə yetirilməlidir.
7. Məsələ elə çətinlikdə olmalıdır ki, heç olmasa, güclü iştirakçılardan bir neçəsi onu həll edə bilsin, zəif iştirakçılar isə ən sadə testlərlə, heç olmasa, az da olsa xal ala bilsinlər.
8. Məsələ elə tərtib olunmalıdır ki, onun həm tam, həm də xüsusi həll variantları olsun. Bu, aşağı sinif şagirdlərinin də, heç olmasa, xüsusi həll variantını işləyə bilmələrinə imkan verir.
9. Məsələlərdən ən azı biri qalanlardan daha sadə olmalıdır ki, hər bir iştirakçı müəyyən xal toplaya bilsin və ruhdan düşməsin.

Əlbəttə, yuxarıda söylənilənlər subyektiv fikirlərdir və olimpiadalar üzrə çoxillik təcrübəyə əsaslanır. Ancaq olimpiada məsələlərinin çətinlik dərəcəsi ildən-ildə artırılmalıdır.

### 3.2. OLİMPIADA MƏSƏLƏLƏRİNİN MÜASİR MƏZMUNU

İnformatika üzrə olimpiadaların keçirildiyi ilk illərdə şagirdlərə həm nəzəri, həm də praktik məsələlər verilir, olimpiada nəzəri və praktik olmaqla iki turdan ibarət olurdu. Nəzəri turun suallarının qiymətləndirilməsi subyektiv amillərdən asılı olduğundan münisflər heyətinin üzvləri arasında bəzən fikir ayrılığı yaranırdı. Getdikcə yalnız bir tura keçildi və qiymətləndirmə avtomatlaşdırılmış qiymətləndirmə sistemi vasitəsilə aparılmağa başladı. Beləliklə, ildən-ilə həm olimpiada məsələlərinin məzmununda, həm də qiymətləndirmə texnologiyasında təkmilləşmələr getdi və bu proses bu gün də davam etdirilir. Ancaq unutmamaq olmur ki, informatika üzrə olimpiadalar fənn üzrə kurikulumdakı məzmun standartları ilə əlaqədar məsələləri əhatə etməlidir. Bu məsələlər isə aşağıdakılardır:

- alqoritmik təfəkkürün inkişafı;
- kompüterin köməyi ilə məsələ həllinin müxtəlif texnoloji aspektlərinin mənimsənilməsi;
- hesablama mühiti şəraitində işləmək bacarıqlarının formalaşdırılması.

Qeyd edilənləri nəzərə almaqla olimpiada məsələlərinin müasir məzmunu informatikanın aşağıdakı bölmələrini əhatə etməlidir:

1. İnformatikanın riyazi əsasları.
2. Alqoritmlərin işlənilib hazırlanması və təhlili.
3. Proqramlaşdırmanın əsasları.
4. İKT vasitələri.
5. Əməliyyat sistemləri.
6. Proqramlaşdırma texnologiyasının əsasları.
7. Hesablama üsulları və modelləşdirmə.
8. Kompüter şəbəkə texnologiyaları.

Bəzən müəllimlər tərəfindən belə bir fikir səslənir ki, olimpiada məsələlərinin həlli məktəb kursundan əlavə biliklər tələb edir. Bəli, bu doğrudur, ancaq bu biliklər kurikulumdakı məzmun standartlarının tələbindən kənarda olmayan, baza biliklərinin üzərində formalaşdırıla bilən, xüsusilə təmayül siniflərdə verilən biliklərdir.

### 3.3. OLİMPIADA MƏSƏLƏLƏRİNİN HƏLLİ METODİKASI

Hər hansı olimpiada məsələsinin həlli həmişə yaradıcılıq prosesidir və bu işdə müəyyən metodik tövsiyələr yararlı ola bilər. Ancaq bu kitabın həcmi olimpiadaya nəzəri və praktik hazırlıqla bağlı məsələləri tam əhatə etməyə imkan vermədiyindən biz yalnız şagird və informatika müəllimlərinə kömək edə biləcək ümumi istiqamətli metodik tövsiyələrlə kifayətlənəcəyik. Beləliklə, olimpiada məsələsi verilib və onu həll etmək tələb olunur. Nədən başlamalı?

**1-ci növbədə** məsələnin şərti araşdırılmalıdır. Məsələnin şərtindəki hər bir ifadəyə diqqət yetirilməlidir. Bəzən hər bir sözdə müəyyən vacib informasiya olur. Ona görə də zərurət olduqda olimpiada başlanandan bir saat müddətində məsələləri şərtləri ilə bağlı münsiflər heyətinin üzvlərinə izahat üçün suallar vermək olar.

**2-ci addımda** məsələnin şərtini formallaşdırmaq lazımdır. Yəni məsələnin mətninin şifahi təsvirindən onun həllinin formal sxeminə keçmək lazımdır. Bəzən məsələnin şərti sadə və aydın olur, ancaq onun riyazi və alqoritmik şəkildə formallaşdırılması çətinlik yaradır. Bu halda məsələni bir qədər sadələşdirmək, giriş və çıxış verilənlərindən istifadə etməklə sadə hallarda kompüterdən istifadə etmədən həlləri “əllə” almağa çalışmaq lazımdır. Bəzən bu, orijinal ideyanın yaranmasına və əsas məsələnin formallaşdırılmasına kömək edir.

**3-cü addımda** məsələnin həll alqoritmı qurulmalıdır. Bu mərhələ məsələ həllinin ən çox yaradıcılıq tələb edən hissəsidir. Burada informatikanın nəzəri əsaslarından bilikləriniz və olimpiada məsələlərinin həlli üzrə topladığınız təcrübə əsas rol oynayır. Beyninizdə yaranan ideyanı qabaqcadan yoxlamaq və sadə nümunələr üçün “əllə” konkret nəticələr almağa çalışmaq lazımdır. Bu mərhələdə bəzən məsələ həllinin fraqmentlərini kağızda işləmək də kömək edə bilər.

**4-cü addımda** qurulmuş alqoritmə uyğun proqram tərtib edilməlidir. Proqram tərtib edilərkən məsələnin şərtində qeyd edilən giriş və çıxış verilənlərinin formatına fikir vermək lazımdır. Məsələnin şərtində giriş və çıxış fayllarının adları verilmişsə, onları proqramda düzgün göstərmək lazımdır, çünki onların düzgün yazılmaması səhv hesab olunur. Bundan başqa, redaktə müddətində proqramı tez-tez yadda saxlamaq lazımdır, çünki heç kim kompüterdə, eləcə də elektrik sistemində ola biləcək nasazlıqlardan sığortalanmayıb və belə hallar hər an baş verə bilər.

**5-ci addımda** hazırlanmış proqram kompilyasiya olunub icra edilməli, səhvlər düzəldilməlidir.

**Sonuncu – 6-cı addımda** məsələnin həll proqramı yoxlayıcı serverə göndərməli və ya Texniki qrupun üzvünün yaddasaxlama qurğusuna köçürülməlidir.

## **Məktəblilərin Respublika İnformatika Olimpiadasına hazırlaşması üçün zəruri olan mövzu və alqoritmlər**

### **1. Riyaziyyat**

#### **1.1. Hesab və həndəsə**

- Natural ədədlər, tam ədədlər, onlar üzərində əməllər və tam ədədlərin müqayisəsi
- Tam ədədlərin xassələri. Müsbət, mənfi, sadə və mürəkkəb ədədlər. Qarşılıqlı sadə ədədlər. Mükəmməl ədədlər
- Kəsrlər. Adi və onluq kəsrlər
- Nöqtə, vektor. Müstəvi üzərində koordinatlar. Müstəvi üzərindəki nöqtələr arasında Evklid məsafəsi
- Xətlər. Parçalar. Parçaların kəsişməsi
- Bucaqlar
- Üçbucaqlı, kvadrat, düzbucaqlı, çevrə
- Üçbucaqlının görkəmli xətləri və nöqtələri
- Çoxbucaqlılar (təpələri, tərəfləri, sahəsi, qabarıqlıq)

#### **1.2. Ardıcılıqlar**

- Ədədi və həndəsi ardıcılıqlar
- Fibonaççi ardıcılığı
- Rekurrent münasibətlər
- Rekursiya anlayışı

#### **1.3. Kombinatorika**

- Permutasiyalar, aranjemanlar, kombinezonlar (təkrarsız)
- Permutasiyalar, aranjemanlar, kombinezonlar (təkrarlı)
- Nyuton binomu

#### **1.4. Diskret strukturlar**

- Çoxluqlar
- Çoxluqlar üzərində əməllər: birləşməsi, kəsişməsi, fərqi
- Stek, dek, növbə
- Qraf. Əsas anlayışlar və təriflər. Qrafların təsvir üsulları: qonşuluq matrisi, tillər siyahısı, qonşu təpələrin siyahısı, erişimlilik anlayışı

### **2. Alqoritmlər**

#### **2.1. Sadə alqoritmlər**

- Tam ədədin rəqəmləri ilə iş
- Natural ədədlərin sadə vuruqlara ayrılması. Sadə ədədlərin tapılması. Sadə ədədlərin tapılmasının Eratosfen metodu
- Ən böyük ortaq bölən və ən kiçik ortaq bölünənin tapılması

#### **2.2.**

- 2.3. Kombinator alqoritmlər
  - Kombinator obyektlərin generasiyası. Obyektə görə nömrənin hesablanması. Obyektin nömrəyə görə tapılması
  - Ədədlərin toplananlara ayrılması
- 2.4. Çəşidləmə və axtarış
  - Sadə əvəzləmə ilə çəşidləmə
  - Artırmalı çəşidləmə
  - Ən kiçiyi seçməklə çəşidləmə
  - Xətti axtarış
  - Binar axtarış
- 2.5. Alqoritmlər həndəsədə
  - Həndəsi fiqurların sahələrinin hesablanması
  - Parçaların kəsişməsinin müəyyən olunması
  - Çoxbucaqlının qabarıqlığının müəyyən olunması
  - Sadə çoxbucaqlının sahəsinin müəyyən olunması
  - Sadə həndəsi fiqurların kəsişməsinin müəyyən olunması
- 2.6. Seçim və onun qısaldılması metodları
  - Variantların seçimi
  - Dönüşlü seçim
  - Dinamik proqramlaşdırma alqoritmləri
  - “Açgöz” alqoritmlər
  - Dalğavarı alqoritmlər
- 2.7. Qraflarla bağlı alqoritmlər
  - Qrafda ən qısa yolların axtarışı
  - Dərinliyə axtarış
  - Eninə axtarış





## IV BÖLÜM

# OLİMPİADANIN MƏKTƏB TURUNA HAZIRLIQ ÜÇÜN MƏSƏLƏLƏR

### 4.1. MƏSƏLƏLƏR

**A01.** Üçrəqəmli ədəd verilib. Bu ədədin rəqəmləri arasına boşluq simvolu qoymaqla çıxışa verin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
123	1 2 3
987	9 8 7
100	1 0 0

**A02.** İki tam ədəd verilib. Onların qiymətlərinin yerini dəyişin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 3	3 5
56 56	56 56
-4 7	7 -4

**A03.**  $X$  və  $Y$  dəyişənlərinin qiymətlərini elə paylayın ki,  $X$  onların böyüyü,  $Y$  isə kiçiyi olsun.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
7 3	7 3
-3 8	8 -3
-3 -1	-1 -3

**A04.** İkirəqəmli tam ədəd verilib. Onun cüt olub-olmadığını müəyyənləşdirin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
25	NO
46	YES
1234	YES

**A05.** İkirəqəmli tam ədəd verilib. Bu ədədin öncə soldakı rəqəmini (onluqları), sonra isə 2 boşluq simvolundan sonra sağdakı rəqəmini (təklidləri) çıxışa verin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
98	9 8
82	8 2
11	1 1

**A06.** İkirəqəmli müsbət tam ədəd verilib. Onun rəqəmlərinin cəmini və hasilini tapın.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
34	7 12
40	4 0
99	18 81

**A07.** İkirəqəmli tam ədəd verilib. Bu ədədin rəqəmlərinin yerini dəyişməklə çıxışa verin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
30	3
33	33
–90	–9

**A08.** Verilmiş natural ədədin sağdan üçüncü rəqəmini çıxışa verin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1234567	5
98765	7
34	0

**A09.** Üçrəqəmli tam ədəd verilib. Bu ədədin öncə sonuncu rəqəmini (təklidləri), sonra isə ortadakı rəqəmini (onluqları) çıxışa verin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
307	70
520	02
100	00

**A10.** Üçrəqəmli tam ədəd verilib. Onun rəqəmlərinin cəmini və hasilini tapın.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
456	15 120
–456	–15 –120
230	5 0

**A11.** Üçrəqəmli tam ədəd verilib. Bu ədədin rəqəmlərini sağdan sola oxuduqda alınan ədədi çıxışa verin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
123	321
999	999
560	65

**A12.** Üçrəqəmli tam ədəd verilib. Onun soldan birinci rəqəmini pozub ədədin sağına yazdılar. Alınan ədədi çıxışa verin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
123	231
999	999
120	201

**A13.** Üçrəqəmli tam ədəd verilib. Onun sağdan birinci rəqəmini pozub ədədin soluna yazdılar. Alınan ədədi çıxışa verin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
123	312
999	999
120	12

**A14.** Üçrəqəmli tam ədəd verilib. Bu ədədin yüzliklərinin və onluqlarının yerini dəyişməklə alınan ədədi çıxışa verin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
123	213
999	999
105	15

**A15.** Üçrəqəmli tam ədəd verilib. Bu ədədin onluqlarının və təklirlərinin yerini dəyişməklə alınan ədədi çıxışa verin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
123	132
999	999
105	150

**A16.** Üçrəqəmli müsbət tam ədəd verilib. Həmin ədədin başlanğıcına və sonuna 3 rəqəmini yazmaqla yeni ədəd alın.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
123	31233
999	39993
105	31053

**A17.** 999-dan böyük tam ədəd verilib. Bir qalıqsız bölmə (**div**) və bir bölmənin qalığını alma (**mod**) əməlindən istifadə etməklə həmin ədədin yazılışındakı minlik mərtəbənin rəqəmini tapın.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
12345	2
123456	3
1234567	4

**A18.** Dödrəqəmli tam ədəd verilib. Bu ədədin palindrom olub-olmadığını müəyyən edin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1234	NO
1221	YES
9999	YES

**A19.** Beşrəqəmli ədəd verilib. Bu ədədin ilk iki rəqəminin cəmi ilə son iki rəqəminin cəminin fərqini hesablayın.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
12345	-6
99999	0
98765	6

**A20.** Beşrəqəmli ədəd verilib. Bu ədədin palindrom olub-olmadığını müəyyən edin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
12345	NO
12321	YES
99999	YES

**A21.** Beşrəqəmli ədəd verilib. Bu ədədin rəqəmlərinin kvadratları cəmini tapın.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
12345	55
99999	405
11111	5

**A22.** Beşrəqəmli ədəd verilib. Bu ədədin soldan birinci rəqəmdən başlayaraq bütün rəqəmlərinin artan sıra ilə düzülüb-düzülmədiyini müəyyən etmək lazımdır. Məsələn, 15689 ədədinin bütün rəqəmləri artan sıra ilə düzülüb.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
12345	YES
21345	NO
13245	NO

**A23.** Beşrəqəmli ədəd verilib. Bu ədədin rəqəmlərini əks sıra ilə yazmaqla yeni ədəd alın. Məsələn, verilən ədəd 27346-dırsa, nəticədə 64372 ədədi alınmalıdır.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
12345	54321
99999	99999
12340	4321

**A24.** Beşrəqəmli ədəd verilib. Bu ədədin yazılışında eyni rəqəmlərin olub-olmadığını müəyyənləşdirmək lazımdır.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
12345	NO
11345	YES
12145	YES

**A25.** Dödrəqəmli tam ədəd verilib. Bu ədəddə düz üç eyni rəqəmin olub-olmadığını müəyyənləşdirin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1234	NO
1114	YES
1211	YES

**A26.** Dödrəqəmli tam ədəd verilib. Onun bütün rəqəmlərinin fərqli olub-olmadığını müəyyənləşdirin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1234	YES
1134	NO
1214	NO

**A27.** Dördrəqəmli tam ədəd verilib. Bu ədədin yazılışında 3 rəqəminin olub-olmadığını müəyyənləşdirin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1245	NO
3124	YES
1324	YES

**A28.** Dördrəqəmli tam ədəd verilib. Bu ədədin yazılışında ardıcıl gələn 3 və 7 rəqəmlərinin olub-olmadığını müəyyənləşdirin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1372	YES
1237	YES
1327	NO

**A29.** Üçrəqəmli tam ədəd verilib. Bu ədədin Armstrong ədədi olub-olmadığını müəyyənləşdirin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
153	YES
154	NO
370	YES

**A30.** Dördrəqəmli tam ədəd verilib. Onun rəqəmlərinin cəminin 2-nin və 3-ün misli olub-olmadığını müəyyənləşdirin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1234	NO
9999	YES
1221	YES



**A31.** Döndrəqəmli tam ədəd verilib. Bu ədədin özünün bütün rəqəmlərinə bölünüb-bölünmədiyini müəyyənləşdirin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1 2 3 4	NO
2 2 2 2	YES
3 2 2 2	YES

**A32.** X, Y, Z həqiqi ədədləri verilib. Bu ədədlərin ən böyüyünü tapın.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2    5    3	5
5    3    2	5
-2   -7   -3	-2

**A33.** X, Y, Z həqiqi ədədləri verilib. Bu ədədlərin ən kiçiyini tapın.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2    5    3	2
5    3    2	2
-2   -7   -3	-7

**A34.** X, Y, Z həqiqi ədədləri verilib. Əgər  $Z \geq Y \geq X$  olarsa, bu ədədləri ikiqat artırın, əks halda onları mütləq qiymətləri ilə əvəz edin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2    4    9	4.00   8.00   18.00
-7   -3   -2	-14.00   -6.00   -4.00
-3   -4    8	3.00   4.00   8.00

**A35.** İki həqiqi ədəd verilib. Əgər birinci ədəd ikincidən böyükdürsə, birinci ədədi çıxışa verin, əks halda hər iki ədədi çıxışa verin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3   -7	3.00
-3   -7	-3.00
2   5	2.00   5.00

**A36.** İki həqiqi ədəd verilib. Əgər birinci ədəd ikincidən kiçikdirsə, yaxud ona bərabərdirsə, birinci ədədi sıfırla əvəzləyin, əks halda heç nəyi dəyişməyin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
8   8	0.00   8.00
-8   -3	0.00   -3.00
-3   -6	-3.00   -6.00

**A37.** Üç həqiqi ədəd verilib. Onlardan (1, 3) intervalına daxil olan ədədləri müəyyən edin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2.1   2.5   2.8	2.10 2.50 2.80
2   5   8	2.00
5   2   5	2.00

**A38.** X və Y ( $Y \neq X$ ) həqiqi ədədləri verilib. Onların ədədi ortasını hesablayıb kiçik olan ədədə, ikiqat hasilini isə o biri (böyük) ədədə mənimsədin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4   2	16.00   3.00
3   8	5.50   48.00
4   -2	-16.00   1.00

**A39.** Üç həqiqi ədəd verilib. Onlardan mənfi olmayanları kvadrata yüksəldin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2 4 8	4.00 16.00 64.00
-3 2 5	-3.00 4.00 25.00
2 -5 5	4.00 -5.00 25.00

**A40.** X və Y ( $Y \neq X$ ) həqiqi ədədləri verilib. Əgər X mənfidirsə, onlardan böyüyünü, əks halda kiçiyini çıxışa verin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 8	5.00
-4 8	8.00
-4 -7	-4.00

**A41.** A, B, C və D həqiqi ədədləri verilib. Əgər  $D \leq C \leq B \leq A$  olarsa, onlardan ən böyüyünün qiymətini bütün dəyişənlərə mənimsədin; əgər  $A < B < C < D$  olarsa, heç bir dəyişiklik edilmir; əks halda bütün ədədləri özlərinin kvadratları ilə əvəz edin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 4 4 4	4 4 4 4
2 3 4 5	2 3 4 5
2 5 3 4	4 25 9 16

**A42.** X və Y həqiqi ədədləri verilib. Əgər hər iki ədəd mənfidirsə, onların qiymətlərini öz modulları ilə əvəzləyin; əgər onlardan biri mənfidirsə, hər iki qiyməti 0.5 artırın; əgər hər iki qiymət mənfi deyilsə və onlardan biri  $[0.5, 2.0]$  parçasına daxil deyilsə, hər iki qiyməti 10 dəfə azaldın; qalan hallarda heç bir dəyişiklik etməyin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
-5 3	-4.50 3.50
5 -3	5.50 -2.50
0.1 3	0.01 0.30

**A43.** A, B, R, S tam ədədləri verilib ( $A \geq 0, B > 0$ ). A-nın B-yə bölünməsindən alınan qalığın R, yaxud S-ə bərabər olub-olmadığını müəyyənləşdirin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
10 3 1 3	YES
10 3 2 2	NO
0 2 1 0	YES

**A44.** A, B, C həqiqi ədədləri verilib. Tərəfləri bu ədədlər olan üçbucaqlı qurmaq olarmı?

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 5 8	YES
4 5 9	NO
4 7 25	NO

**A45.** A, B, C həqiqi ədədləri verilib. Əgər tərəfləri bu ədədlər olan üçbucaqlı qurmaq mümkündürsə, həmin üçbucaqlının növünü (düzbucaqlı, itibucaqlı, yaxud korbucaqlı olmasını) müəyyən edin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3 4 6	Korbucaqlı
5 4 3	İtibucaqlı
3 4 8	Belə üçbucaqlı yoxdur

**A46.** Dördrəqəmli natural ədəd verilib. Bu ədədin yazılışında bütün cüt rəqəmləri atın (0 cüt rəqəmdir).

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1357	1357
4357	357
1257	157

**A47.** A, B, C, D həqiqi ədədləri verilib. Tərəfləri bu ədədlər olan dördbucaqlının mümkünlüyünü müəyyən edin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1   2   3   4	YES
5   5   5   5	YES
1   2   3   7	NO

**A48.** Natural ədəd verilib. Onun hansısa natural ədədin kvadratı olub-olmadığını müəyyən edin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
49	YES
50	NO
100	YES

**A49.** Üçrəqəmli müsbət ədəd verilib. Onun rəqəmlərinin yerini elə dəyişdirin ki, həmin rəqəmlərdən ibarət maksimal ədəd alınsın.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
124	421
412	421
555	555

**A50.** Üçrəqəmli müsbət ədəd verilib. Onun rəqəmlərinin yerini elə dəyişdirin ki, həmin rəqəmlərdən ibarət minimal ədəd alınsın.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
124	124
412	124
555	555

## 4.2. HƏLLƏR

A01

Bu məsələnin həll algoritmi çox sadədir: üçrəqəmli ədəd daxil edirik, həmin ədədin rəqəmlərini ayırırıq və bu rəqəmlərin aralarında boşluq qoymaqla onları çıxışa veririk. Bu algoritmin proqram şəklində yazılışları isə fərqli ola bilər:

**1-ci üsul (A1a).** Ədədin rəqəmlərini ayırırıq və bu rəqəmləri *a*, *b*, *c* dəyişənlərinə mənim-sədirik. Sonra aralarında boşluq qoymaqla bu dəyişənlərin qiymətlərini ekrana çıxardırıq.

**2-ci üsul (A1b).** Ədədin rəqəmlərini qabaqcadan ayırmırıq, bu rəqəmləri ayıran operatorları birbaşa çıxış operatorlarında (**writeln** və ya **cout**) yazırıq. Nəticədə proqram bir az qısa alınır.

**3-cü üsul (A1c).** Bu üsulda daxil olunan ədədlər yoxlanılır, istifadəçi üçün məlumat verilir və ədədin rəqəmləri ayrı-ayrı dəyişənlərə yazılır. Əlbəttə, burada 2-ci üsulda olduğu kimi, rəqəmləri ayırmamaq da olardı.

**Qeyd 1.** Növbəti misallarda proqramların tərtibatında biz çox zaman 1-ci üsuldan istifadə edəcəyik, ancaq həmin proqramları 2-ci və 3-cü üsuldakı kimi yazmağı da sizə məsləhət görürük. Bununla belə, nəzərə alın ki, bunlar, sadəcə, “zövq” məsələsidir, “Hansı üsul daha yaxşıdır?” sualı isə önəm daşımır.

**Qeyd 2.** Üçüncü üsulda biz daxil olunan ədədin üçrəqəmli olub-olmadığını yoxlayırıq, birinci və ikinci üsullarda isə belə bir yoxlama yoxdur. Bu onu göstərir ki, əgər biz üçrəqəmli ədədin əvəzinə dördürəqəmli ədəd daxil etsək, cavab doğru olmayacaq. Bu çatışmazlığı aradan qaldırmaq üçün

```
c := n div 100;
```

yaxud

```
c = n / 100;
```

operatorlarının yerinə

```
c := (n div 100) mod 10;
```

yaxud

```
c = (n / 100) % 10;
```

operatorlarını yazmaq olar. Gələcəkdə biz tez-tez belə hərəkət edəcəyik.

**Qeyd 3.** Bütün proqramlarda verilənləri **DÜZGÜN** daxil edəcəyik, ancaq istənilən verilənləri daxil etmək istədikdə onların yoxlandığı 3-cü üsulu tətbiq edəcəyik.

Pascal

**Program A01a;**

var

n, a, b, c: integer;

begin

```

    readln(n);

    a := n div 100;
    b := (n div 10) mod 10;
    c := n mod 10;
    writeln(a, ' ', b, ' ', c);
end.

```

#### **Program A01b;**

```

var
    n: integer;

begin
    readln(n);
    writeln(n div 100, ' ', (n div 10) mod 10, ' ', n mod 10);
end.

```

#### **Program A01c;**

```

var
    n, a, b, c: integer;

begin
    write('Üçrəqəmli ədəd daxil edin: n =');
    readln(n);

    if (n < 100) or (n > 999) then
        writeln('Daxil etdiyiniz ədəd düzgün deyil.')
    else begin
        a := n div 100;
        b := (n div 10) mod 10;
        c := n mod 10;
        writeln('Çıxış: ', a, ' ', b, ' ', c);
    end;
    readln;
end.

```

**C++**

#### **//Program A01a**

```

#include <iostream>

using namespace std;

int main()
{
    int n, a, b, c;

    cin >> n;

    a = n / 100;

```

```

    b = (n / 10) % 10;
    c = n % 10;

    cout << a << " " << b << " " << c << endl;

    return 0;
}

//Program A01b
#include <iostream>

using namespace std;

int main()
{
    int n;

    cin >> n;
    cout << n/100 << " " << (n/10)%10 << " " << n%10 << endl;

    return 0;
}

```

```

//Program A01c
#include <iostream>

using namespace std;

int main()
{
    int n, a, b, c;

    cout << "Üçrəqəmli ədəd daxil edin: n =";
    cin >> n;

    if (n < 100 || n > 999)
        cout << "Daxil etdiyiniz ədəd düzgün deyil." << endl;
    else
    {
        a = n / 100;
        b = (n / 10) % 10;
        c = n % 10;

        cout << "Çıxış: " << a << " " << b << " " << c << endl;
    }

    return 0;
}

```



## A02

Belə “mübadilə” üçün biz üçüncü yardımçı dəyişəndən istifadə edəcəyik. Belə bir bənzətmə apararaq; tutaq ki, sizin iki stəkanınız var və bu stəkanların biri “Fanta”, o biri isə “Koka-Kola” ilə doludur. “Fanta” və “Koka-Kola”nın yerini dəyişmək tələb olunur. Aydındır ki, əlavə bir boş qab (stəkan) olmadan bunu etmək mümkün deyil. Dəyişənlərlə də məhz bu cür hərəkət etmək lazımdır.

Öncə **a** dəyişəninin qiymətini yeni dəyişəndə saxlayırıq, yəni **a** dəyişəninin qiymətini yeni **c** dəyişəninə mənimsədirik (**c** := **a**; “Fanta”nı boş stəkana tökürük). Sonra **b** dəyişəninin qiymətini “boşalmış” **a** dəyişəninə keçiririk (**a** := **b**; boşalmış stəkana “Kola”nı tökürük). Nəhayət, **c** dəyişəninin qiymətini **b** dəyişəninə mənimsədirik (**b** := **c**; birinci addımda istifadə etdiyimiz yardımçı stəkandakı “Fanta”nı boş stəkana tökürük).

## Pascal

### Program A02;

```
var
    a, b, c: integer;
begin
    readln(a, b);

    c := a;
    a := b;
    b := c;

    writeln(a, ' ', b);
    readln;
end.
```

## C++

### // Program A02;

```
#include <iostream>

using namespace std;

int main()
{
    int a, b, c;

    cin >> a >> b;

    c = a;
    a = b;
    b = c;

    cout << a << " " << b << endl;

    return 0;
}
```

## A03

Aydındır ki,  $x$  və  $y$  dəyişənlərinin qiymətlərini bir-biriylə o zaman dəyişdirmək lazımdır ki, birinci ədəd ikincidən kiçikdir. Bu əməliyyat üç operatorndan ibarət olduğundan onları MÜTLƏQ **begin** və **end** (yaxud { və }) arasına almaq lazımdır.

### Pascal

#### **Program A03;**

```
var
    x, y, c: integer;

begin
    readln(x, y);

    if (x < y) then begin
        c := x;
        x := y;
        y := c;
    end;

    writeln(x, ' ', y);
    readln;
end.
```

### C++

#### **//Program A03**

```
#include <iostream>

using namespace std;

int main()
{   int x, y, c;

    cin >> x >> y;

    if (x < y) {
        c = x;
        x = y;
        y = c;
    }
    cout << x << " " << y << endl;

    return 0;
}
```

## A04

Əgər 2-yə bölmənin nəticəsində (mod, yaxud %) alınan qalıq 0-a bərabərdirsə, onda ədəd cütdür.

### Pascal

```
Program A04;
var
    n: integer;

begin
    readln(n);

    if (n mod 2 = 0) then
        writeln('YES')
    else
        writeln('NO');

    readln;
end.
```

### C++

```
//Program A04
#include <iostream>

using namespace std;

int main()
{    int n;

    cin >> n;

    if (n % 2 == 0)
        cout << "YES" << endl;
    else
        cout << "NO" << endl;

    return 0;
}
```

## A05

Bu məsələ 1-ci məsələyə bənzərdir: ədədin rəqəmlərini ayırmaq və onların arasına iki boşluq simvolu artırmaqla çıxışa vermək.

### Pascal

#### **Program A05;**

```
var
    n, a, b: integer;

begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;

    writeln (b, ' ', a);
    readln;
end.
```

### C++

#### **//Program A05**

```
#include <iostream>

using namespace std;

int main()
{
    int n, a, b;

    cin >> n;
    a = n % 10;
    b = (n / 10) % 10;

    cout << b << " " << a << endl;

    return 0;
}
```

## A06

Ədədin rəqəmlərini ayırıb onların cəmini və hasilini tapırıq. Qeyd etmək lazımdır ki, Pascal dilində dəyişənin adında böyük, yaxud kiçik hərflərdən istifadə olunmasının fərqi yoxdur, yəni bu dildə *s* və *S* eyni bir dəyişəndir. C++ dilində isə belə deyil, yəni *s* və *S* fərqli dəyişənlərdir. Bu proqramda biz böyük *S* və *P* hərflərindən istifadə etmişik (sadəcə, gözəllik xatirinə).

### Pascal

#### Program A06;

```
var
    n, a, b, S, P: integer;

begin
    readln(n);

    a := n mod 10;
    b := n div 10 mod 10;
    S := a + b;
    P := a * b;

    writeln('S = ', S, ' P = ', P);
    readln;
end.
```

### C++

#### //Program A06

```
#include <iostream>

using namespace std;

int main()
{
    int n, a, b, s, p;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    s = a + b;
    p = a * b;

    cout << s << " " << p << endl;

    return 0;
}
```

Əgər bizdə 5 (a) və 3 (b) rəqəmləri varsa, onda 53 ədədini almaq üçün  $10*5 + 3$  ( $10*a + b$ ) əməliyyatını yerinə yetirməliyik. Ona görə də bu məsələdə öncə rəqəmləri ayırıb sonra m ədədini qururuq.

### Pascal

#### Program A07;

```
var
    n, m, a, b: integer;

begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    m := 10*a + b;

    writeln(m);
    readln;
end.
```

### C++

#### //Program A07

```
#include <iostream>

using namespace std;

int main()
{
    int n, m, a, b;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    m = 10 * a + b;

    cout << m << endl;

    return 0;
}
```

Qeyd edək ki, proqramın yazılışında anlaşıqlılıq üçün biz bəzi operatorları sağa sürüşdürmüşük, **begin** və **end** ( { və } ) isə eyni səviyyədə yazılıb. Bu proqramı biz aşağıdakı kimi də yaza bilərdik və həmin proqram da normal işləyəcək. Ancaq razılaşın ki, yuxarıdakı yazılış şəkli ilə işləmək daha əlverişlidir. Ona görə də bundan sonra yuxarıdakı yazılış şəklinə istifadə edəcəyik.

### Pascal

```
Program A07;var n,m,a,b:integer;begin  readln(n); a:=n mod 10;
b:=(n div 10) mod 10;
  m:=10*a+b; writeln (m); readln;end.
```

### C++

```
//Program A07
#include <iostream>
using namespace std;int main() {int
n,m,a,b;cin>>n;a=n%10;b=(n/10)%10;
m=10*a+b;cout << m << endl;return 0;}
```

### A08

Bu məsələnin həlli çox sadədir: öncə ədədin sağdan iki rəqəmini uzaqlaşdırırıq ( $n \div 100$ , yaxud  $n / 100$ ), sonra isə qalan ədədin axırncı rəqəmini ayırırıq ( $\text{mod } 10$ , yaxud  $\%10$ ).

### Pascal

#### Program A08;

```
var
    n, a: integer;

begin
    readln(n);

    a := (n div 100) mod 10;

    writeln(a);

    readln;
end.
```

### C++

#### //Program A08

```
#include <iostream>

using namespace std;

int main()
{  int n, a;
```

```

    cin >> n;

    a = (n / 100) % 10;
    cout << a << endl;

    return 0;
}

```

## A09

Ədəd üçrəqəmli olsa da, bizə onun yalnız sonuncu iki rəqəmi lazımdır. Onların da necə ayrılmasını bilirsiniz.

## Pascal

### Program A09;

```

var
    n, a, b: integer;

begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    writeln(a, b);
    readln;
end.

```

## C++

### //Program A09

```

#include <iostream>

using namespace std;

int main()
{
    int n, a, b;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;

    cout << a << b << endl;

    return 0;
}

```



## A10

Bu məsələ 6-cı məsələnin tam bənzəridir.

### Pascal

```
Program A10;
var
    n, a, b, c, p, s: integer;

begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    s = a + b + c;
    p = a * b * c;

    writeln(s, ' ', p);
    readln;
end.
```

### C++

```
//Program A10
#include <iostream>

using namespace std;

int main()
{   int n, a, b, c, s, p;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    s = a + b + c;
    p = a * b * c;

    cout << s << "  " << p << endl;

    return 0;
}
```

## A11

Bu məsələ 7-ci məsələ kimi həll olunur. Öncə ədədin rəqəmləri ayrılır. Sonra yeni ədəd formalaşdırılır: kiçik mərtəbədəki rəqəm böyük mərtəbəyə keçdiyindən 100-ə vurulur, ortadakı (onluqdakı) rəqəm yerində qaldığından 10-a vurulur, böyük mərtəbədəki rəqəm təklilik olacağından ona toxunmuruq və bunların hamısı cəmlənir.

### Pascal

#### **Program A11;**

```
var
    n, m, a, b, c: integer;

begin
    write('Input N=');
    readln(n);

    // Rəqəmlərin ayrılması
    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;

    // Yeni ədədin hesablanması
    m := 100*a + 10*b + c;

    writeln('m=', m);
    readln;
end.
```

### C++

#### **//Program A11**

```
#include <iostream>

using namespace std;

int main()
{
    int n, m, a, b, c;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    m = 100*a + 10*b + c;

    cout << m << endl;

    return 0;
}
```

## A12

Yəqin ki, hər şey aydındır. Sadəcə, hər bir rəqəmin harada duracağına diqqət edin.

### Pascal

```
Program A12;
var
    n, m, a, b, c: integer;

begin
    readln(n);

    a := n mod 10;
    b := n div 10 mod 10;
    c := n div 100 mod 10;

    m := 100*b + 10*a + c;

    writeln(m);
    readln;
end.
```

### C++

```
//Program A12
#include <iostream>

using namespace std;

int main()
{
    int n, m, a, b, c;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    m = 100*b + 10*a + c;

    cout << m << endl;

    return 0;
}
```

## A13

Bu məsələnin həllini iki variantda verəcəyik.

*1-ci variant:* bütün rəqəmləri ayırırıq ( $a$  – təklik,  $b$  – onluq,  $c$  – yüzlik) və  $a$ -nı birinci,  $c$ -ni ikinci,  $b$ -ni isə üçüncü yerə qoymaqla  $100*a + 10*c + b$  şəklində birləşdiririk.

*2-ci variant:* sonuncu rəqəmi ayırırıq ( $a$ ), onu verilmiş ədəddən uzaqlaşdırırıq (ikirəqəmli ədəd qalır), sonra isə bu qalan ikirəqəmli ədədi sağdan  $a$  rəqəminə “yapışdırırıq” ( $100*a + b$ ).

### Pascal

#### Program A13a;

```
var
    n, m, a, b, c: integer;

begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    m := 100*a + 10*c + b;

    writeln(m);
    readln;
end.
```

#### Program A13b;

```
var
    n, m, a, b: integer;

begin
    readln(n);
    a := n mod 10;
    b := n div 10;
    m := 100*a + b;

    writeln(m);
    readln;
end.
```

### C++

#### //Program A13a

```
#include <iostream>

using namespace std;
```

```

int main()
{
    int n, m, a, b, c;
    cin >> n;
    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    m = 100*a + 10*c + b;

    cout << m << endl;

    return 0;
}

```

#### //Program A13b

```

#include <iostream>

using namespace std;

int main()
{
    int n, m, a, b;
    cin >> n;
    a = n%10;
    b = n/10;
    m = 100*a + b;

    cout << m << endl;

    return 0;
}

```

### A14

Əvvəlki məsələlərə bənzərdir. Ədədin rəqəmlərini ayırırıq və onları tələb olunan ardıcılıqla düzürük.

### Pascal

#### Program A14;

```

var
    n, m, a, b, c: integer;

begin
    readln(n);
    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;

```

```

    m := 100*b + 10*c + a;

    writeln(m);
    readln;
end.

```

**C++**

```

//Program A14
#include <iostream>

using namespace std;

int main()
{
    int n, m, a, b, c;

    cin >> n;
    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    m = 100*b + 10*c + a;

    cout << m << endl;

    return 0;
}

```

**A15**

Bu məsələnin həlli 14-cü məsələdəki kimidir.

**Pascal**

```

Program A15;
var
    n, m, a, b, c: integer;

begin
    readln(n);
    a := n mod 10;
    b := n div 10 mod 10;
    c := n div 100 mod 10;
    m := 100*c + 10*a + b;

    writeln(m);
    readln;
end.

```

## C++

### //Program A15

```
#include <iostream>

using namespace std;

int main()
{
    int n, m, a, b, c;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    m = 100*c + 10*a + b;

    cout << m << endl;

    return 0;
}
```

## A16

Hər hansı ədədin sağ tərəfinə rəqəm, məsələn, 5 yazmaq üçün həmin ədədi 10-a vurub üzərinə 5 əlavə etmək lazımdır. Məsələn, 9637 ədədinin sonuna 5 rəqəmini qoşmaq  $9637*10 + 5 = 96375$  əməlini yerinə yetirmək deməkdir.

Ədədin soluna bir rəqəm yazmaq üçün ədəddə olan rəqəmlərin sayını, daha doğrusu, yeni rəqəmin ədəddə hansı mərtəbədə duracağını bilmək lazımdır. Məsələn, qoşulan rəqəm 7-dirsə və o, ikinci mərtəbədə duracaqsə, onun qiyməti 70, üçüncü mərtəbədə duracaqsə – 700, beşinci mərtəbədə duracaqsə – 70000 olacaq. Bu məsələdə solə qoşulan rəqəm yeni ədədin beşinci mərtəbəsinə düşür.

***Qeyd.*** Pascal translyatorunun bəzi versiyalarında **integer** tipinin ala biləcəyi maksimal qiymət 32767-dir. Bu məsələdə nəticə göstərilən ədəddən böyük ola bildiyindən biz proqramda **longint** tipindən istifadə edirik. Bu tiptən olan dəyişənin ala biləcəyi maksimal qiymət 2147483647-yə bərabərdir. C++ dilində isə bu məqsədlə **long** tipindən istifadə edirik.

## Pascal

### Program A16;

```
var
    n: integer;
    m: longint;
```

```
begin
```

```

    readln(n);
    m := 30000 + 10*n + 3;
    writeln(m);
    readln;
end.

```

C++

**//Program A16**

```

#include <iostream>

using namespace std;

int main()
{   int n;
    long m;

    cin >> n;
    m = 30000 + 10*n + 3;

    cout << m << endl;

    return 0;
}

```

A17

Pascal

**Program A17;**

```

var
    c: integer;
    n: longint;

begin
    readln(n);

    c := (n div 1000) mod 10;

    writeln(c);
    readln;
end.

```

C++

**//Program A17**

```

#include <iostream>

```



```
using namespace std;
int main()
{   int c;
    long n;

    cin >> n;

    c = (n/1000)%10;

    cout << c << endl;

    return 0;
}
```

### A18

Dörd rəqəmli ədəd palindromdursa, demək, onun birinci (soldan) rəqəmi dördüncüyə, ikinci isə üçüncüyə bərabərdir. Ona görə də məsələnin həll alqoritmi belə olacaq: ədədin bütün rəqəmlərini ayırırıq və uyğun rəqəmləri müqayisə edirik. Əgər müqayisə olunan ədədlər bərabədirsə, verilmiş ədəd palindromdur.

### Pascal

#### **Program A18;**

```
var
    n, a, b, c, d: integer;

begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    d := (n div 1000) mod 10;
    if (a = d) and (b = c) then
        writeln('YES')
    else
        writeln('NO');

    readln;
end.
```

### C++

#### **//Program A18**

```
#include <iostream>

using namespace std;

int main()
```

```

{  int n, a, b, c, d;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    d = (n / 1000) % 10;
    if (a == d && b == c)
        cout << "YES\n";
    else
        cout << "NO\n";

    return 0;
}

```

**A19**

Birinci, ikinci, dördüncü və beşinci rəqəmi ayırıb onların cəmini və fərqini tapırıq.

**Pascal**

**Program A19;**

```

var
    m, a, b, d, e: integer;
    n: longint;

begin
    readln(n);
    a := n mod 10;
    b := (n div 10) mod 10;
    d := (n div 1000) mod 10;
    e := (n div 10000) mod 10;
    m := (e + d) - (b + a);
    writeln(m);
    readln;
end.

```

**C++**

**//Program A19**

```

#include <iostream>

using namespace std;

int main()
{  int a, b, d, e, s;
    long n;

    cin >> n;

```

```

    a = n % 10;
    b = (n / 10) % 10;
    d = (n / 1000) % 10;
    e = (n / 10000) % 10;
    s = (d + e) - (a + b);

    cout << s << endl;

    return 0;
}

```

## A20

Bu məsələ də 16-cı məsələ kimi həll olunur.

## Pascal

### Program A20;

```

var
    a, b, d, e: integer;
    n: longint;

begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    d := (n div 1000) mod 10;
    e := (n div 10000) mod 10;

    if (a = e) and (b = d) then
        writeln ('YES')
    else
        writeln ('NO');

    readln;
end.

```

## C++

### //Program A20

```

#include <iostream>

using namespace std;

int main()
{
    int a, b, d, e;

```

```

long n;

cin >> n;

a = n % 10;
b = (n / 10) % 10;
d = (n / 1000) % 10;
e = (n / 10000) % 10;
if (a == e && b == d)
    cout << "YES\n";
else
    cout << "NO\n";

return 0;
}

```

## A21

Bu çox sadədir. Ədədin bütün rəqəmlərini ayırırıq və tələb olunan cəmi tapırıq.

## Pascal

### Program A21;

```

var
    m, a, b, c, d, e: integer;
    n: longint;

begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    d := (n div 1000) mod 10;
    e := (n div 10000) mod 10;
    m := a*a + b*b + c*c + d*d + e*e;

    writeln(m);
    readln;
end.

```

## C++

### //Program A21

```

#include <iostream>

using namespace std;
int main()

```

```

{   int a, b, c, d, e, s;
    long n;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    d = (n / 1000) % 10;
    e = (n / 10000) % 10;
    s = a*a + b*b + c*c + d*d + e*e;

    cout << s << endl;

    return 0;
}

```

## A22

Ədədin rəqəmlərini ayıraraq onları a, b, c, d, e dəyişənlərinə mənimsədirik.  $(e < d)$  və  $(d < c)$  və  $(c < b)$  və  $(b < a)$  şərtini yoxlayırıq. Əgər şərt ödənilirsə, rəqəmlər artma sırası ilə düzülüb.

## Pascal

### Program A22;

```

var
    a, b, c, d, e: integer;
    n: longint;

begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    d := (n div 1000) mod 10;
    e := (n div 10000) mod 10;
    if (e < d) and (d < c) and (c < b) and (b < a) then
        writeln('YES')
    else
        writeln('NO');
    readln;
end.

```

## C++

### //Program A22

```

#include <iostream>

```

```

using namespace std;

int main()
{   int a, b, c, d, e;
    long n;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    d = (n / 1000) % 10;
    e = (n / 10000) % 10;
    if (e < d && d < c && c < b && b < a)
        cout<<"YES\n";
    else
        cout << "NO\n";

    return 0;
}

```

**A23**

Bütün rəqəmləri ayırıb lazım olan ardıcılıqla düzürük.

**DİQQƏT.**  $((a*10+b)*10+c)*10+d)*10+e$  və  $10000*a+1000*b+100*c+10*d+e$  yazılışları tamamilə ekvivalentdir.

**Pascal**

**Program A23;**

```

var
    a, b, c, d, e: integer;
    n, m: longint;

begin
    readln(n);
    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    d := (n div 1000) mod 10;
    e := (n div 10000) mod 10;
    m := ((a*10 + b)*10 + c)*10 + d)*10 + e;
    { m := 10000*a + 1000*b + 100*c + 10*d + e }

    writeln (m);
    readln;
end.

```

## C++

### //Program A23

```
#include <iostream>

using namespace std;
int main()
{   int a, b, c, d, e;
    long n,m;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    d = (n / 1000) % 10;
    e = (n / 10000) % 10;
    m = 10000*a + 1000*b + 100*c + 10*d + e;

    cout << m << endl;

    return 0;
}
```

## A24

Bu çox asandır. Bütün rəqəmlər ayrılır və bütün cütlər bir-biri ilə müqayisə olunur. Əgər hər hansı iki rəqəm üst-üstə düşərsə, onda ədədin yazılışında eyni rəqəmlər var.

## Pascal

### Program A24;

```
var
    a, b, c, d, e: integer;
    n: longint;

begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    d := (n div 1000) mod 10;
    e := (n div 10000) mod 10;
    if (a = b) or (a = c) or (a = d) or (a = e) or
        (b = c) or (b = d) or (b = e) or
        (c = d) or (c = e) or (d = e) then
        writeln ('YES')
    else
```

```

        writeln ('NO');
    readln;
end.

```

### C++

#### //Program A24

```

#include <iostream>

using namespace std;

int main()
{
    int a, b, c, d, e;
    long n;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    d = (n / 1000) % 10;
    e = (n / 10000) % 10;
    if (a==b || a==c || a==d || a==e ||
        b==c || b==d || b==e ||
        c==d || c==e || d==e)
        cout << "YES\n";
    else
        cout<<"NO\n";

    return 0;
}

```

### A25

Öncə bütün rəqəmləri ayırırıq. Yalnız üç rəqəmin eyni olduğu cəmi 4 kombinasiya var:

1. 1, 2, 3-cü rəqəmlər eynidir, 4-cü fərqli.
2. 1, 2, 4-cü rəqəmlər eynidir, 3-cü fərqli.
3. 1, 3, 4-cü rəqəmlər eynidir, 2-ci fərqli.
4. 2, 3, 4-cü rəqəmlər eynidir, 1-ci fərqli.

Sonra bu kombinasiyalrı müqayisə edirik.

### Pascal

#### Program A25;

```

var
    n, a, b, c, d: integer;

```



```

begin
    readln(n);
    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    d := (n div 1000) mod 10;
    if (a = b) and (a = c) and (a <> d) or
        (a = b) and (a = d) and (a <> c) or
        (a = c) and (a = d) and (a <> b) or
        (b = c) and (b = d) and (b <> a) then
        writeln ('YES')
    else
        writeln ('NO');
    readln;
end.

```

C++

#### //Program A25

```

#include <iostream>

using namespace std;

int main()
{   int n, a, b, c, d;
    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    d = (n / 1000) % 10;
    if ((a == b && a == c && a != d) ||
        (a == b && a != c && a == d) ||
        (a != b && a == c && a == d) ||
        (a != b && b == c && b == d))
        cout << "YES\n";
    else
        cout<<"NO\n";

    return 0;
}

```

Öncə, əlbəttə, rəqəmləri ayırmaq, sonra bütün rəqəmləri bir-biriylə müqayisə etmək lazımdır. Hər hansı müqayisəni təkrarlamamaq, yaxud hansısa cütün müqayisəsini “unutmamaq” üçün belə bir sistemli strategiya tətbiq etmək olar:  $d$ -ni  $c$ ,  $b$  və  $a$  ilə müqayisə etmək;  $c$ -ni  $b$  və  $a$  ilə müqayisə etmək;  $b$ -ni  $a$  ilə müqayisə etmək. Əgər heç bir cüt eyni deyilsə, deməli, ədədin bütün rəqəmləri fərqlidir.

### Pascal

#### Program A26;

```
var
    a, b, c, d, n: integer;

begin
    readln(n);
    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    d := (n div 1000) mod 10;
    if (d = c) or (d = b) or (d = a) or
       (c = b) or (c = a) or (b = a) then
        writeln('NO')
    else
        writeln('YES');
    readln;
end.
```

### C++

#### //Program A26

```
#include <iostream>

using namespace std;

int main()
{
    int n, a, b, c, d;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    d = (n / 1000) % 10;
    if (a != b && a != c && a != d &&
        b != c && b != d && c != d)
        cout << "YES\n" ;
    else
        cout << "NO\n";

    return 0;
}
```

## A27

Həll aydındır: rəqəmlər növbə ilə 3 ilə müqayisə olunur.

### Pascal

#### **Program A27;**

```
var
    a, b, c, d, n: integer;

begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    d := (n div 1000) mod 10;
    if (a = 3) or (b = 3) or (c = 3) or (d = 3) then
        writeln('YES')
    else
        writeln('NO');
    readln;
end.
```

### C++

#### **//Program A27**

```
#include <iostream>

using namespace std;

int main()
{
    int n, a, b, c, d;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    d = (n / 1000) % 10;
    if (a == 3 || b == 3 || c == 3 || d == 3)
        cout << "YES\n" ;
    else
        cout << "NO\n";

    return 0;
}
```

## A28

Dörd rəqəmli ədəddə 3 və 7 rəqəmlərinin ardıcıl yerləşdiyi variantların sayı 3-dür: 37XX, X37X, XX37 (X – ixtiyari rəqəmdir).

### Pascal

#### Program A28;

```
var
    a, b, c, d, n: integer;

begin
    readln(n);
    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    d := (n div 1000) mod 10;
    if (d = 3) and (c = 7) or (c = 3) and (b = 7) or
        (b = 3) and (a = 7) then
        writeln('YES')
    else
        writeln('NO');
    readln;
end.
```

### C++

#### //Program A28

```
#include <iostream>

using namespace std;

int main()
{
    int n, a, b, c, d;
    cin >> n;
    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    d = (n / 1000) % 10;
    if ((d == 3 && c == 7) || (c == 3 && b == 7) ||
        (b == 3 && a == 7))
        cout << "YES\n";
    else
        cout << "NO\n";

    return 0;
}
```

## A29

Xüsusi şərhə ehtiyac yoxdur: rəqəmlər ayrılır, onların kubları cəmi hesablanır və həmin cəm ilkin verilmiş ədədlə müqayisə olunur.

### Pascal

#### Program A29;

```
var
    a, b, c, n: integer;
begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    if a*a*a + b*b*b + c*c*c = n then
        writeln('YES')
    else
        writeln('NO');
    readln;
end.
```

### C++

#### //Program A29

```
#include <iostream>

using namespace std;

int main()
{
    int n, a, b, c;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    if (a*a*a + b*b*b + c*c*c == n)
        cout << "YES\n" ;
    else
        cout << "NO\n";

    return 0;
}
```

## A30

Aydındır ki, bu məsələdə öncə ədədin bütün rəqəmlərini ayırmaq və onların cəmini tapmaq lazımdır. Sonra isə həmin cəmin 2 və 3-ün misli olub-olmadığını yoxlamaq lazımdır. Əlbəttə, cəmin ayrı-ayrılıqda 2-nin və 3-ün misli olub-olmadığını yoxlamaq olar. Ancaq aydındır ki, ədəd 2-yə və 3-ə bölünərsə, o, 6-ya da bölünür və əksinə, ədəd 6-ya bölünərsə, o, 2-yə və 3-ə də bölünür. Ona görə də proqramda ədədin 6-ya bölünüb-bölünmədiyi yoxlanılır.

### Pascal

#### Program A30;

```
var
    a, b, c, d, n: integer;
begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    d := (n div 1000) mod 10;
    if ((d + c + b + a) mod 6 = 0) then
        writeln('YES')
    else
        writeln('NO');
    readln;
end.
```

### C++

#### //Program A30

```
#include <iostream>
using namespace std;
int main()
{
    int n, a, b, c, d;
    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    d = (n / 1000) % 10;
    if ((a + b + c + d) % 6 == 0)
        cout << "YES\n" ;
    else
        cout<<"NO\n";

    return 0;
}
```

## A31

Ədədin öz rəqəmlərindən hər hansı birinə bölünməsinə yoxlamaq üçün, ilk növbədə, həmin rəqəmin 0-dan fərqli olmasını yoxlamaq lazımdır. Programda da ədədi təşkil edən rəqəmlərdən heç birinin 0-a bərabər olmaması və ədədin bütün rəqəmlərə bölünməsi yoxlanılır. Nəticə müsbət olarsa, çıxışa “YES” verilir.

### Pascal

#### Program A31;

```
var
    a, b, c, d, n: integer;
begin
    readln(n);
    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;
    d := (n div 1000) mod 10;
    if (d <> 0) and (c <> 0) and (b <> 0) and (a <> 0) and
        (n mod a = 0) and (n mod b = 0) and
        (n mod c = 0) and (n mod d = 0) then
        writeln('YES')
    else
        writeln('NO');
    readln;
end.
```

### C++

#### //Program A31

```
#include <iostream>
using namespace std;
int main()
{
    int n, a, b, c, d;
    cin >> n;
    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    d = (n / 1000) % 10;
    if (a != 0 && b != 0 && c != 0 && d != 0 &&
        n%a == 0 && n%b == 0 && n%c == 0 && n%d == 0)
        cout << "YES\n";
    else
        cout<<"NO\n";
    return 0;
}
```

## A32

Program A32a-da alqoritm çox sadədir: əgər ədədlərdən biri qalanlardan böyükdürsə, o, maksimaldır. Program A32b-də isə başqa alqoritmdən istifadə olunur (bu alqoritm daha maraqlıdır – onu yadda saxlayın!). Birinci ədəd *max* dəyişəninə mənimsədilir və sonra bu dəyişən qalan ədədlərlə müqayisə olunur. Əgər növbəti ədəd *max*-dan böyükdürsə, *max*-a həmin ədəd mənimsədilir.

*Qeyd.* Bu məsələdə yeni tip ədədlərə – həqiqi ədədlərə rast gəlinir. Pascal dilində bu tip **real**, C++ dilində isə **double** kimi göstərilir.

### Pascal

#### Program A32a;

```
var
    x, y, z, max: real;

begin
    readln(x, y, z);

    if (x > y) and (x > z) then
        max := x
    else
        if (y > x) and (y > z) then
            max := y
        else
            max := z;

    writeln(max:10:2);
    readln;
end.
```

#### Program A32b;

```
var
    x, y, z, max: real;

begin
    readln(x, y, z);

    max := x;
    if (y > max) then
        max := y;
    if (z > max) then
        max := z;

    writeln(max:10:2);
    readln;
end.
```



**//Program A32a**

```
#include <iostream>

using namespace std;

int main()
{  double x,y,z,max;

    cin >> x >> y >> z;

    if (x > y && x > z)
        max = x;
    else
        if (y > x && y > z)
            max = y;
        else
            max = z;

    cout << max << endl;

    return 0;
}
```

**//Program A32b**

```
#include <iostream>

using namespace std;

int main()
{  double x, y, z, max;

    cin >> x >> y >> z;
    max = x;
    if (y > max)
        max = y;
    if (z > max)
        max = z;

    cout << max << endl;

    return 0;
}
```

## A33

Bu məsələ 32-ci məsələyə çox bənzəyir. Sadəcə, proqramda böyükdür (>) işarəsinin yerinə kiçikdir (<) işarəsi qoymaq lazımdır.

### Pascal

#### **Program A33;**

```
var
    x, y, z, min: real;

begin
    readln(x, y, z);

    min := x;
    if (y < min) then
        min := y;
    if (z < min) then
        min := z;

    writeln(min:10:2);
    readln;
end.
```

### C++

#### **//Program A33**

```
#include <iostream>

using namespace std;

int main()
{
    double x, y, z, min;

    cin >> x >> y >> z;

    min = x;
    if (y < min)
        min = y;
    if (z < min)
        min = z;

    cout << min << endl;

    return 0;
}
```

## A34

Alqoritm proqramın mətnindən aydındır. C++ proqramçıları bir şeyi nəzərə almalıdırlar: *mütləq qiyməti tapan **abs** funksiyası **cmath** kitabxanasında yerləşir və ona görə də həmin kitabxana proqrama qoşulub (3-cü sətirdə).*

### Pascal

#### Program A34;

```
var
    x, y, z, a: real;

begin
    readln(x, y, z);

    if (z >= y) and (y >= x) then begin
        x := x * 2;
        y := y * 2;
        z := z * 2;
    end
    else begin
        x := abs(x);
        y := abs(y);
        z := abs(z);
    end;

    writeln(x:8:2, y:8:2, z:8:2);
    readln;
end.
```

### C++

#### //Program A34

```
#include <iostream>
#include <cmath>

using namespace std;

int main()
{
    double x, y, z;

    cin >> x >> y >> z;

    if (z >= y && y >= x) {
        x *= 2;
        y *= 2;
        z *= 2;
    }
    else {
```

```

        x = abs(x);
        y = abs(y);
        z = abs(z);
    }

    cout << x << " " << y << " " << z << endl;

    return 0;
}

```

### A35

Burada hər şey aydındır.

### Pascal

```

Program A35;
var
    a, b: real;

begin
    readln(a, b);

    if (a > b) then
        writeln(a:8:2)
    else
        writeln(a:8:2, ' ', b:8:2);
    readln;
end.

```

### C++

```

//Program A35
#include <iostream>

using namespace std;

int main()
{
    double a, b;
    cin >> a >> b;
    if (a > b)
        cout << a << endl;
    else
        cout << a << " " << b << endl;

    return 0;
}

```

## A36

Programın mətnində hər şey məsələnin şərtində olduğu kimidir.

### Pascal

```
Program A36;
var
  a, b: real;

begin
  readln(a, b);

  if (a <= b) then
    a := 0;

  writeln(a:8:2, ' ', b:8:2);
  readln;
end.
```

### C++

```
//Program A36
#include <iostream>

using namespace std;

int main()
{
  double a, b;
  cin >> a >> b;
  if (a <= b)
    a = 0;

  cout << a << " " << b << endl;
  return 0;
}
```

## A37

Hər bir ədədin şərtdə göstərilmiş intervala daxil olub-olmadığını yoxlayırıq. Bunun necə edildiyinə diqqət yetirin! Bu yoxlamayı əslə riyaziyyatdakı kimi **if** ( $1 < a < 3$ ) şəklində yazmaq olmaz: o, təbii olsa da, **yanlışdır!**)

### Pascal

#### Program A37;

```
var
    a, b, c: real;

begin
    readln(a,b,c);

    if (a > 1) and (a < 3) then
        writeln('a=', a:0:2);
    if (b > 1) and (b < 3) then
        writeln('b=', b:0:2);
    if (c > 1) and (c < 3) then
        writeln('c=', c:0:2);
    readln;
end.
```

### C++

#### //Program A37

```
#include <iostream>

using namespace std;

int main()
{
    double a, b, c;

    cin >> a >> b >> c;

    if (a > 1 && a < 3)
        cout << "a=" << a << endl;
    if (b > 1 && b < 3)
        cout << "b=" << b << endl;
    if (c > 1 && c < 3)
        cout << "c=" << c << endl;

    return 0;
}
```

## A38

Məsələ ilk baxışda çox sadə görünür, ancaq orada çox maraqlı bir nüans var. Bunu bir nümunə üzərində aydınlaşdıraq.

Tutaq ki,  $x$  ədədi 4-ə,  $y$  isə 6-ya bərabərdir. Onda məsələnin şərtinə görə  $x$  ədədi  $(4+6)/2 = 5$ ,  $y$  isə  $2*4*6 = 48$  olmalıdır. Ancaq biz proqramda  $x$ -i 5-ə bərabər etsək, sonra  $y$ -i hesablasaq,  $y = 2*x*y = 2*5*6 = 60$  alarıq (ancaq 48 olmalı idi!). Məsələ ondadır ki, biz  $y$ -i hesablayarkən  $x$ -in dəyişilmiş qiymətindən istifadə etdik. Ona görə də proqramda biz qabaqcadan  $x$  və  $y$  ədədlərinin yarım cəmini və ikiqat hasilini hesablayıb yardımçı  $a$  və  $b$  dəyişənlərində saxlayırıq, yalnız bundan sonra onların yeni qiymətlərini hesablayırıq.

## Pascal

### Program A38;

```
var
    a, b, x, y: real;

begin
    readln(x, y);

    a := (x + y) / 2;
    b := 2*x*y;

    if x < y then begin
        x := a;
        y := b;
    end
    else begin
        x := b;
        y := a;
    end;
    writeln('x=', x:0:2, ' y=', y:0:2);
    readln;
end.
```

## C++

### //Program A38

```
#include <iostream>

using namespace std;
int main()
{
    double x, y, a, b;

    cin >> x >> y;

    a = (x + y) / 2;
    b = 2*x*y;
```

```

    if (x < y) {
        x = a;
        y = b;
    }
    else {
        x = b;
        y = a;
    }
    cout << x << " " << y << endl;

    return 0;
}

```

### A39

Alqoritm aydındır.

**C++ proqramçıları üçün qeyd.** a ədədini kvadrata yüksəltmək üçün  $a = a * a$  yazmaq olar; yaxud belə də yazmaq olar:  $a *= a$ ;  $s = s + 5$  ifadəsini  $s += 5$  şəklində də yazmaq olar;  $y = y / x$  ifadəsini isə  $y /= x$  şəklində də yazmaq olar. Belə daha qəşəng deyilmi?

**Pascal proqramçıları üçün qeyd.** Proqramlarını Pascal dilində yazanların qəlblərini qırmamaq üçün, qeyd edək ki, Free Pascal da belə bir imkana malikdir. Bu imkanı qoşmaq üçün Option/Compiler menyusunda C-like operator bəndini qeyd etmək lazımdır.

### Pascal

#### Program A39;

```

var
    a, b, c: real;

begin
    readln(a, b, c);
    if a > 0 then
        a := sqr(a);
    if b > 0 then
        b := sqr(b);
    if c > 0 then
        c := sqr(c);

    writeln(a:8:2, b:8:2, c:8:2);
    readln;
end.

```

### C++

#### //Program A39

```
#include <iostream>
```



```

using namespace std;

int main()
{
    double a, b, c;

    cin >> a >> b >> c;

    if (a > 0)
        a *= a;
    if (b > 0)
        b *= b;
    if (c > 0)
        c *= c;

    cout << a << " " << b << " " << c << endl;

    return 0;
}

```

**A40**

Alqoritm programının mətnindən aydındır.

**Pascal**

```

Program A40;
var
    x, y, z: real;
begin
    readln(x, y);

    z := x;
    if (x < 0) then begin
        if x < y then
            z := y;
    end
    else begin
        if x > y then
            z := y;
    end;

    writeln(z:8:2);
    readln;
end.

```

## C++

### //Program A40

```
#include <iostream>

using namespace std;

int main()
{
    double x, y, z;
    cin >> x >> y;
    z = x;
    if (x < 0) {
        if (x < y)
            z = y;
    }
    else {
        if (x > y)
            z = y;
    }

    cout << z << endl;

    return 0;
}
```

## A41

Programda, sadəcə, məsələnin şərtində göstərilənləri ardıcıl yerinə yetirmək lazımdır.

## Pascal

### Program A41;

```
var
    a, b, c, d: real;

begin
    readln(a, b, c, d);
    if (d <= c) and (c <= b) and (b <= a) then begin
        d := a;
        c := a;
        b := a;
    end
    else
        if not((a < b) and (b < c) and (c < d)) then begin
            a := sqr(a);
            b := sqr(b);
            c := sqr(c);
            d := sqr(d);
        end;
end;
```

```

        writeln(a:10:2, b:10:2, c:10:2, d:10:2);
        readln;
    end.

```

### C++

#### //Program A41

```

#include <iostream>

using namespace std;

int main()
{
    double a, b, c, d;

    cin >> a >> b >> c >> d;

    if (d <= c && c <= b && b <= a) {
        d = a;
        c = a;
        b = a;
    }
    else
        if (!(a < b && b < c && c < d)) {
            a *= a;
            b *= b;
            c *= c;
            d *= d;
        }

    cout << a << " " << b << " " << c << " " << d << endl;

    return 0;
}

```

### A42

Burada da, əvvəlki məsələdə olduğu kimi, sadəcə, məsələnin şərtini ardıcıl proqramlaşdırmaq lazımdır.

### Pascal

#### Program A42;

```

var
    x, y: real;

begin
    readln(x, y);
    if (x < 0) and (y < 0) then begin

```

```

        x := abs(x);
        y := abs(y);
    end
else begin
    if (x > 0) and (y < 0) or (y > 0) and (x < 0) then begin
        x := x + 0.5;
        y := y + 0.5;
    end
    else
        if ((x < 0.5) or (x > 2)) and ((y < 0.5) or (y > 2))
        then begin
            x := x / 10;
            y := y / 10;
        end;
    end;
end;

writeln(x:0:2, y:10:2);
readln;
end.

```

## C++

### //Program A42

```

#include <iostream>
#include <cmath>
using namespace std;
int main()
{ double x, y;

    cin >> x >> y;

    if (x < 0 && y < 0) {
        x = abs(x);
        y = abs(y);
    }
    else {
        if ((x < 0 && y > 0) || (x > 0 && y < 0)) {
            x += 0.5;
            y += 0.5;
        }
        else
            if ((x < 0.5 || x > 2) && (y < 0.5 || y > 2)) {
                x /= 10;
                y /= 10;
            }
    }
    cout << x << " " << y << endl;

    return 0;
}

```

## A43

Sade yoxlama.

### Pascal

#### **Program A43;**

```
var
    a, b, r, s, d: integer;

begin
    readln(a, b, r, s);

    d := a mod b;

    if (d = r) or (d = s) then
        writeln('YES')
    else
        writeln('NO');
    readln;
end.
```

### C++

#### **//Program A43**

```
#include <iostream>

using namespace std;

int main()
{   int a, b, r, s, d;

    cin >> a >> b >> r >> s;

    d = a % b;
    if (d == r || d == s )
        cout << "YES\n";
    else
        cout << "NO\n";

    return 0;
}
```

Üç parça o zaman üçbucaqlının tərəfləri ola bilər ki, böyük parçanın uzunluğu qalan iki parçanın uzunluqları cəmindən kiçik olsun. Aydındır ki, böyük parçanın uzunluğu qalan iki parçanın uzunluqları cəmindən kiçikdirsə, o biri tərəflərin uzunluqları da qalan iki tərəfin uzunluqları cəmindən kiçik olacaq. Hansı parçanın uzun olmasını müəyyənləşdirməmək üçün biz bu qaydanı başqa cür ifadə edə bilərik: üç parça o zaman üçbucaqlının tərəfləri ola bilər ki, hər bir parçanın uzunluğu qalan iki parçanın uzunluqları cəmindən kiçikdir. Programda da məhz bu yoxlanılır.

### Pascal

#### Program A44;

```
var
    a, b, c: real;

begin
    readln(a, b, c);
    if (a < b + c) and (b < a + c) and (c < a + b) then
        writeln('YES')
    else
        writeln('NO');
    readln;
end.
```

### C++

```
//Program A44
#include <iostream>

using namespace std;

int main()
{
    double a, b, c;

    cin >> a >> b >> c;

    if (a < b + c && b < a + c && c < a + b)
        cout << "YES\n";
    else
        cout << "NO\n";

    return 0;
}
```

## A45

Verilmiş üç parçadan üçbucaqlının qurulmasının mümkün olub-olmadığının necə yoxlanmasını əvvəlki məsələdən bilirik. Üçbucaqlının növünü müəyyən etmək üçün isə əlavə şərtlər ödənilməlidir: əgər üçbucaqlının hər bir tərəfinin kvadratı qalan iki tərəfin kvadratları cəmindən kiçikdirsə, onda üçbucaqlı itibucaqlıdır. Əgər hər hansı tərəfin kvadratı (aydındır ki, ən böyük) qalan iki tərəfin kvadratları cəmindən böyükdürsə, üçbucaqlı korbucaqlıdır. Üçüncü halda isə üçbucaqlı düzbucaqlıdır.

### Pascal

#### Program A45;

```
var
    a, b, c: real;

begin
    readln(a, b, c);

    if (a < b + c) and (b < a + c) and (c < a + b) then begin
        if (a*a<b*b+c*c) and (b*b<a*a+c*c) and (c*c<a*a+b*b) then
            writeln('İtibucaqlı')
        else
            if (a*a>b*b+c*c) or (b*b>a*a+c*c) or (c*c>a*a+b*b)
then
                writeln('Korbucaqlı')
            else
                writeln('Düzbucaqlı');
        end
    else
        writeln('Belə üçbucaqlı yoxdur');
    readln;
end.
```

### C++

#### //Program A45

```
#include <iostream>

using namespace std;

int main()
{
    double a, b, c;

    cin >> a >> b >> c;

    if (a < b + c && b < a + c && c < a + b) {
        if (a*a<b*b+c*c && b*b<a*a+c*c && c*c<a*a+b*b)
            cout << "İtibucaqlı\n";
    }
```

```

        else {
            if (a*a>b*b+c*c || b*b>a*a+c*c || c*c>a*a+b*b)
                cout << "Korbucaqlı\n";
            else
                cout << "Düzbucaqlı\n";
        }
    }
else
    cout << "Belə üçbucaqlı yoxdur\n" << endl;

return 0;
}

```

#### A46

Biz  $a$  rəqəmini ədədin sağına əlavə etməyi bilirik:  $m := m*10 + a$ ;

Alqoritm də buradan alınır: ədədin rəqəmlərini soldan ayırıraq və əgər növbəti rəqəm təkdirsə, onu sağdan  $m$  ədədinə əlavə edirik. Ancaq proqramın başlanğıcında  $m$ -ə 0 mənimsətməyi unutmayın!

#### Pascal

##### Program A46;

```

var
    n, m, a: integer;
begin
    readln(n);

    m := 0;

    a := n div 1000;
    if (a mod 2 <> 0) then
        m := m*10 + a;
    a := (n div 100) mod 10;
    if (a mod 2 <> 0) then
        m := m*10 + a;

    a := (n div 10) mod 10;
    if (a mod 2 <> 0) then
        m := m*10 + a;

    a := n mod 10;
    if (a mod 2 <> 0) then
        m := m*10 + a;
    writeln(m);
    readln;
end.

```



## C++

### //Program A46

```
#include <iostream>

using namespace std;

int main()
{   int n, m = 0, a, b, c, d;

    cin >> n;

    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    d = (n / 1000) % 10;
    if (d%2 != 0) m = 10*m + d;
    if (c%2 != 0) m = 10*m + c;
    if (b%2 != 0) m = 10*m + b;
    if (a%2 != 0) m = 10*m + a;

    cout<<m<<endl;

    return 0;
}
```

## A47

Bu məsələ 44-cü məsələyə bənzərdir. Əgər ən böyük parçanın uzunluğu qalan üç parçanın uzunluqları cəmindən kiçikdirsə, onda bu parçalardan dördbucaqlı düzəltmək olar. Proqramlaşdırmanı asanlaşdırmaq üçün biz bu fikri də başqa ifadə ilə əvəz edəcəyik: əgər hər bir parçanın uzunluğu qalan üç parçanın uzunluqları cəmindən kiçikdirsə, onda bu parçalardan dördbucaqlı düzəltmək olar. Proqramda da məhz bu şərt yoxlanılır.

## Pascal

### Program A47;

```
var
    a, b, c, d: real;

begin
    readln(a, b, c, d);
    if (a < b + c + d) and (b < a + c + d) and
        (c < a + b + d) and (d < a + b + c) then
        writeln('YES')
    else
        writeln('NO');

    readln;
end.
```

## C++

### //Program A47

```
#include <iostream>

using namespace std;

int main()
{ double a, b, c, d;

  cin >> a >> b >> c >> d;

  if (a < b + c + d && b < a + c + d &&
      c < a + b + d && d < a + b + c)
    cout << "YES\n";
  else
    cout << "NO\n";
  return 0;
}
```

## A48

Verilmiş natural ədədin başqa bir natural ədədin kvadratı olub-olmadığını müəyyənləşdirmək üçün biz belə bir alqoritmdən istifadə edəcəyik. Verilmiş natural ədəddən kvadrat kök alacağıq (nəticə həqiqi ədəd olacaq). Onu tam ədədədək yuvarlaşdıracağıq. Əgər yeni alınan ədədin kvadratı ilkin verilmiş ədədə bərabər olarsa, deməli, verilmiş natural ədəd başqa bir natural ədədin kvadratıdır.

Məsələn:  $\sqrt{49}$  ədədinin yuvarlağı 7-yə bərabərdir və  $7^2 = 49$ .

$\sqrt{50}$  ədədinin yuvarlağı 7-yə bərabərdir, ancaq  $7^2 \neq 50$ .

$\sqrt{48}$  ədədinin yuvarlağı 7-yə bərabərdir, ancaq  $7^2 \neq 48$ .

**Pascal programçıları üçün qeyd.** Pascal dilində kvadrat kökalma **sqrt**, yuvarlaqlaşdırma isə **round** funksiyası ilə yerinə yetirilir.

**C++ programçıları üçün qeyd.** C++ dilində kvadrat kökalma **sqrt** funksiyası ilə yerinə yetirilir, tam ədədədək yuvarlaqlaşdırmanı isə biz belə həyata keçirəcəyik: `a = (int) sqrt(n+0.5);` başqa sözlə, azacıq böyük ədəd alırıq və onun kəsr hissəsini atırıq. İstənilən ifadənin önündə `(int)` yazılması kəsr hissəni uzaqlaşdırır.

## Pascal

### Program A48;

```
var
  n, a: integer;

begin
  readln(n);
```

```

a := round(sqrt(n));
if (a*a = n) then
    writeln('YES')
else
    writeln('NO');

readln;
end.

```

## C++

### //Program A48

```

#include <iostream>
#include <cmath>

using namespace std;

int main()
{
    int n, a;

    cin >> n;

    a = (int)sqrt(n+0.5);
    if (a*a == n)
        cout << "YES\n";
    else
        cout << "NO\n";

    return 0;
}

```

## A49

Maksimal ədədi almaq üçün ədədin rəqəmlərini elə düzməliyik ki, hər sonrakı rəqəm əvvəlkindən böyük olmasın. Məsələn: 539 ədədinin rəqəmlərindən düzəldilmiş maksimal ədəd 953 olacaq, 377 ədədinin rəqəmlərindən düzəldilmiş maksimal ədəd isə 773 olacaq. Ona görə də məsələnin həll alqoritmini belə ifadə etmək olar:

1. Bütün rəqəmləri ayırırıq:  $a$  – vahidlərin sayı,  $b$  – onluqların sayı,  $c$  – yüzliklərin sayı.
2. Əgər  $b$   $a$ -dan və  $c$ -dən böyükdürsə, onda  $a$  və  $b$ -nin qiymətlərini qarşılıqlı əvəzləyirik, yəni  $a$  ən böyük olur. Əks halda əgər  $c$   $a$ -dan və  $b$ -dən böyükdürsə, onda  $a$  və  $c$ -nin qiymətlərini qarşılıqlı əvəzləyirik, yəni bu halda da  $a$  ən böyük olur.
3. İndi  $b$  və  $c$ -ni müqayisə edirik. Əgər  $b < c$ , onda  $b$  və  $c$ -nin qiymətlərini qarşılıqlı əvəzləyirik. Bu əməliyyatlardan sonra  $a \leq b \leq c$  alırıq. İndi tələb olunan üçrəqəmli ədədi belə ifadə etmək olar:  $m := 100*a + 10*b + c$ .

## Pascal

### Program A49;

```
var
    n, m, a, b, c, r : integer;

begin
    readln(n);
    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;

    if (b > a) and (b > c) then begin
        r := a;
        a := b;
        b := r;
    end
    else
        if (c > a) and (c > b) then begin
            r := a;
            a := c;
            c := r;
        end;

        if (b < c) then begin
            r := b;
            b := c;
            c := r;
        end;

        m := 100*a + 10*b + c;
        writeln(m);
        readln;
    end.
```

## C++

### //Program A49

```
#include <iostream>

using namespace std;

int main()
{
    int n, m, a, b, c, r;
    cin >> n;
    m = 0;
    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
```

```

    if (b > a && b > c){
        r = a; a = b; b = r;
    }
    else
        if (c > a && c > b) {
            r = a; a = c; c = r;
        }

    if (b < c) {
        r = b; b = c; c = r;
    }

    m = 100*a + 10*b + c;
    cout << m << endl;

    return 0;
}

```

### A50

Bu məsələ 49-cu məsələ kimidir. Sadəcə, proqramda böyükdür (>) işarəsinin yerinə kiçikdir (<) işarəsi qoymaq lazımdır.

### Pascal

```

Program A50;

var
    n, m, a, b, c, r: integer;
begin
    readln(n);

    a := n mod 10;
    b := (n div 10) mod 10;
    c := (n div 100) mod 10;

    if (b < a) and (b < c) then begin
        r := a; a := b; b := r;
    end
    else
        if (c < a) and (c < b) then begin
            r := a; a := c; c := r;
        end;

    if (b > c) then begin
        r := b; b := c; c := r;
    end;

    m := 100*a + 10*b + c;
    writeln(m);
    readln;
end.

```

C++

**//Program A50**

```
#include <iostream>

using namespace std;

int main()
{
    int n, m, a, b, c, r;

    cin >> n;

    m = 0;
    a = n % 10;
    b = (n / 10) % 10;
    c = (n / 100) % 10;
    if (b < a && b < c) {
        r = a; a = b; b = r;
    }
    else
        if (c < a && c < b) {
            r = a; a = c; c = r;
        }

    if (b > c) {
        r = b; b = c; c = r;
    }

    m = 100*a + 10*b + c;
    cout << m << endl;

    return 0;
}
```



## V BÖLÜM

# OLİMPIADANIN İKİNCİ (RAYON, ŞƏHƏR) TURUNUN MƏSƏLƏLƏRİ

## 5.1. MƏSƏLƏLƏRİN ŞƏRTLƏRİ

Bu bölümə 2000–2014-cü illərdə olimpiadanın ikinci (rayon, şəhər) mərhələsində təqdim olunmuş məsələlər verilib. Hər bir məsələ aşağıdakı kimi kodlaşdırılıb: başlanğıcdakı “T2” yazısı məsələnin ikinci tura aid olduğunu bildirir, sonra olimpiadanın keçirildiyi tədris ili, sonda isə məsələnin sıra nömrəsi verilir. Koddan sonra isə məsələnin adı yazılacaq. Məsələn, “T2-2009-01. Palindromlar” kodlaşdırması məsələnin 2008/2009 tədris ilində keçirilmiş ikinci turun 1-ci məsələsi olduğunu göstərir.

### T2-2000-01. Mətnin ixtisarı

**N** simvoldan ibarət ( $N < 20000$ ) **A** mətni verilib. Mətnin təsviri:

**Basic:** DIM \$ (20000) ;

**Pascal:** A: array[1..20000] of char

Mətnə sözlər bir-birindən ya boşluq, ya da vergül işarəsi ilə, cümlələr isə nöqtə ilə ayrılıb. **M** ( $M < 200$ ) simvoldan ibarət olan və **A** mətni kimi təsvir edilmiş başqa bir **B** mətni də verilib.

**A** mətnindən ayrı-ayrı sözləri, vergülləri, yaxud nöqtələri silməklə **B** mətninin alınıb-alınmadığını müəyyən edən proqram tərtib edin (sözlərin yerini dəyişmək olmaz).

## T2-2000-02. Sıralama (bax: T2-2009-04)

Verilmiş  $N$  natural ədədi üçün 1-dən böyük olmayan bütün sadə, ixtisar olunmayan kəsrlər ardıcılığını artan sıra ilə düzən proqram yazın.

**Örnək:**  $N = 7$  üçün alınan ardıcılıq belə olacaq:

1/7, 1/6, 1/5, 1/4, 2/7, 1/3, 2/5, 3/7, 1/2, 4/7, 3/5, 2/3, 5/7, 3/4, 4/5, 5/6, 6/7, 1/1

## T2-2000-03. Vahidə çevrilən ədəd

İstənilən natural ədəd götürüb rəqəmlərinin kvadratlarını cəmləyək. Alınan yeni natural ədəd üzərində də həmin əməliyyatı aparaq. Bu prosesin sonlu sayda təkrarlanmasından sonra nəticədə birrəqəmli ədəd alarıq. Məsələn, 324 ədədi üçün proses belə olacaq:

$$a_1 = 324$$

$$a_2 = 3 \cdot 3 + 2 \cdot 2 + 4 \cdot 4 = 29$$

$$a_3 = 2 \cdot 2 + 9 \cdot 9 = 85$$

$$a_4 = 8 \cdot 8 + 5 \cdot 5 = 89$$

$$a_5 = 8 \cdot 8 + 9 \cdot 9 = 145$$

$$a_6 = 1 \cdot 1 + 4 \cdot 4 + 5 \cdot 5 = 42$$

$$a_7 = 4 \cdot 4 + 2 \cdot 2 = 20$$

$$a_8 = 2 \cdot 2 + 0 \cdot 0 = 4 \dots (\text{prosesin sonu})$$

[1, 1000] intervalında yerləşən natural ədədlər içərisindən yuxarıda göstərilən prosesin 1-lə nəticələndiyi bütün ədədləri aşkarlayan proqram tərtib edin.

## T2-2000-04. Cədvəl

$M$  sətir və  $N$  sütundan ibarət ( $M \times N$ ) ikiölçülü tam ədədlər cədvəli verilib. Sətirlər artan sıra ilə çeşidlənib.

Bütün sətirlərdə iştirak edən ədədləri tapan və onları ekrana çıxaran proqram yazın.

**Örnək:**  $M = 4$ ,  $N = 5$  üçün

2	3	5	7	9
3	6	9	11	13
1	3	7	9	10
3	3	4	8	9

cədvəlinde belə ədədlər 3 və 9-dur.

**Qeyd.** 2000/2001 tədris ilində keçirilmiş rayon turunun məsələlərini əldə edə bilmədiyimizdən onları başqa məsələlərlə əvəzləmək qərarına gəldik.



## T2-2001-01\*. Erişilməz xanalar

“Eurovision-2012” mahnı müsabiqəsi ilə əlaqədar Bakı küçələrinin birinin ortasında küçə boyunca bir sırada kvadrat formalı  $N$  sayda yastı daş yerləşdirdilər. Daşların hər birinin üzərində ingilis əlifbasının bir hərfi yazılıb. Daşlar korlana bilər və bəzən onları dəyişmək lazımdır. Bu məqsədlə yalnız  $M$  sayda düzbucaqlı formalı (bir sıradan ibarət) müxtəlif daş nümunələri var. Nümunələr içərisində tam eyniləri ola bilər. Belə daşın  $i$ -ci nümunəsi  $L_i$  hərfdən (yəni  $L_i$  kvadratdan) ibarətdir. Düzbucaqlı daşı fırlatmaq, yaxud hissələrə bölmək olmaz. O yalnız elə yerləşdirilə bilər ki, daşın üzərindəki hərflər ilkin vəziyyətdəki ardıcılıqda olsun. Daşlar üst-üstə qoyula bilər və nümunədəki daşlardan dəfələrlə istifadə etmək olar.

Əgər küçənin hər hansı kvadratı daşla örtülə bilmirsə, həmin xana *erişilməz* hesab olunur. Erişilməz xanaların sayını hesablayın.

### Giriş verilənləri

Birinci sətirdə küçənin uzunluğunu bildirən  $N$  ( $1 \leq N \leq 300\,000$ ) müsbət tam ədədi olur. İkinci sətirdə küçə boyunca ardıcıl düzülmüş hərfləri bildirən  $N$  sayda adi hərf (ingilis əlifbasının) yerləşir. Üçüncü sətirdə nümunə üçün olan düzbucaqlı daş lövhələrin sayını göstərən  $M$  ( $1 \leq M \leq 5000$ ) müsbət tam ədədi olur. Sonrakı  $M$  sətirin hər birində  $L_i$  ( $1 \leq L_i \leq 5000$ ) uzunluqlu nümunə lövhənin təsviri verilir. Nümunənin təsviri ingilis əlifbasının adi hərflərindən ibarətdir.

### Çıxış verilənləri

Bir sətirdə erişilməz xanaların sayını bildirən ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
6 abcbab 2 cb cbab	2
4 abab 2 bac baba	4
6 abcabc 2 abca cab	1

## T2-2001-02\*. Populyarlıq

Bakıda keçirilən reklam kampaniyasında müxtəlif zaman intervallarının televiziya reytingini, yəni müəyyən zaman intervalında televizora nə qədər insanın baxdığını müəyyənləşdirmək qərara alındı. Bu məqsədlə  $N$  sayda insan sorğudan keçirildi. Bu insanların hər biri aşağıdakı formatda olan iki verilənlə xarakterizə olunur:

**HH:MM:SS - HH:MM:SS**, burada *HH* – saati, *MM* – dəqiqəni, *SS* – saniyəni bildirir.

Birinci verilən insanın televizora baxmağa başladığı, ikinci verilən isə bitirdiyi vaxtı göstərir. Nəzərə almaq lazımdır ki, televizora baxmağa gecəyarısı başlamaq (məsələn, 23:45:30) və növbəti gün (məsələn, 01:15:00) bitirmək olar. Bütün verilənlər toplandıqdan sonra reklam mütəxəssisləri onları təhlil etmək üçün toplaşdılar.

Hər hansı *saniyənin* “*populyarlığını*” həmin saniyə ərzində televizora baxan insanların ümumi sayı kimi müəyyən edək. Bundan başqa, müəyyən *zaman intervalının* “*populyarlığını*” bu intervalda olan saniyələrin “populyarlıqları” cəminin intervalın uzunluğuna nisbəti kimi təyin edək.

Reklam mütəxəssisləri üçün xüsusi maraq kəsb edən verilmiş zaman intervallarının  $Q$  populyarlığını hesablayın.

### Giriş verilənləri

Birinci sətirdə insanların sayını göstərən  $N$  ( $1 \leq N \leq 100\,000$ ) tam ədədi verilir. Sonrakı  $N$  sətirdə yuxarıda təsvir olunan formatda iki verilən olur:

**HH:MM:SS - HH:MM:SS** ( $0 \leq HH \leq 23, 0 \leq MM \leq 59, 0 \leq SS \leq 59$ ).

Sonrakı sətirdə reklam mütəxəssislərini maraqlandıran zaman intervallarının sayını göstərən  $Q$  ( $1 \leq Q \leq 100\,000$ ) tam ədədi yerləşir. Sonrakı  $Q$  sətirdə yuxarıda göstərilən formatda zaman intervalları verilir.

### Çıxış verilənləri

Verilmiş  $Q$  sayda zaman intervalının hər biri üçün onun populyarlığı ayrıca sətirdə göstərilir. Cavabı nüsxədə göstərildiyi kimi, nöqtədən sonra 6 işarəyə qədər yuvarlaqlaşdırmaqla çıxarmaq lazımdır.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5	2.000000
00:00:00 - 00:00:01	1.000000
00:00:01 - 00:00:03	2.000000
00:00:00 - 00:00:02	1.200000
00:00:05 - 00:00:09	1.400000
00:00:06 - 00:00:06	

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 00:00:00 – 00:00:03 00:00:07 – 00:00:09 00:00:06 – 00:00:06 00:00:05 – 00:00:09 00:00:00 – 00:00:09	
3 00:00:00 – 10:00:00 10:00:00 – 00:00:00 01:01:01 – 02:02:02 4 00:00:00 – 23:59:59 23:59:59 – 23:59:58 23:59:59 – 23:59:59 08:34:43 – 12:22:17	1.042407 1.042407 1.000000 1.000073

### T2-2001-03\*. Ən qısa yol

Yeniləşmiş Bakıda uzun mərkəzi küçələrdən birində yolun bir tərəfində inzibati binalar, qarşı tərəfdə isə ticarət evləri yerləşir. İnzibati binalar arasında, eləcə də ticarət evləri arasında məsafə 50 metrdir. İnzibati binalar 0-dan başlayaraq nömrələnib. 0 nömrəli binada qalan binalara mallar çatdırmaq üçün mərkəzi ofis yerləşir. Bu gün səhər sifariş daxil olub ki, küçənin bir obyektindən başqa bir obyektinə **N** sayda mal çatdırmaq lazımdır. Daşımanın rəhbəri Alpay öz yoluna 0 nöqtəsində başlayır, küçənin sonunda **M** nöqtəsində bitirir və götürdüyü malları bu nöqtələr arasında paylaşacaq. Hər dəfə daşınan mallar o qədər olur ki, onların hamısı Alpayın maşınına yerləşir.

Əlbəttə, Alpay elə hərəkət etmək istəyir ki, mümkün qədər az məsafə qət etsin. Məsələn, əgər Alpay bu gün işini 10 nömrəli binada bitirməlidirsə və onun 2 nöqtəsindən 8 nöqtəsinə, 6 nöqtəsindən 4 nöqtəsinə mallar daşımaq üçün sifarişləri varsa, o, belə hərəkət edə bilər: yoluna 0 nöqtəsindən başlayır; 2 nöqtəsində malı götürüb 6 nöqtəsinə gedir; 6 nöqtəsində malı götürüb 4 nöqtəsinə qayıdır və malı verir; sonra 8 nöqtəsinə gedib malı verir və 10 nöqtəsinə gedir. Biz birinci testin həlli üçün ssenarini təsvir etdik.

Alpayın bütün sifarişləri yerinə yetirərək **M** nöqtəsinə çatması üçün tələb olunan ən az məsafəni tapan proqram yazın.

### Giriş verilənləri

Birinci sətirdə **N** və **M** ( $N \leq 300\,000$ ,  $3 \leq M \leq 10^9$ ) tam ədədləri olur. Növbəti **N** sətirin hər birində iki tam ədəd olur; bu ədədlər bu gün yerinə yetirilməsi zəruri olan hər bir sifarişin başlanğıc və son məntəqələrini göstərir və 0-dan **M**-dək dəyişə bilər.

### Çıxış verilənləri

Alpayın keçəcəyi ən kiçik məsafəni bildirən bir tam ədəddən ibarət olur.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2 10 2 8 6 4	700
8 15 1 12 3 1 3 9 4 2 7 13 12 11 14 11 14 13	1350

### T2-2001-04\*. Bataqlıqdan keçmək

Alpay bataqlıqdan yalnız torpaq talalarının birindən digərinə tullanmaqla keçə bilər. Alpay ən çoxu **K** sayda talanın üzərindən tullana bilər (o, ən çoxu **K**-cı talaya düşə bilər). Alpay ayağını talaya basan kimi tala ya suya batır, ya da batmır. Alpay hər bir tala haqqında məlumata malikdir – həmin tala nə qədər suya bata bilər.

Alpaya ümumilikdə mümkün qədər az suya batmaqla yolu tapmaqda kömək edin. Alpayın yolu bataqlığın bir tərəfində başlayır, o biri tərəfində isə başa çatır.

### Giriş verilənləri

Birinci sətirdə iki tam ədəd olur: talaların sayını bildirən **N** ( $2 \leq N \leq 10^5$ ) ədədi və tullanmaların maksimal uzunluğunu bildirən **K** ( $1 \leq K < N$ ) ədədi.

İkinci sətirdə talaları xarakterizə edən və bir-birindən boşluq simvolu ilə ayrılmış **N** sayda tam ədəd olur. Ədədin müsbət olması talanın suya batmamasını, mənfi olması isə göstərilən kəmiyyət qədər suya batmasını bildirir.

### Çıxış verilənləri

Alpayın toplam olaraq ən az suya düşməsinə bildirən bir müsbət ədəddən (yaxud sıfırdan) ibarət olur.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
6 2 3 5 -3 4 3 7	0
6 2 3 5 -3 -4 3 7	3

## T2-2002-01. Ardıcılıq

$2N$  sayda natural ədəddən ibarət ardıcılıq verilmişdir. Əvvəlcədən məlumdur ki, bu ardıcılığın bütün ədədlərini elə cütlüklərə ayırmaq olur ki, həmin cütlüklərin hər birində olan ədədlərin cəmi bərabərdir. Məsələn, 99, 23, 77, 1 ədədlər ardıcılığında  $1 + 99 = 77 + 23$  olduğundan həmin ardıcılığın  $(1; 99)$  və  $(77; 23)$  kimi iki cütlüyə ayırmaq olar.

Belə ardıcılığın hasilləri bərabər olan cütlüklərə ayırmağın mümkün olub-olmadığını müəyyən edən proqram yazın.

**Giriş verilənləri** (SEQ.IN faylından və ya klaviaturadan)

Birinci sətirdə ardıcılığın  $2N$  hədlərinin sayı, sonrakı  $2N$  sətirdə isə bu ardıcılığın 1-dən  $10^9$ -a qədər natural ədədlər olmaqla elementləri yerləşir ( $1 < N < 50$ ).

**Çıxış verilənləri** (SEQ.OUT faylına və ya ekrana)

Giriş verilənlərindəki ardıcılığın hasilləri bərabər olan cütlüklərə ayırmaq mümkün olarsa, cavab 1, əks halda 0 olur.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 99 23 77 1	0

## T2-2002-02. “Antisadə” ədəd

Müsbət  $N$  ədədinin bölənlərinin sayı  $N$ -dən kiçik istənilən ədədin bölənlərinin sayından çoxdursa, onda həmin ədədə “antisadə” ədəd deyəcəyik. Məsələn: 1, 2, 4, 6, 12, 24 ədədləri “antisadə” ədədlərdir.

Verilmiş  $N$ -ə görə  $N$ -i aşmayan ən böyük “antisadə” ədədi tapan proqram tərtib edin.

**Giriş verilənləri** (ANTI.IN faylından və ya klaviaturadan)

$N$  ədədi ( $1 < N < 2000000000$ ).

**Çıxış verilənləri** (ANTI.OUT faylına və ya ekrana)

$N$ -i aşmayan ən böyük “antisadə” ədəd.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1000	840

### T2-2002-03. Armstrong ədədi

**N** rəqəmli natural ədədin rəqəmlərinin **N**-ci qüvvətlərinin cəmi həmin ədədin özünə bərabər olarsa, (məsələn,  $153 = 1^3 + 5^3 + 3^3$  kimi), həmin ədədə *Armstrong ədədi* deyilir.

3-, 4- və 5-rəqəmli Armstrong ədədlərini tapmaq üçün proqram tərtib edin.

**Giriş verilənləri:** yoxdur.

**Çıxış verilənləri** (NUMBER.OUT faylına və ya ekrana)

Tapılan bütün Armstrong ədədlərinin hər biri ayrıca sətirdə verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
	153 370 371

### T2-2002-04. Paraleloqram

Müstəvi üzərində təpə nöqtələrinin koordinatları saat əqrəbinin hərəkəti istiqamətində götürüldükdə **X1, Y1, X2, Y2, X3, Y3, X4, Y4** olan paraleloqram və koordinatları **X, Y** olan nöqtə verilmişdir. Həmin nöqtənin paraleloqramın daxilində və ya xaricində yerləşdiyini müəyyən edən proqram tərtib edin.

**Giriş verilənləri** (PAR.IN faylından və ya klaviaturadan)

Bir sətirdə aralarında boşluq və ya vergül olmaqla 10 ədəd – 5 nöqtənin koordinatları verilir: X, Y, X1, Y1, X2, Y2, X3, Y3, X4, Y4.

**Çıxış verilənləri** (PAR.OUT faylına və ya ekrana)

Əgər nöqtə paraleloqramın daxilində və ya onun tərəfi üzərində olarsa, cavab 1, xaricində olarsa isə cavab 0 olur.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5, 6, 3, 5, 6, 11, 9, 8, 6, 2	1
-1, 3, 3, 5, 6, 11, 9, 8, 6, 2	0

### T2-2002-05. Hesabi ifadə

Açılan və bağlanan dairəvi mötərizələrdən ibarət **2N** sayda simvollar ardıcılığı verilmişdir. Bu ardıcılığa rəqəm və hesab əməlləri əlavə etməklə düzgün hesabi ifadə alınmasının mümkünlüyünü müəyyənləşdirən proqram tərtib edin.

### Giriş verilənləri (HESAB.IN faylından və ya klaviaturadan)

Birinci sətirdə  $N$  ədədi, ikinci sətirdə isə  $2N$  sayda bir-biri ilə bir boşluq simvolu ilə ayrılmış açılan və bağlanan dairəvi mötərizələr verilir.

### Çıxış verilənləri (HESAB.OUT faylına və ya ekrana)

Əgər verilən ardıcılıqdan düzgün hesabi ifadə almaq mümkün olarsa, cavab 1, əks halda isə 0 olur.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
$\begin{matrix} 4 \\ ( ( ) ( ( ) ) ) \end{matrix}$	1
$\begin{matrix} 4 \\ ( ( ) ( ) ) ) \end{matrix}$	0

### T2-2003-01. Hissələrə ayırma

$N$  ( $N < 30000$ ) natural ədədini  $M$  ( $M < 20$ ) sayda *təxmini bərabər* tam hissələrin cəmi şəklində göstərin. Bir-birindən ən çoxu 1 vahid fərqli olan ixtiyari iki hissəni təxmini bərabər hissə adlandırırıq.

### Giriş verilənləri (IN1.TXT faylında və ya klaviaturadan)

Birinci sətirdə  $N$  ədədi, ikinci sətirdə isə  $M$  ədədi verilir.

### Çıxış verilənləri (OUT1.TXT faylına və ya ekrana)

$N$  ədədinin ayrıldığı  $M$  sayda hissələr azalmayan sıra ilə bir sətirdə boşluq işarəsi ilə ayrılmaqla verilməlidir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
$\begin{matrix} 13 \\ 4 \end{matrix}$	$\begin{matrix} 3 & 3 & 3 & 4 \end{matrix}$

### T2-2003-02. Sadə çəki daşları

1 q, 2 q, ...,  $N$  q ( $N < 5000000$ ) kütləli çəki daşları verilmişdir. Bu çəki daşlarını maksimal mümkün sayda elə cütliklərə bölün ki, hər cütlikdə olan daşların çəkirlərinin cəmi sadə ədədlə ifadə olunsun. Proqramı tərtib edin.

### Giriş verilənləri (IN2.TXT faylından və ya klaviaturadan)

Bir sətirdə  $N$  ədədi verilir.

### Çıxış verilənləri (OUT2.TXT faylına və ya ekrana)

Tapılan cütliklərin siyahısı hər cütlik bir sətirdə olmaqla ekrana və ya çıxış faylına verilir. Cütliklərdəki ədədlər boşluq işarəsi ilə ayrılır.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
7	1 6 7 4 5 2

### T2-2003-03. Neçə sikkə?

Məmməd sandığa hər hansı sayda (*a*) sikkə qoydu. İkinci il o, sandıqdan müəyyən sayda (*b*) sikkə çıxardı. Üçüncü ildən başlayaraq hər il sandığa iki il əvvəl orada olan qədər sikkə əlavə etdi. Əgər *X* ildən sonra sandıqda *Y* sayda sikkə olarsa, birinci və ikinci il sandıqda neçə sikkə var idi?

**Açıqlama:** əgər birinci il sandığa 5 sikkə qoyulsa və ikinci il 3 sikkə çıxarılsa, birinci ildən başlayaraq sandıqda 5, 2, 7, 9, 16, 25... sikkə olacaq. Tərtib edilmiş proqram klaviaturadan boşluq işarəsi ilə ayrılmış iki *X* və *Y* ədədinin daxil edilməsini istəməlidir.

**Giriş verilənləri** (klaviaturadan)

*X Y*

**Çıxış verilənləri** (OUT3.TXT faylına və ya ekrana)

*a b*

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
9 107	5 2

### T2-2003-04. Maksimal ardıcıl cəm

$x_1, x_2, x_3, \dots, x_n$  ( $n < 50$ ) tam ədədlər ardıcılığı verilmişdir. Ardıcıl gələn ədədlərin maksimal cəmini tapın.

**Örnek**

2, 4, -7 -3, 4, 1, 3, -2, 4, 2, -25, -4, 4, 3, 3, 1, -4

Ardıcılığında maksimal cəm 12-yə bərabərdir və o, 5-cidən 10-cuya kimi ədədlərin cəmindən ibarətdir.

**Giriş verilənləri** (IN4.TXT faylından və ya klaviaturadan)

*n*

$x_1 \ x_2 \ x_3 \ \dots \ x_n$

**Çıxış verilənləri** (OUT4.TXT faylına və ya ekrana)

Maksimal cəm



Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
17 2 4 -7 -3 4 1 3 -2 4 2 -25 -4 4 3 3 1 -4	12

## T2-2003-05. Yerlərini dəyişmək

$N \times N$  ( $N \leq 10$ ) kvadrat cədvəlinin hər sətirindəki maksimal və minimal elementlərin yerini dəyişən program yazın.

### Örnek

2	5	-4	1	-2
7	3	2	-3	4
1	5	-5	4	8
4	-3	6	9	4
3	-5	5	8	6

5×5 kvadrat cədvəlinin hər sətirindəki maksimal və minimal elementlərin yerini dəyişdikdə

2	-4	5	1	-2
-3	3	2	7	4
1	5	8	4	-5
4	9	6	-3	4
3	8	5	-5	6

cədvəli alınar.

### Giriş verilənləri (IN5.TXT faylından və ya klaviaturadan)

Birinci sətirdə bir  $N$  ədədi yerləşir. Sonrakı  $N$  sayda sətirin hər birində isə kvadrat cədvəlin sətirləri olan  $N$  ədəd olur.

### Çıxış verilənləri (OUT5.TXT faylına və ya ekrana)

Çıxışa çevrilmiş cədvəl verilir.

## T2-2004-01. Ötürülmə

Bir (1) və sıfırlardan (0) ibarət ardıcılıq şəklində olan məlumat rabitə kanalı ilə ötürülür. Siqnalların düzgün qəbulunun etibarlılığı üçün hər bir siqnal üç dəfə ardıcıl (1 əvəzinə 111, 0 əvəzinə 000) göndərilir.

Alınan məlumat əsasında ilkin məlumatı bərpa edən program tərtib etmək lazımdır. Ötürülmə vaxtı yanlışlıqlar da ola bilər. Ona görə də alınan hər bir “üçlük” hansı simvol çoxdursa, “üçlük” həmin simvolla əvəzlənməlidir.

### Giriş verilənlərinin formatı

Giriş verilənləri yalnız “0” və “1” simvollarından ibarət bir sətirdir. Sətrin uzunluğu 3-ün misli sayda olub 2-dən böyük və 255-dən kiçikdir.

### Çıxış verilənlərinin formatı

Çıxışa bərpa edilmiş məlumat bir sətirdə verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
110111010001	1100

### T2-2004-02. Bloklar (bax: T2-2009-02)

Hər bir elementi 0, 1, 5 və ya 11 olan  $A(N, N)$  cədvəli – ikiölçülü massivi ( $2 < N < 50$ ) verilib. Həmin cədvəldə bütün elementləri müxtəlif olan 4 elementli blokların (blokun elementləri:  $a_{i,j}$ ,  $a_{i+1,j}$ ,  $a_{i,j+1}$ ,  $a_{i+1,j+1}$ ) sayını hesablayan proqram tərtib edin. Tapılmış blokların orta elementləri də ola bilər.

### Giriş verilənlərinin formatı

Birinci sətirdə bir  $N$  ədədi yerləşir. Sonrakı  $N$  sayda sətirin hər birində isə kvadrat cədvəlin sətirləri olan  $N$  ədəd olur.

### Çıxış verilənlərinin formatı

Çıxışa bir ədəd – axtarılan blokların sayı verilir.

### T2-2004-03. Say və cəm (bax: T2-2009-03)

$N$  ( $2 \leq N < 10$ ) sayda tam ədəddən ibarət ardıcılıq verilib; ardıcılıqda həm müsbət, həm də mənfi ədədlər var. Bu ardıcılıqdakı mənfi ədədlərin ən böyüyü ilə müsbət ədədlərin ən kiçiyi arasında qalan ədədlərin sayını və cəmini tapın (hər iki ədəd say və cəmə daxil olur).

### Giriş verilənlərinin formatı

1-ci sətirdə bir  $N$  ədədi, 2-ci sətirdə isə  $N$  sayda ədəd – verilmiş ardıcılıq yerləşir.

### Çıxış verilənlərinin formatı

Çıxışa iki ədəd – axtarılan say və cəm verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
10 5 9 -3 4 7 -4 2 8 -5 10	5 6

**T2-2004-04. Yerdəyişmə (bax: T2-2009-05)**

$N$  ( $1 \leq N < 100$ ) sayda natural ədəddən ibarət ardıcılıq verilib. Bu ardıcılığın  $1, 2, 3, \dots, N$  ardıcılığının yerdəyişməsindən alınıb-alınmadığını müəyyən-ləşdirən proqram yazın.

**Nümunə:**  $5, 7, 1, 3, 2, 6, 4$  ardıcılığı  $1, 2, 3, 4, \dots, 7$  ardıcılığından yerdəyişmə yolu ilə alınıb, ancaq  $5, 7, 1, 3, 1, 6, 4$  ardıcılığı  $1, 2, 3, 4, \dots, 7$  ardıcılığının yerdəyişməsindən alınmayıb.

**Giriş verilənlərinin formatı**

Birinci sətirdə yalnız bir  $N$  ədədi yerləşir. İkinci sətirdə isə verilən  $N$  sayda ədəddən ibarət ardıcılıq olur.

**Çıxış verilənlərinin formatı**

Ardıcılıq yerdəyişmədən alınbsa, çıxışa “YES”, əks halda, “NO” sözü verilir.

**T2-2004-05. Sezar şifri**

Yalnız Latin əlifbasının baş hərfləri və boşluq simvollarından ibarət olan, uzun-luğu 255 simvoldan çox olmayan mətn verilib. Verilmiş açara görə mətndəki hər bir hərfi əlifba sırası üzrə ( $A B C D E F G H I J K L M N O P Q R S T U V W X Y Z$ ) həmin hərfdən açar qədər məsafədə yerləşən hərflə əvəzləyən proqram tərtib edin. Məsələn, əgər açar 2-yə bərabədirsə, onda  $A$  hərfi  $C$  ilə,  $B$  hərfi  $D$  ilə,  $C$  hərfi  $E$  ilə, ...,  $Z$  hərfi isə  $B$  ilə əvəz olunacaq.

**Giriş verilənlərinin formatı**

Birinci sətirdə yalnız açarın qiyməti olan  $K$  ədədi, ikinci sətirdə isə verilmiş mətn yerləşir.

**Çıxış verilənlərinin formatı**

Bir sətirdə şifrlənmiş mətn verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3 A GOOD STUDENT	D JRRG VWXGHQW

**T2-2007-01. İkinci qüvvətləri**

$N$  ( $1 \leq N \leq 10^6$ ) ədədi daxil edilir. Həmin ədəd 2-nin qüvvətidirsə, YES, əks halda NO sözünü çap edən proqram tərtib edin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
8	YES
22	NO

### T2-2007-02. Bölünmə

Üç  $N$ ,  $M$  və  $K$  natural ədədləri daxil edilir.  $N$ -dən  $M$ -dək ( $M$  özü də daxil olmaqla) olan ədədlərin içərisindən rəqəmlərinin cəmi  $K$ -ya qalıqsız bölünən ədədlərin sayını tapın.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1 100 3	33
10 22 4	3

### T2-2007-03. Dovşan

Dovşan uzunluğu  $N$  olan yolda yalnız irəliyə tullana bilir. Dovşanın tullanış məsafəsinin uzunluğu  $K$ -dan ( $1 \leq K$ ,  $N \leq 100$ ) böyük deyildir. Dovşanın bütün yolu əvvəldən axıradək tullanaraq keçməsi üçün müxtəlif variantların sayını tapın.

**Məsələn:**  $N = 3$ ,  $K = 2$  üçün dovşanın mümkün tullanışları bunlardır: 1, 1, 1; 1, 2; 2, 1. Bu halda cavab 3 olacaq.

#### Giriş verilənləri

$N$  və  $K$  ədədləridir.

#### Çıxış verilənləri

Çıxışa dovşanın bütün yolu tullanaraq keçməsi üçün variantların sayını göstərən bir ədəd verilir.

### T2-2007-04. Tənlik

$A+B=C$  tənliyi verilmişdir; burada  $A$ ,  $B$  və  $C$  – onluq yazılışında bəzi rəqəmləri ? işarəsi ilə əvəz olunmuş mənfi olmayan tam ədədlərdir. Məsələn:  $?2+34=4?$ .

? işarəsinin yerinə elə rəqəmləri qoymaq tələb olunur ki, ya bu tənlik doğru bərabərliyə çevrilsin, ya da müəyyən edilsin ki, belə bərabərlik mümkün deyil. Mümkün həllərdən yalnız birini tapın.

#### Giriş verilənləri

Uzunluğu 40 simvoldan çox olmayan və boşluq simvolunun olmadığı tənlikdir.

#### Çıxış verilənləri

Verilmiş tənlikdə ? işarəsinin yerinə rəqəmlər yazılması nəticəsində alınan doğru bərabərlik və ya belə bərabərlik mümkün olmadıqda “No solution” sözü olur.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
??2?4+9?=355	00264+91=355

## T2-2007-05. Cədvəl

Soldan sağa və yuxarıdan aşağıya 1-dən  $10^8$ -dək ardıcıl natural ədədlərlə doldurulmuş  $10^4 \times 10^4$  damadan ibarət kvadrat cədvəl verilmişdir. Daxil edilən hər hansı ədədə görə bu cədvəldə həmin ədədin olduğu damanın qonşu damalarındakı ədədləri tapan proqram yazın. Verilmiş dama ilə ortaq tərəfləri olan damalar *qonşu damalar* hesab edilir.

### Giriş verilənləri

Bir  $N$  ( $1 \leq N \leq 10^8$ ) ədədindən ibarətdir.

### Çıxış verilənləri

Proqram hər bir sətirdə bir ədəd olmaqla artma sırası ilə verilmiş damanın qonşu damalarındakı ədədləri verməlidir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
12345	2345 12344 12346 22345

## T2-2008-01. Qadağan olunmuş ədədlər

Ölkələrdən birində yazılışında ardıcıl üç 6 rəqəmi və ya 13 ədədi olan bütün ədədlərdən istifadə qadağan edilmişdir. Məsələn, avtomobillər üçün 2666 və ya 7134 nömrələrindən istifadə etmək olmaz, ancaq 6266, 7314 və ya 1734 nömrələrindən istifadə etmək olar. **K**-rəqəmli ədədlər içərisindən qadağan edilmiş ədədlərin sayını tapmaq üçün proqram tərtib edin.

### Giriş verilənləri

Girişdə bir  $K$  ( $1 \leq K \leq 1000$ ) tam ədədi daxil edilir.

### Çıxış verilənləri

Çıxışa  $10^{K-1}$  və  $10^K - 1$  aralığında olan qadağan edilmiş ədədlərin sayını bildirən bir ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3	20
2	1

### T2-2008-02. Eyni ədədlər

Verilmiş  $N$  ( $1 \leq N \leq 100$ ) sayda natural ədəd içərisində eyni olan ədədləri tapmaq tələb olunur.

#### Giriş verilənləri

Birinci sətirdə  $N$  ədədi daxil edilir, ikinci sətirdə isə  $N$  sayda ədəd aralarında boşluq simvolu olmaqla verilir.

#### Çıxış verilənləri

Çıxışa bir sətirdə aralarında boşluq simvolu olmaqla 2 ədəd – eyni elementlərin verilən ardıcılıqdakı yerinin nömrəsini göstərən ədədlər verilir. Belə cütlüklər çox olduqda çıxışa onlardan istənilən bir cütlük verilir. Bütün elementlər müxtəlif olduqda isə çıxışa “0 0” verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 1 2 1 3 4	1 3
4 1 2 3 4	0 0

### T2-2008-03. Ədədlərin sayı

Bir sətirdə aralarında boşluq simvolu olmaqla 1-dən 9-dək ədədlər daxil edilir və bu proses 0 (sıfır) daxil edilənədək davam etdirilir. Daxil edilən ədədlərin sayı 1000-dən çox deyil.

Bu ardıcılıqdakı 1-lərin, 2-lərin, 3-lərin, ..., 9-ların sayını tapan proqram tərtib edin (çıxışa doqquz ədəd verilməlidir).

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3 3 3 3 3 3 3 3	0 0 7 0 0 0 0 0 0
1 1 4 1 5 8 6 3 5 1 0	4 0 1 1 2 1 0 1 0

### T2-2008-04. Axıncı rəqəm

$a$  və  $b$  müsbət tam ədədləri ( $3 \leq a, b \leq 10^9$ ) verilmişdir.  $a^b$  ( $a$  üstü  $b$ ) ədədinin axıncı rəqəmini tapan proqram tərtib edin.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 4	6
3 3	7

## T2-2008-05. Ağca və yeddi Cırt dan

Hər gün Cırt danlar mədəndə işləyən zaman Ağca onlara nahar yeməyi hazırlayır. Hər dəfə 7 ac Cırt dan nahara gələnədək masanın üstünə 7 boşqab, 7 çəngəl və 7 bıçaq qoyulur. Bir dəfə 7 Cırt dan əvəzinə 9 Cırt dan qayıtdı və onların hər biri özünün həqiqətən əvvəlki 7 Cırt dandan biri olduğunu israr etdi. Xoşbəxtlikdən hər Cırt danın başındakı qalpaqda onun nömrəsi – 100-dən kiçik tam ədəd yazılmışdı. Ağca bilirdi ki, onun öz həqiqi Cırt danlarının qalpaqlarındakı nömrələrin cəmi 100-ə bərabərdir. Elə bir proqram tərtib edin ki, 9 Cırt danın arasından öz həqiqi 7 Cırt danını seçməkdə Ağcaya kömək etsin.

### Giriş verilənləri

Aralarında boşluq simvolu olan 1-dən 99-dək ədədlər – nahara gələn Cırt danların başlarındakı qalpaqların nömrələri bir sətirdə verilir.

### Çıxış verilənləri

Çıxışa bir sətirdə cəmi 100 olan 7 ədəd – həqiqi Cırt danların nömrələri verilir (məlumdur ki, ədədlərin yalnız bir belə kombinasiyası var).

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
7 8 10 13 15 19 20 23 25	7 8 10 13 19 20 23
8 6 5 1 37 30 28 22 36	8 6 5 1 30 28 22

## T2-2009-01. Palindromlar

Verilmiş simvollar sətirinin *palindrom*, yəni soldan sağa və sağdan sola eyni cür oxunan olub-olmadığını müəyyənləşdirən proqram yazın.

### Giriş verilənlərinin formatı

Uzunluğu 255 simvoldan çox olmayan sətir.

### Çıxış verilənlərinin formatı

Verilmiş sətir palindrom olarsa, çıxışa “YES”, əks halda “NO” sözü verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
abcdefedcba	YES
k a z a k	YES
k a z a k	NO

## **T2-2009-02. Kvadratlar** (tut: T2-2004-02)

Hər bir elementi 0, 1, 5 və ya 11-ə bərabər olan  $N \times N$  ölçülü ( $2 < N < 50$ )  $A$  kvadrat cədvəli (ikiölçülü massivi) verilmişdir. Həmin kvadratın daxilində olan və hər birinin bütün elementləri fərqli olan 4-elementli ( $a_{ij}, a_{i+1,j}, a_{i,j+1}, a_{i+1,j+1}$ ) kiçik kvadratların sayını tapın. Tapılan kiçik kvadratların ortaq elementləri ola bilər.

### **Giriş verilənlərinin formatı**

Birinci sətirdə bir  $N$  ədədi, sonrakı  $N$  sətirin hər birində kvadrat cədvəlin  $N$ -ci sətirindəki  $N$  sayda ədəd verilir.

### **Çıxış verilənlərinin formatı**

Çıxışa bir ədəd – axtarılan kiçik kvadratların sayı verilir.

## **T2-2009-03. Say və cəm** (tut: T2-2004-03)

$N$  ( $2 \leq N < 100$ ) tam ədəddən ibarət ardıcılıq verilmişdir. Bu ardıcılığın mənfi elementlərindən ən böyüyü ilə müsbət elementlərinin ən kiçiyi arasında qalan elementlərin sayını və həmin elementlərin cəmini tapın (cəm və saya həmin iki elementin özləri daxil edilmir). Ardıcılıqda həm müsbət, həm də mənfi elementlər var.

### **Giriş verilənlərinin formatı**

Birinci sətirdə bir  $N$  ədədi, ikinci sətirdə isə  $N$  sayda ədəd – verilmiş ardıcılığın elementləri verilir.

### **Çıxış verilənlərinin formatı**

Çıxışa iki ədəd – axtarılan say və cəm verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
10 5 9 -3 4 7 -4 2 8 -5 10	3 7

## **T2-2009-04. Kəsrlər** (tut: T2-2000-02)

Verilmiş  $N$  natural ədədinə görə bütün sadə, ixtisar olunmayan və 1-dən böyük olmayan kəsrlər ardıcılığını artma sırası ilə almaq üçün proqram tərtib edin.

**Nümunə:**  $N = 7$  üçün çıxışa verilən ardıcılıq belə olacaq:

1/7, 1/6, 1/5, 1/4, 2/7, 1/3, 2/5, 3/7, 1/2, 4/7, 3/5, 2/3, 5/7, 3/4, 4/5, 5/6, 6/7, 1/1



## **T2-2009-05. Yerdəyişmə (tut: T2-2004-04)**

$N$  ( $1 \leq N < 100$ ) ədəddən ibarət natural ədədlər ardıcılığı verilmişdir. Bu ardıcılığın 1, 2, 3, ...,  $N$  ardıcılığından yerdəyişmə ilə alınıb-alınmadığını yoxlayın.

### **Giriş verilənlərinin formatı**

Birinci sətirdə bir  $N$  ədədi, ikinci sətirdə isə  $N$  sayda ədəd – verilmiş ardıcılığın elementləri verilir.

### **Çıxış verilənlərinin formatı**

Verilmiş ardıcılıq yerdəyişmə nəticəsində alınmışsa, çıxışa “YES”, əks halda “NO” sözü verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
7 5 3 7 2 1 4 6	YES
9 8 4 2 7 9 1 5 2 3	NO

## **T2-2010-01. Hissələrə ayırma**

$N < 30000$  natural ədədini  $M$  sayda ( $M < 20$ ) təxminən bərabər hissələrin cəmi şəklində göstərin. İxtiyari iki hissə bir-birindən ən çoxu 1 vahid fərqlənsə, onları *təxminən bərabər hissələr* adlandıracağıq.

### **Giriş verilənlərinin formatı**

Bir sətirdə  $N$  və  $M$  ədədləri verilir.

### **Çıxış verilənlərinin formatı**

Bir sətirdə  $N$  ədədinin bölündüyü  $M$  sayda hissələr azalmayan sıra ilə verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
13 4	3 3 3 4

## **T2-2010-02. Rəqəmsal kök**

İxtiyari natural ədəd götürülür və onun rəqəmlərinin cəmi tapılır. Sonra həmin cəmin rəqəmlərinin cəmi hesablanır və bu proses cəmdə birrəqəmli ədəd alınanaqədək davam etdirilir. Alınan axırıncı birrəqəmli ədədi *rəqəmsal kök* adlandırmaq.

Verilmiş  $N$  natural ədədinin ( $N < 10^{100}$ ) rəqəmsal kökünü tapan proqram yazmaq tələb olunur.

### Giriş verilənlərinin formatı

Girişə  $N$  natural ədədi verilir.

### Çıxış verilənlərinin formatı

Çıxışa  $N$  ədədinin rəqəmsal kökü verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
247	4

### T2-2010-03. Cırcıramanın marşrutu

Uzunluğu  $N$  ( $N \leq 30$ ) santimetr olan xətkəşin sol ucuna (sıfır bölgüsünə) cırcırma qonub və o, xətkəşin sağ ucuna ( $N$ -ci bölgüyə) çatmaq istəyir. Cırcırma yalnız irəliyə sıçraya bilər və hər sıçrayışın uzunluğu  $N$ -i aşmır. Müxtəlif sıçrayışların sayı  $M$ -ə bərabərdir ( $M \leq 10$ ).

Cırcıramanın ilk vəziyyətdən son vəziyyətə getmək üçün müxtəlif marşrutlarının sayını hesablamaq üçün proqram yazmaq lazımdır.

### Giriş verilənlərinin formatı

Birinci sətirdə bir-birindən boşluq simvolu ilə ayrılmış iki ədəd –  $N$  və  $M$  ədədləri, ikinci sətirdə isə heç biri  $N$ -i aşmayan, hər biri mümkün sıçrayışların uzunluğunu göstərən  $M$  sayda natural ədəd, aralarında boşluq simvolları olmaqla verilir.

### Çıxış verilənlərinin formatı

Çıxışa yeganə sətirdə müxtəlif marşrutların sayını göstərən  $S$  ədədi verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3 2 1 2	3

### T2-2010-04. Hasilin bölənləri

Verilmiş  $N$  sayda  $a_1, a_2, \dots, a_N$  ( $1 \leq N \leq 20$ ) natural ədədlərinin hər biri ( $1, 10000$ ) intervalında yerləşir.  $a_1 * a_2 * \dots * a_N$  hasilinin müxtəlif natural bölənlərinin sayını tapmaq tələb olunur.

**Qeyd.** Bütün ədədlərin hasili çox böyük ədəd ola bilər.

### Giriş verilənlərinin formatı

Birinci sətirdə  $N$  ədədi, ikinci sətirdə isə bir-birindən boşluq simvolu ilə ayrılmış  $a_1, a_2, \dots, a_N$  ədədləri verilir.

### Çıxış verilənlərinin formatı

Çıxışa axtarılan ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 3 5 7 720	120

### T2-2010-05. Sonuncuların cəmi

$a_1, a_2, a_3, \dots$  natural ədədlər ardıcılığında dördüncüdən başlayaraq hər bir sonrakı hədd əvvəlki 3 ədədin cəminin sonuncu rəqəminə bərabərdir.

Verilmiş  $a_1, a_2, a_3$  üçün  $a_N$  ədədini ( $N \leq 1000000000$ ) müəyyənləşdirən proqram yazmaq tələb olunur.

### Giriş verilənlərinin formatı

Girişə bir sətirdə bir-birindən boşluq simvolu ilə ayrılmış  $N$  və  $a_1, a_2, a_3$  natural ədədləri verilir.

### Çıxış verilənlərinin formatı

Çıxışa yeganə sətirdə axtarılan rəqəm verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1000000000 2 3 6	7

### T2-2011-01. İlbiz

İlbiz hündürlüyü  $V$  metr olan hamar ağac dirəyin dibindədir. O, dirəyin təpəsinə qalxmaq istəyir. İlbiz gündüzlər  $A$  metr yuxarı qalxır, gecələr isə yatarkən  $B$  metr aşağı sürüşür. İlbizin dirəyin təpə nöqtəsinə neçə günə çatacağını müəyyənləşdirin.

### Giriş verilənlərinin formatı

Girişə bir sətirdə 3 ədəd –  $A, B$  və  $V$  ( $1 \leq B < A \leq V \leq 1\,000\,000\,000$ ) verilir.

### Çıxış verilənlərinin formatı

Çıxışa ilbizin dirəyin təpə nöqtəsinə qalxması üçün lazım olan günlərin sayı olan bir ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2 1 5	4
5 1 6	2

## T2-2011-02. Neçə dəfə?

Natural  $N$  ədədi verilmişdir. Bu ədəddən onun rəqəmlərinin cəmi çıxılır. Alınan ədəddən yenə özünün rəqəmlərinin cəmi çıxılır və proses bu qayda ilə sonuncu müsbət ədəd alınanadək davam etdirilir. Bu əməliyyatın neçə dəfə təkrar ediləcəyini müəyyənləşdirmək lazımdır.

### Giriş verilənlərinin formatı

Girişə bir sətirdə 2 000 000 000-ı aşmayan bir  $N$  natural ədədi verilir.

### Çıxış verilənlərinin formatı

Çıxışa icra edilən əməliyyatların sayı olan ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
23	3

## T2-2011-03. Zolaqda tullanış

$1 \times N$  ölçülü üfüqi zolaq  $1 \times 1$  ölçülü damalara bölünmüşdür. Oyun daşı  $S$  nömrəli damadadır. Birinci gedişdə daş bir dama yerini dəyişməlidir (sağa və ya sola). İkinci gedişdə iki dama və hər sonrakı gedişdə əvvəlkindən bir dama artıq adlamalıdır. Bu proses daşı zolaqda saxlayan son gedişədək davam etdirilir. Daşın edə biləcəyi gedişlərin maksimum və minimum sayını müəyyənləşdirin.

### Giriş verilənlərinin formatı

Bir sətirdə iki tam  $N$  və  $S$  ədədləri verilir ( $1 \leq N \leq 10^9$ ,  $1 \leq S \leq N$ ).

### Çıxış verilənlərinin formatı

Çıxışa bir sətirdə iki ədəd – daşın gedişlərinin maksimum və minimum sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3 2	2 2
6 2	5 3

## T2-2011-04. Kürəciklər

Satıcının  $N$  sayda müxtəlifrəngli kürəciyi vardır. Ancaq son vaxtlar belə bir göstəriş verilmişdir ki, yalnız hansısa bir rəngdə olan kürəciklər satıla bilər. Bu göstərişi pozmaq üçün satıcı satdığı kürəciklərin bir qismini yenidən rəngləmək qərarına gəlib.

Minimal sayda rəngləməni müəyyənləşdirən proqramı tərtib edin.

### Giriş verilənlərinin formatı

Birinci sətirdə kürəciklərin  $N$  ( $1 \leq N \leq 100000$ ) sayı verilir. İkinci sətirdə kürəciklərin rəngini müəyyənləşdirən və 1-dən 9-dək qiymətlər alan  $N$  ədəd yerləşir (1 – göy, 2 – yaşıl, 3 – mavi, 4 – qırmızı, 5 – narıncı, 6 – sarı, 7 – boz, 8 – qara, 9 – ağ).

### Çıxış verilənlərinin formatı

Çıxışa bir sətirdə bütün kürəciklərin eyni bir rəngdə olması üçün rənglənən şarların minimal sayını göstərən bir ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 3 1 2 1	2

### T2-2011-05. Robot

Eni 1 olan və hər iki tərəfə sonsuz uzanan zolaq  $1 \times 1$  ölçülü damalara bölünmüşdür. Bu damaların birində dayanan robot (şəkildə həmin robot kvadratla işarə edilmişdir) bir damadan başqasına hərəkət edə bilər.



Robotun yerdəyişməsini müəyyənləşdirən proqramın hər bir komandası latın əlifbasının 3 baş  $L$ ,  $R$ ,  $S$  hərflərinə uyğundur.  $L$  komandası icra edildikdə robot bir dama sola,  $R$  komandasında bir dama sağa hərəkət edir,  $S$  komandasında isə olduğu damada qalır. Proqramın icrası dedikdə orada yazılan bütün komandalарın ardıcıl yerinə yetirilməsi başa düşülür.

Robotun neçə müxtəlif damaya gedəcəyini müəyyənləşdirən proqram yazın.

### Giriş verilənlərinin formatı

Birinci sətirdə robot üçün yazılmış proqramda komandalарın sayı olan  $N$  ( $N \leq 1000$ ) ədədi, ikinci sətirdə isə robot üçün  $L$ ,  $R$ ,  $S$  simvollarından ibarət proqram verilir.

### Çıxış verilənlərinin formatı

Çıxışa öz proqramını icra edərkən robotun gəzdiyi müxtəlif damaların sayını göstərən bir ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
8 RRSRRLRR	6

## T2-2012-01. Ayı balası

Ayı balası yol ilə gedir və bu yolun qıraqları boyunca aralarındakı məsafə **M** olan ağaclar əkilib. Hər bir ağacın altında dayanan ayı balası haradan gəldiyini unudur və müəyyən müddət irəli getdikdən sonra hərəkət istiqamətini bu və ya digər tərəfə təsadüfi olaraq seçir. **K** mərhələdən sonra ayı balası “başlanğıc” ağacdən hansı məsafələrdə ola bilər?

### Giriş verilənlərinin formatı

Bir sətirdə aralarında boşluq simvolu olmaqla iki müsbət tam **M** və **K** ( $1 \leq M, K \leq 10000$ ) ədədləri verilir.

### Çıxış verilənlərinin formatı

Bir sətirdə aralarında boşluq simvolu olmaqla ayı balasının ola biləcəyi məsafələr (kiçikdən böyüyə artma sırası ilə) verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2 6	0 4 8 12

## T2-2012-02. Xırda mismarlar

Textanın üzərinə xırda mismarlar vurulmuşdur. İxtiyari iki mismarı sapla birləşdirmək olar. Hansısa mismarlar cütliklərini sapla elə birləşdirmək tələb olunur ki, hər bir mismara, heç olmasa, bir sap bağlansın və bütün sapların uzunluqları cəmi minimum olsun.

### Giriş verilənlərinin formatı

Birinci sətirdə mismarların sayını bildirən **N** ədədi ( $2 \leq N \leq 100$ ) verilir. Sonrakı sətirdə isə bütün mismarların koordinatları olan **N** sayda ədəd (10000-i aşmayan mənfi olmayan tam ədədlər) verilir.

### Çıxış verilənlərinin formatı

Çıxışa bir sətirdə bütün sapların uzunluqlarının minimum cəmini göstərən bir ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 4 10 0 12 2	6

## T2-2012-03. Proqramçılar komandası

Proqramlaşdırma üzrə komanda birinciliyində iştirak etmək üçün cəmi **n** sayda oğlan və qız şagird yazılıbdır. Professor Paskal onları bir sıraya düzür və ardıcıl

dayanmış şagirdlərdən elə komanda seçmək istəyir ki, orada oğlan və qızların sayı eyni olsun. Professorun seçim üçün neçə mümkün variantı var?

### Giriş verilənlərinin formatı

Birinci sətirdə  $n$  ədədi ( $1 \leq n \leq 10^6$ ) verilir. İkinci sətirdə oğlan və qızlardan təşkil edilmiş sıra, yəni  $n$  sayda  $a$  və  $b$  simvollarından ( $a$  simvolu qızlara,  $b$  isə oğlanlara uyğundur) ibarət sətir verilir.

### Çıxış verilənlərinin formatı

Çıxışa bir sətirdə bir ədəd – tələb olunan komandaların seçim variantlarının miqdarı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3 bab	2
8 abbababa	13

### T2-2012-04. Xoşbəxt ədədlər

4 və 7 rəqəmləri *xoşbəxt rəqəmlər*, qalan rəqəmlər isə xoşbəxt olmayan rəqəmlər sayılır. Yalnız xoşbəxt rəqəmlərdən ibarət tam ədəd *xoşbəxt ədəd* hesab olunur. **K**-cı xoşbəxt müsbət tam ədədin hansı ədəd olduğunu bilmək tələb olunur.

### Giriş verilənlərinin formatı

Bir sətirdə bir **K** müsbət tam ədədi ( $1 \leq K \leq 10^9$ ) verilir.

### Çıxış verilənlərinin formatı

Bir sətirdə bir ədəd – **K**-cı xoşbəxt müsbət tam ədədin özü verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1	4
2	7
3	44

### T2-2012-05. Hemming ədədləri

Hemming ardıcılığı sadə bölənləri yalnız 2, 3, 5 olan artan ədədlər ardıcılığıdır. Məsələn, bu ardıcılığın ilk 10 həddi bunlardır: 2, 3, 4, 5, 6, 8, 9, 10, 12, 15. Verilmiş **N** üçün bu ardıcılığın **N**-ci həddini tapın.

### Giriş verilənlərinin formatı

Birinci sətirdə bir müsbət tam  $N$  ədədi ( $1 \leq N \leq 100$ ) verilir.

### Çıxış verilənlərinin formatı

Bir sətirdə Hemming ardıcılığının  $N$ -ci həddi olan ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1	2
4	5
8	10

### T2-2013-01. Hərflər

STUDENT sözündə olan hərfləri verilmiş sözdə tapıb silən proqram yazın.

### Giriş verilənlərinin formatı

İngilis əlifbasının 3-dən az və 20-dən çox olmamaqla baş hərflərindən ibarət söz daxil edilir.

### Çıxış verilənlərinin formatı

Verilmiş sözdən STUDENT sözünün hərflərini sildikdən sonra alınan söz çıxışa verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
DAY	AY
DUTY	Y

### T2-2013-02. Yeni qaydalar

Yeni qaydalara görə, olimpiadanın respublika turunda 8 məsələ təklif ediləcək. Hər bir olimpiada iştirakçısı istənilən məsələləri həll edə bilər, ancaq hər bir iştirakçının yekun balı toplam ən yüksək bal almış 5 məsələ üzrə qiymətləndirilir. Hər məsələ üzrə toplanmış bala görə iştirakçının yekun balını və balları yekun qiymətləndirməyə daxil olan məsələlərin nömrələrini müəyyənləşdirən proqram yazın.

### Giriş verilənlərinin formatı

Bir sətirdə aralarında boşluq simvolu olmaqla iştirakçının hər bir məsələ üzrə aldığı ballar olan 8 tam ədəd yerləşir.  $i$ -ci ədəd  $i$ -ci məsələ üzrə toplanmış baldır.



### Çıxış verilənlərinin formatı

Çıxış verilənləri iki sətirdən ibarətdir. Birinci sətirdə iştirakçının cəmi topladığı maksimum bal (5 ən yaxşı həll üzrə) yerləşir. İkinci sətirdə balları yekun qiymətləndirmədə nəzərə alınan məsələlərin nömrələri aralarında boşluq simvolu olmaqla artma ardıcılığı ilə verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
20 30 50 48 33 66 0 64	261 3 4 5 6 8
20 0 50 80 77 110 56 48	373 3 4 5 6 7

### T2-2013-03. Dördüncü təpə

Məmməd müstəvi üzərində elə 4 nöqtə seçməlidir ki, onlar tərəfləri koordinat oxlarına paralel olan düzbucaqlının təpə nöqtələri olsun. O artıq üç nöqtə seçib. Dördüncü nöqtəni müəyyənləşdirməkdə ona kömək edin.

### Giriş verilənlərinin formatı

Üç sətirdə aralarında bir boşluq simvolu olmaqla düzbucaqlının seçilmiş təpə nöqtələrinin (X, Y) koordinatları yerləşir. Hər bir təpə nöqtəsi ayrıca sətirdədir. Bütün ədədlər tamdır və 1-lə 1000 arasında yerləşir.

### Çıxış verilənlərinin formatı

Bir sətirdə aralarında bir boşluq simvolu olmaqla düzbucaqlının dördüncü təpə nöqtəsinin koordinatları olan iki tam ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 5 5 7 7 5	7 7
30 20 10 10 10 20	30 10

### T2-2013-04. Bayram tortu

Məmmədin atası Yeni il bayramı üçün kvadrat tort sifariş verdi və bu tortu onun tərəflərinə paralel düz xətlərlə maksimum sayda neçə hissəyə kəsməyin mümkünlüyünü hesablamağı Məmmədə tapşırırdı. Hər bir xətt tortu tam kəsib keçir.

### Giriş verilənlərinin formatı

Bir sətirdə bir natural  $N$  ( $1 \leq N \leq 100$ ) ədədi verilir.

### Çıxış verilənlərinin formatı

Tortu kəsməyin mümkün olduğu hissələrin maksimum sayı olan bir ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1	2
3	6

### T2-2013-05. Ədədi silsilə

Məmməd ədədi silsilə əmələ gətirən 4 ədəd seçdi, yəni həmin ədədləri artma sırası ilə düzdükdə istənilən iki qonşu ədəd arasındakı fərq eyni olur. Həmişə olduğu kimi, Məmməd ədədlərdən birini itirib və yadına da sala bilmir. Bu dördüncü ədədi tapmaqda Məmmədə kömək edin.

### Giriş verilənlərinin formatı

Bir sətirdə boşluq simvolu ilə ayrılmış üç natural ədəd verilir. Hər bir ədəd 100-ü aşmır.

### Çıxış verilənlərinin formatı

Verilmiş üç ədədlə ədədi silsilə əmələ gətirən bir ədəd verilir. Əgər belə ədədlər birdən çoxdursa, onda onlardan ən böyüyü seçilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 6 8	10
10 1 4	7

### T2-2014-01. Nöqtə və kvadrat

Tərəfi  $N$  dama olan kvadratla damalı vərəqdəki tam koordinatlı hansı ən böyük sayda nöqtəni örtmək olar?

### Giriş verilənləri

Kvadratın tərəfi olan  $N$  ( $1 \leq N \leq 10000$ ) ədədi verilir.

### Çıxış verilənləri

Örtülmüş nöqtələrin  $K$  maksimum sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1	4

## T2-2014-02. Üç dovşan

Üç dovşan düz xətt boyunca yerləşən kvadratların üzərində oynayır. Hər dovşan yalnız bir kvadratda yerləşə bilər. Oyunun bir gedişində istənilən dovşan digər iki dovşanın arasındakı boş kvadrata tullana bilər. Dovşanların verilmiş başlanğıc vəziyyətinə görə onların nə qədər oynaya biləcəklərini müəyyən-ləşdirin.

### Giriş verilənləri

Bir sətirdə başlanğıc anda dovşanların durduqları kvadratların koordinatları olan üç tam **A**, **B** və **C** ( $0 < A < B < C < 100$ ) ədədləri verilir.

### Çıxış verilənləri

Oyunda gedişlərin maksimal sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2 3 5	1
3 5 9	3

**Açıqlama:** 1-ci nümunədə **A** dovşanı yalnız 4-cü kvadrata tullana bilər və bundan sonra əlavə heç bir gediş qalmır.

	A	B		C					
1	2	3	4	5	6	7	8	9	10

## T2-2014-03. Cəm

Tam ədədlərdən ibarət birölçülü massiv verilmişdir. Massivin müsbət element-lərinin ən kiçiyi ilə mənfi elementlərinin ən böyüyünün cəmini tapın.

### Giriş verilənləri

Birinci sətirdə massivin elementlərinin sayını bildirən **N** ( $N \leq 100$ ) ədədi verilir. İkinci sətirdə aralarında boşluq simvolu olmaqla hər birinin qiyməti mütləq qiymətcə 100-ü aşmayan massiv elementləri verilir.

### Çıxış verilənləri

Massivin müsbət elementlərinin ən kiçiyi ilə mənfi elementlərinin ən böyü-yünün cəmi çıxışa verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
10 2 4 -3 -7 5 -2 8 6 9 -1	1

## T2-2014-04. Yeni oyun

Yeni il bayramında oyun istehsalçısı olan firma Alpaya yeni başsındıran oyun qurğusu hədiyyə etdi. Oyun qurğusu 25 düymə və böyük ekrandan ibarətdir. Hər bir düymə bir oyunu işə salır. Alpay bütün başsındıran oyunları həll edən kimi firma həmin oyun qurğusunu yenisi ilə əvəz etməyi öhdəsinə götürür. Birinci düymə aşağıdakı oyunu işə salır: oyun başladılan kimi ekrana böyük A hərfi çıxır. Oyun başlayandan sonra bu düyməni ilk dəfə sıxdıqda A hərfi B hərfi ilə əvəz olunur. Bu düymənin sonrakı sıxılmalarında B hərfi BA ilə, sonra BA cütünü BAB ilə, daha sonra BAB üçlüyü BABBA ilə əvəzlənir. Alpay anlayır ki, hər dəfə birinci düyməni sıxanda sözdəki bütün B hərfləri BA ilə, A hərfi isə B ilə əvəz olunur. Birinci düyməni K dəfə sıxdıqdan sonra ekranda neçə A hərfi və neçə B hərfi olacaq? Alpay bu başsındıran məsələni həll etdi. Siz necə?

### Giriş verilənləri

Bir sətirdə düymənin sıxılma sayı olan bir  $K$  ( $1 \leq K \leq 45$ ) ədədi verilir.

### Çıxış verilənləri

Bir sətirdə aralarında boşluq simvolu olmaqla A və B hərflərinin sayını göstərən iki ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1	0 1
4	2 3
10	34 55

## T2-2014-05. Mobil telefon nömrəsi

Mobil telefon nömrəsi verilmişdir. Bu nömrədə hansı rəqəmlərin olmadığını müəyyənləydirin.

### Giriş verilənləri

Bir sətirdə mobil telefon nömrəsi verilir.

### Çıxış verilənləri

Birinci sətirdə nömrədə olmayan rəqəmlərin sayı, ikinci sətirdə isə aralarında boşluq simvolu olmaqla nömrədə olmayan rəqəmlər artan sıra ilə çıxışa verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
0504974560	4 1 2 3 8

## 5.2. HƏLL NÜMUNƏLƏRİ

T2\_2003\_01.

Hissələrə ayırma

**Program T2\_2003\_01;**

```
var
    n, m, n1, n2, x, i: integer;      {dəyişənlər təsvir edilir}
begin
    readln(n);                        {n daxil edilir}
    readln(m);                        {m daxil edilir}
    n2 := n mod m; {n-in m-ə bölünməsindən alınan qalıq tapılır}
    n1 := m - n2;
    x := n div m;  {n-in m-ə bölünməsindən alınan qismət tapılır}

    for i := 1 to n1 do
        write(' ', x);
    for i := 1 to n2 do
        write(' ', x + 1);
end.
```

T2\_2003\_02.

Sadə çəki daşları

**Program T2\_2003\_02;**

```
var
    n, m, p, l, j, k: longint;
begin
    readln(n);
    m := n;
    while m > 1 do begin
        if odd(m) then p := m else p := m + 1;
        repeat
            j := 3;
            k := round(sqrt(p));
            while (j <= k) and (p mod j <> 0) do
                j := j + 2;
            if j <= k then p := p + 2;
        until j > k;
        l := m;
        if l = p then l := l - 1;
        repeat
            writeln(p - l, ' ', l);
```

```

        l := l - 1;
    until l < p - 1;

    m := p - m - 1;
end;
end.

```

**T2\_2003\_04.**

Maksimal ardıcıl cəm

**Program T2\_2003\_04;**

```

var
    A: array[1..50] of integer;
    St, Smax, i, N: integer;
begin
    readln(N);
    for i := 1 to N do {Ardıcılığın elementləri daxil edilir}
        read(A[i]);

    St := A[1];
    Smax := St;

    for i := 2 to N do begin {Bütün ardıcılığa baxılması}

        if St < 0 then St := 0; {Əgər cəm sıfırdan kiçik olarsa,}
                                {onu sıfıra bərabər etməli}
        St := St + A[i];        {Yeni cəmlərin hesablanması}
        if St > Smax then
            Smax := St;
    end;
    writeln;
    writeln(Smax);
end.

```

**T2\_2010\_01.**

Hissələrə ayırma

**Program T2\_2010\_01;**

```

var
    i, m, n, k: integer;
begin
    readln(n, m);
    k := n mod m;
    for i := 1 to m - k do
        write(n div m, ' ');
    for i := m - k + 1 to m do
        write((n div m)+1, ' ');
    end.

```

**Program T2\_2010\_02;**

```

var
    k : longint;
    ch: char;

function f(k: longint): longint;
var
    t: longint;
begin
    t := 0;
    while k > 0 do begin
        t := t + (k mod 10);
        k := k div 10;
    end;
    f := t;
end;

begin
    while not eoln do begin
        read(ch);
        k := k + ord(ch) - 48;
    end;
    while k > 9 do
        k := f(k);
    writeln(k);
end.

```

**Program T2\_2010\_03;**

```

var
    s, a: array[1..1000] of integer;
    g: array[1..1000] of boolean;
    i, j, n, m: integer;

begin
    readln(n, m);
    for i:=1 to m do begin
        read(j);
        a[i] := j;
        g[j] := true;
    end;
    for i:=1 to n do begin

```

```

        for j:=1 to m do
            if i-a[j] > 0 then s[i] := s[i-a[j]] + s[i];
            if g[i] = true then inc(s[i]);
        end;
        writeln(s[n]);
    end.

```

**T2\_2010\_04.**

Hasilin bölənləri

**Program T2\_2010\_04;**

```

var
    n, i, k : longint;
    a       : array [1..10000] of longint;

procedure p(x: longint);
var
    k : longint;
begin
    k := 2;
    while x > 1 do begin
        while (x mod k) <> 0 do
            inc(k);
        x := x div k;
        inc(a[k]);
    end;
end;

begin
    readln(n);
    for i := 1 to n do begin
        read(k);
        p(k);
    end;
    k := 1;
    for i := 1 to 10000 do
        k := k * (a[i] + 1);
    writeln(k);
end.

```

**T2\_2010\_05.**

Sonuncuların cəmi

**Program T2\_2010\_05;**

```

var
    n, i, x, y, z, a1, a2, a3, a4, count : longint;

```



```

begin
  readln(n, x, y, z);
  a1 := y; a2 := z;
  a3 := x + y + z;
  count := 1;
  while (a1 <> x) or (a2 <> y) or (a3 <> z) do begin
    a4 := a1 + a2 + a3;
    a1 := a2;
    a2 := a3;
    a3 := a4 mod 10;
    inc(count);
  end;
  n := ((n - 1) mod count) + 1;
  if n = 1 then writeln(a1)
    else if n = 2 then writeln(a2)
      else if n = 3 then writeln(a3)
        else
          begin
            a1 := x;
            a2 := y;
            a3 := z;
            for i := 1 to n - 3 do begin
              a4 := a1 + a2 + a3;
              a1 := a2;
              a2 := a3;
              a3 := a4 mod 10;
            end;
            writeln(a3);
          end;
  end;
end.

```

T2\_2011\_01.

İlbiz

### 1-ci variant

**Program T2\_2011\_01;**

```

var
  a, b, v, t, x: longint;

begin
  t := 1; x := 0;
  readln(a, b, v);
  while (x + a < v) do begin

```

```

        x := x + a - b;
        inc(t);
    end;
    writeln(t);
end.

```

## 2-ci variant

**Program T2\_2011\_01;**

```

var
    a, b, v: longint;

begin
    readln(a, b, v);
    writeln((v - b - 1) div (a - b) + 1);
end.

```

**T2\_2011\_02.**

Neçə dəfə?

**Program T2\_2011\_02;**

```

var
    n, m, k: longint;
    s       : integer;

begin
    readln(n);
    k := 0;
    while (n > 0) do begin
        inc(k);
        m := n;
        s := 0;
        while (m <> 0) do begin
            s := s + (m mod 10);
            m := m div 10;
        end;
        n := n - s;
    end;
    writeln(k);
end.

```

**Program T2\_2011\_03;**

```

const
    m = 1000;
var
    a, b: array[0..1, 1..m] of integer;
    n, i, j, p1, p2, q1, q2, k1, k2, s, l: integer;

begin
    readln(n, s);
    k1 := 0;
    k2 := 1;
    if (n = 1) then begin writeln(0, ' ', 0); end
    else begin
        for i := 0 to 1 do
            for j := 1 to n do begin
                a[i, j] := 0;
                b[i, j] := 0;
            end;
        l := n div 2;
        if (n mod 2 = 1) then inc(l);
        a[1, 1] := 1;
        a[1, n] := 1;
        b[1, 1] := 1;
        b[1, n] := 1;

        for i := n - 2 downto 1 do begin
            j := 1;
            while (j <= l) and ((i+j) <= n) do begin
                if (i + j <= n) then begin
                    p1 := a[k2, i+j] + 1;
                    q1 := b[k2, i+j] + 1;
                end
                else begin
                    p1 := n + 1;
                    q1 := 0;
                end;
            end;
            if (j - i >= 1) then begin
                p2 := a[k2][j-i] + 1;
                q2 := b[k2][j-i] + 1;
            end
            else begin
                p2 := n + 1;
                q2 := 0;
            end;
        end;
    end;
end;

```

```

        end;
        if (p1 > p2) then
            a[k1, j] := p2
        else
            a[k1, j] := p1;
        a[k1, n - j + 1] := a[k1, j];

        if (q1 > q2) then
            b[k1, j] := q1
        else
            b[k1, j] := q2;
        b[k1, n - j + 1] := b[k1, j];

        inc(j);
    end;
    k1 := 1 - k1;
    k2 := 1 - k2;
end;
writeln(b[k2, s], ' ', a[k2, s]);
end;
end.

```

**T2\_2011\_04.**

**Kürəciklər**

**Program T2\_2011\_04;**

```

const
    a: array[0..9] of integer = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0);
var
    n, b, max, i: integer;
begin
    readln(n);
    max := 0;
    for i:=1 to n do begin
        read(b);
        inc(a[b]);
        if (a[b] > max) then
            max := a[b];
        end;
    writeln(n - max);
end.

```

**T2\_2011\_05.**

Robot

**Program T2\_2011\_05;**

```
var
    a: char;
    n, min, max, t, i: integer;

begin
    readln(n);
    for i := 1 to n do begin
        read(a);
        if (a = 'R') then inc(t)
        else
            if (a = 'L') then dec(t);
        if (t > max) then max := t
        else
            if (t < min) then min := t;
    end;
    writeln(max - min + 1);
end.
```

**T2\_2012\_01.**

Ayı balası

**Program T2\_2012\_01;**

```
var
    i, j, k: longint;

begin
    read(j, k);
    for i := 0 to (k div 2) do
        write(j * (2 * i + (k mod 2)), ' ');
    writeln;
end.
```

**T2\_2012\_02.**

Xırda mismarlar

**Program T2\_2012\_02;**

```
var
    a, b, c, d: array[0..100] of integer;
    n, i, x, y: integer;

procedure sort;
var
    i, j, c: integer;
```

```

begin
  for i := 0 to n - 2 do
    for j := n - 2 downto i do
      if a[j] > a[j+1] then begin
        c := a[j];
        a[j] := a[j+1];
        a[j+1] := c;
      end;
    end;
  end;
begin
  readln(n);

  for i := 0 to n - 1 do
    read(a[i]);

  sort;
  b[0] := 0;
  c[0] := 0;
  c[1] := 0;
  for i := 0 to n - 2 do
    b[i+1] := a[i+1] - a[i];
  for i := 0 to n - 3 do
    c[i+2] := b[i+2] + b[i+1];

  d[1] := b[1];
  d[2] := c[2];
  d[3] := b[3] + b[1];

  for i := 4 to n - 1 do begin
    x := b[i] + d[i-2];
    y := c[i] + d[i-3];
    if (x < y) then
      d[i] := x
    else
      d[i] := y;
  end;
  writeln(d[n-1]);
end.

```

**Program T2\_2012\_03;**

```

var
  c: array[0..250] of shortint;
  n, s, i, j, t: integer;
  ss: string;

begin
  readln(n);
  readln(ss);
  s := 0;
  for i := 0 to n - 1 do
    if (ss[i+1] = 'a') then
      c[i] := 1
    else
      c[i] := -1;
  for i := 1 to n - 1 do begin
    t := c[i] ;
    for j := i - 1 downto 0 do begin
      t := t + c[j];
      if t = 0 then inc(s);
    end;
  end;
  writeln(s);
end.

```

**Program T2\_2012\_04;**

```

var
  a, b : string;
  n     : integer;

begin
  readln(n);
  a := '';
  b := '47';
  while n > 0 do begin
    n := n - 1;
    a := b[n mod 2 + 1] + a;
    n := n div 2;
  end;
  writeln(a)
end.

```

**Program T2\_2012\_05;**

```
var
    n, i, m, k, j: integer;
begin
    readln(n);
    i := 3;
    if n > 3 then begin
        j := 2;
        while (j <= n) do begin
            m := i;
            k := 2;
            while (k < 6) do begin
                while (m mod k = 0) do
                    m := m div k;
                inc(k);
            end;
            if m = 1 then inc(j);
            inc(i);
        end;
    end;
    writeln(i-1);
end.
```





## VI BÖLÜM

# OLİMPİADANIN YEKUN (RESPUBLİKA) TURUNUN MƏSƏLƏLƏRİ

## 6.1. MƏSƏLƏLƏRİN ŞƏRTLƏRİ

Bu bölümdə 2000–2014-cü illərdə olimpiadanın üçüncü – yekun mərhələsində təqdim olunmuş məsələlər verilib. Hər bir məsələ aşağıdakı kimi kodlaşdırılıb: başlanğıcdakı “T3” yazısı məsələnin üçüncü tura (respublika turuna) aid olduğunu bildirir, sonra olimpiadanın keçirildiyi tədris ili, sonda isə məsələnin sıra nömrəsi verilir. Koddan sonra isə məsələnin adı yazılacaq. Məsələn, “T3-2003-01. Uzun ədədlər” kodlaşdırması məsələnin 2002/2003 tədris ilində keçirilmiş respublika turunun 1-ci məsələsi olduğunu göstərir.

### T3-2000-01. Yerdəyişmə

Verilmiş  $N$  tam ədədi ( $N \leq 10^{100}$ ) üzərində aşağıdakı çevirmə aparılır: ədəddəki rəqəmlərin yerini dəyişməklə maksimal və minimal ədədlər müəyyən olunur və onların fərqi tapılır. Məsələn, verilmiş ədəd 471-dirsə, onda maksimal ədəd 741, minimal ədəd isə 147 olar. Bunların fərqi 594 olacaq, yəni 471 ədədinə tətbiq edilən çevirmənin nəticəsi 594 olur. Yenidən çevirmə 594 ədədinə tətbiq edilsə, 495 ədədi alınar. Bundan sonra dövrü 495 olan ardıcılıq başlanır. Beləliklə, 471 ədədi üçün 471, 594, 495, 495, 495, ... ardıcılığı alınır.

Daxil edilmiş ədəd üçün bu cür ardıcılığın uzunluğunu təyin edən proqram yazmaq tələb olunur.

**Giriş verilənləri** (IN1.TXT faylından və ya klaviaturadan)

$N$  tam ədədi ( $N \leq 10^{100}$ ) verilir.

**Çıxış verilənləri** (OUT1.TXT faylına və ya ekrana)

Çıxışa verilmiş çevirmənin  $N$  ədədinə ardıcıl tətbiq etməklə alınan ardıcılığın uzunluğu verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
471	1

### T3-2000-02. Mötərizələrin açılması

Bir neçə dəyişənli iki cəmin hasilindən ibarət cəbri ifadə verilib. Bu ifadəni sadələşdirən – mötərizələri açan və oxşar hədləri islah edən proqram yazın.

**Giriş verilənləri** (IN2.TXT faylından və ya klaviaturadan)

Giriş cəbri ifadəsi  $(x_1 + x_2 + \dots + x_k) \times (y_1 + y_2 + \dots + y_k)$  şəklindədir. Bütün dəyişənlər latın əlifbasının kiçik hərfləri ilə işarə olunur. Cəmlərdə dəyişənlər təkrar olunmur.

**Çıxış verilənləri** (OUT2.TXT faylına və ya ekrana)

Ekrana  $xy$ ,  $2xy$ ,  $x^2$  şəklində olan birhədlilərin cəmini vermək lazımdır.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
$(a + b) * (a + b)$	$a^2 + 2ab + b^2$

### T3-2000-03. Dəniz xəritəsi


$N \times M$  ölçülü ( $N, M \leq 100$ ) damalı sahədə dəniz xəritəsi çəkilmişdir. Dənizin damaları “0” simvolu, quru sahənin damaları isə “#” simvolu ilə işarələnmişdir. Adalar tərəfləri bir-birinə toxunan quru damalarından ibarətdir. Xəritəni daxil edən, adaların sayını tapan və ən uzun sərhədi olan adanın damalarından birinin koordinatlarını, eləcə də bu sərhədin uzunluğunu çıxışa verən proqram tərtib edin.

Giriş verilənlərinin formatı (IN3.TXT faylından və ya klaviaturadan)	Çıxış verilənlərinin formatı (OUT3.TXT faylına və ya ekrana)
<b>N</b> boşluq simvolu <b>M</b> Xəritənin 1-ci sətri ( <b>M</b> uzunluqlu) Xəritənin 2-ci sətri ... ... Xəritənin <b>N</b> -ci sətri	Adaların sayı Damanın koordinatları (boşluqla ayrılmış) Sərhədin uzunluğu

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
6 7 0000000 0####00 0#000#0 0000##0 00000#0 0000000	2 2 2 12

### T3-2001-01. Dəniz döyüşü

“Dəniz döyüşü” kompüter oyununda döyüş zonasının xəritəsi  $N \times N$  ölçülü ( $N \leq 100$ ) kvadrat sahədir. Su olan hissələr “O”, ada və hərbi gəmilər isə “X” ilə işarə olunub. Oyunda 4 növ gəmi iştirak edir:

1-ci növ – biryaruslu gəmi	
2-ci növ – ikiyaruslu gəmi	
3-cü növ – üçyaruslu gəmi	
4-cü növ – təyyarədaşıyan gəmi	

Quru ərazilər formasına görə gəmilərdən fərqlənir. Gəmilər öz aralarında yan tərəfdən toxunmur. Gəmilər həm üfüqi, həm də şaquli yerləşə bilər.

Verilmiş döyüş əməliyyatı xəritəsinə görə bütün növ gəmilərin və adaların miqdarını təyin edin.

<b>Giriş verilənlərinin formatı (SEA.IN faylından və ya klaviaturadan)</b>	<b>Çıxış verilənlərinin formatı (SEA.OUT faylına və ya ekrana)</b>
N (sahənin ölçüsü) 1-ci sətir. N sayda “O” və ya “X” simvolu 2-ci sətir. N sayda “O” və ya “X” simvolu 3-cü sətir. N sayda “O” və ya “X” simvolu ... ... N-ci sətir. N sayda “O” və ya “X” simvolu	1-ci növ gəmi – gəmilərin miqdarı 2-ci növ gəmi – gəmilərin miqdarı 3-cü növ gəmi – gəmilərin miqdarı 4-cü növ gəmi – gəmilərin miqdarı Adalar – adaların miqdarı

<b>Giriş verilənlərinə nümunə</b>	<b>Çıxış verilənlərinə nümunə</b>
10 OOOOOOOOOO OXOOOOOOOO OOOOOOXOOO OOXOOOXOOO XXXXOOXXOO OOXOOOOOOO OOOOOOXXOO XOOOOOOOOX XOOOXXOOOX	Biryaruslu - 2 İkiyaruslu - 2 Üçyaruslu - 1 Təyyarədaşıyan gəmi - 1 Adalar - 2

### T3-2001-02. Dəmir yolu

N sayda ( $N < 50$ ) şəhər və onların ( $X$ ,  $Y$ ) koordinatları verilmişdir. Bu şəhərlərə müxtəlif dəmiryol şirkətləri xidmət göstərir. Bəzən bir şəhərdən başqasına dəmir yolu ilə getmək mümkün olmur və bu zaman avtobusla getməli olursan. Başqa sözlə, dəmir yolu şəbəkələri öz aralarında əlaqə yarada bilmir.

Verilmiş şəhərlər və onlar arasında verilmiş şəbəkəyə görə ixtiyari bir şəhərdən başqasına dəmir yolu ilə getmək mümkündürmü, əgər mümkün deyilsə, elə şəhərlər tapın ki, onlar arasında ən qısa uzunluqda dəmir yolu xətti çəkməklə bütün şəhərlər dəmir yolu xətti ilə əlaqəli olsun.

Giriş verilənlərinin formatı (RWAY.IN faylından və ya klaviaturadan)	Çıxış verilənlərinin formatı (RWAY.OUT faylına və ya ekrana)
N (şəhərlərin sayı) $X_1 Y_1$ (1-ci şəhərin koordinatları) $X_2 Y_2$ (2-ci şəhərin koordinatları) ... $X_N Y_N$ (N-ci şəhərin koordinatları) M (şəhərlər arasında birləşmələrin sayı) $U_1 V_1$ (1-ci birləşmə – iki şəhərin nömrəsi) $U_2 V_2$ (2-ci birləşmə) ... $U_M V_M$ (M-ci birləşmə)	Əgər bütün şəhərlər eyni dəmiryol şəbəkəsindədirsə, onda çıxışa aşağıdakı mətn verilir: Bütün şəhərlər birləşmişdir. Əks halda $K L D$ çap edilir. Burada K və L dəmir yolu vasitəsilə birləşdiriləsi şəhərlərin nömrəsidir, D isə onlar arasındakı məsafədir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
6 2.0 5.5 4.5 6.0 5.5 3.5 3.5 2.0 3.0 3.0 2.0 3.5	6 5 1.118

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 1 2 1 6 5 3 3 4	

### T3-2001-03. Faktorial

$N!$  –  $N$ -faktorial ədədinin ( $N < 5000$ ) tam yazılışında təkrarlanan 0–9 rəqəmlərinin sayını tapmaq üçün proqram tərtib edin.

Giriş verilənlərinin formatı (FAKT.IN faylından və ya klaviaturadan)	Çıxış verilənlərinin formatı (FAKT.OUT faylına və ya ekrana)
N ədədi	M ( $N!$ ədədində olan rəqəmlərin sayı) 0 – təkrarlanan 0-ların sayı 1 – təkrarlanan 1-lərin sayı 2 – təkrarlanan 2-lərin sayı ... 9 – təkrarlanan 9-ların sayı

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
999	2665 0 469 1 239 2 248 3 216 4 229 5 213 6 231 7 217 8 257 9 246

### T3-2001-04. Binomlar

$AX+B$  şəklində olan  $N$  sayda ( $N < 100$ ) binomun hasilini hesablayan proqram yazın. Burada  $A$ ,  $B$  tam ədədlərdir və onlar eyni zamanda sifıra bərabər ola bilməz.

Giriş verilənlərinin formatı (BIN.IN faylından və ya klaviaturadan)	Çıxış verilənlərinin formatı (BIN.OUT faylına və ya ekrana)
N (vuruqların – ikihədlilərin sayı) $A_1, B_1$ (1-ci ikihədlinin əmsalları) $A_2, B_2$ (2-ci ikihədlinin əmsalları) ... $A_N, B_N$ ( $N$ -ci ikihədlinin əmsalları)	N (alınan çoxhədlinin qüvvəti) $X^0$ -in əmsalı $X^1$ -in əmsalı ... $X^N$ -in əmsalı

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3 3 -5 2 7 4 -3	3 105 -173 26 24

### T3-2002-01. Bərabər cəmlər

(1000, 9999) intervalında yerləşən, 1-ci və 2-ci rəqəmlərinin cəmi 3-cü və 4-cü rəqəmlərinin cəminə bərabər olan sadə ədədləri tapmaq üçün proqram tərtib edin.

#### Çıxış verilənləri

Hər tapılan sadə ədəd ayrıca sətirdə çıxışa verilir.

### T3-2002-02. Məktəblər

İnformatika üzrə internet-olimpiadanın keçirilməsi məqsədilə Təhsil Nazirliyi Bakı şəhərindəki pilot məktəbləri etibarlı internetlə təmin etməyi qərara alır. Bunun üçün “InternetSchoolAz” provayderindən həmin məktəblərin birinə (həminin olması vacib deyil) telefon xətti çəkilməlidir. Eyni zamanda bəzi məktəblər arasında telefon əlaqəsi olmalıdır. Məktəb o zaman etibarlı internet bağlantısına malik olur ki, ya provayderlə, ya da etibarlı internet bağlantısı olan başqa məktəblə birbaşa əlaqəlidir. Bəzi məktəblər arasında telefon əlaqəsinin qiyməti məlumdur. Təhsil Nazirliyi məktəblərin əlaqə sxemlərindən ən ucuz başa gələn sxemi seçməyi qərara alır.

Məktəblərin əlaqə sxemlərindən iqtisadi cəhətdən səmərəli olan 2 sxemin dəyərini hesablamaq üçün proqram tərtib edin.

#### Giriş verilənləri (INPUT2.TXT faylından və ya klaviaturadan)

Birinci sətirdə bir-birindən boşluq simvolu ilə ayrılmış **N** və **M** kimi iki natural ədəd verilir:  $N$  ( $3 \leq N \leq 100$ ) – şəhərdəki pilot məktəblərin sayı, **M** isə onlar arasında mümkün əlaqələrin sayıdır.

Hər sonrakı sətirdə isə  $A_i$ ,  $B_i$ ,  $C_i$  kimi 3 (üç) ədəd verilir: burada  $C_i$  ( $1 \leq C \leq 300$ )  $A_i$  məktəbindən  $B_i$  ( $i = 1, 2, \dots, N$ ) məktəbinə çəkilən telefon xəttinin qiymətidir.

#### Çıxış verilənləri (OUTPUT2.TXT faylına və ya ekrana)

Çıxış sətirində ən ucuz başa gələn 2 sxemin dəyəri –  $S_1$  və  $S_2$  ( $S_1 \leq S_2$ ) bir-birindən boşluq simvolu ilə ayrılmış natural ədədlər şəklində verilir.  $S_1 = S_2$  halı da mümkündür.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 8 1 3 75 3 4 51 2 4 19 3 2 95 2 5 42 5 4 31 1 2 9 3 5 66	110 121

### T3-2002-03. Telefon

Rəqəmlər ardıcılığı telefon aparatının düymələrinin “şahmat” atının gedişi istiqamətində nömrə yığılması nəticəsində alınarsa, belə ardıcılıq “*şahmat*” *ardıcılığı* adlanır. Məsələn: 1616729 ardıcılığı “şahmat” ardıcılığıdır, 1183606 isə “şahmat” ardıcılığı deyil.

1	2	3
4	5	6
7	8	9
	0	

Verilmiş **K** ( $K \neq 0$ ) rəqəmi ilə başlayan və yığılması mümkün olan **N** uzunluqlu müxtəlif “şahmat” ardıcılıqlarının sayını hesablamaq üçün proqram tərtib edin.

#### Giriş verilənləri

Klaviaturadan bir sətirdə 2 (iki) ədəd: **K** – ardıcılığın ilk rəqəmi və **N** – ardıcılığın uzunluğu daxil edilir.

#### Çıxış verilənləri

Ekrana bir ədəd – axtarılan ardıcılıqların sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 7	168

### T3-2002-04. Yubiley

2100-cü ildə Bakı şəhərinin mərkəzində respublikanın inkişafında böyük xidmətləri olmuş informatika alimlərinin, bəstəkarların, şairlərin və başqalarının yubileylərinə (100, 110, 120 və s. illiyinə) qalan günlərin sayını göstərən lövhə (tablo) quraşdırılır. Hər yubiley 3 gün qeyd edilir. Bu lövhə üçün elə proqram tərtib etmək tələb olunur ki, anadan olma və cari təqvim gününə əsasən növbəti yubileyin neçə illik (100, 110, 120 və s.) olduğunu və bu məşhur tarixə neçə gün qaldığını təyin etsin.

**Qeyd.** Hesablama zamanı *uzun illəri* nəzərə almaq lazımdır. Uzun il elə ildir ki, onun nömrəsi ya 4-ə tam bölünür, ancaq 100-ə bölünmür, ya da 400-ə tam bölünür (məsələn: 1600, 2000 uzun illərdir, ancaq 1800, 1900 uzun illər deyil).

#### Giriş verilənləri (INPUT4.TXT faylından və ya klaviaturadan)

Birinci sətirdə gg.aa.iiii formatında məşhur adamların anadan olma tarixi, ikinci sətirdə isə həmin formatda cari tarix verilir.

#### Çıxış verilənləri (OUTPUT4.TXT faylına və ya ekrana)

Birinci sətirdə yubiley tarixləri – 100, 110, 120 və s. kimi verilir.



İkinci sətirdə yubileyə qalan günlər və ya əgər cari tarix 3 günlük yubiley günlərinə düşübsə, “Artıq bayram gedir” kimi məlumat verilməlidir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
31.08.1935 16.08.2145	210 15
31.08.1935 01.09.2145	210 Artıq bayram gedir

### T3-2002-05. Asfalt

Yolçəkənlər yolun müəyyən sahəsini ( $[0,1]$  parçasını) asfaltlayırlar. Onlar birinci gün  $[a_1, b_1]$ , 2-ci gün  $[a_2, b_2]$ , ...,  $n$ -ci gün  $[a_n, b_n]$  sahəsinə asfalt döşədilər. Yolun asfaltlanmamış sahəsini müəyyənləşdirən proqram tərtib edin (nəzərə alın ki, asfaltlanmış sahələr bir-birini örtə bilər).

**Giriş verilənləri** (INPUT5.TXT faylından və ya klaviaturadan)

Birinci sətirdə asfaltlanmış hissələri göstərən  $N$  ədədi, sonrakı  $N$  sətirdə isə asfaltlanmış hissələrin başlanğıcını və sonunu göstərən 2 (iki) ədəd verilir.

**Çıxış verilənləri** (OUTPUT5.TXT faylına və ya ekrana)

Birinci sətirdə asfaltlanmamış hissələrin sayını göstərən  $M$  ədədi, sonrakı  $M$  sətirdə isə asfaltlanmamış sahələrin başlanğıcını və sonunu göstərən 2 (iki) ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 0     0.2 0.25 0.47 0.3   0.6 0.7   1.0	2 0.2   0.25 0.6   0.7
3 0     0.3 0.2   0.6	0

### T3-2003-01. Uzun ədədlər

$n! + n^n$  ( $0 < n \leq 10000$ ) ədədinin  $m$ -ci rəqəmini tapmaq üçün proqram tərtib edin. Saymanı ( $m$ ) ən kiçik mərtəbədən (sağdan) başlamaq lazımdır.

**Açıqlama:**  $n! = 1 \times 2 \times 3 \times \dots \times n$

#### Giriş verilənləri (klaviaturadan)

Bir sətirdə aralarında boşluq simvolu olmaqla  $n$  və  $m$  ədədləri verilir.

#### Çıxış verilənləri

Bir sətirdə axtarılan ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
15 17	3

### T3-2003-02. Ən uzun ümumi söz

Verilmiş iki mətndə “ən uzun ümumi söz”ü (hər iki mətndə olan) tapmaq üçün proqram tərtib edin. Mətnlərdə sözlər boşluq, vergül və ya nöqtə ilə bir-birindən ayrılır.

Giriş verilənlərinin formatı	Çıxış verilənlərinin formatı
<b>n1</b> (birinci mətndəki sətirlərin sayı) 1-ci mətnin 1-ci sətiri 1-ci mətnin 2-ci sətiri ... 1-ci mətnin <b>n1</b> -ci sətiri <b>n2</b> (ikinci mətndəki sətirlərin sayı) 2-ci mətnin 1-ci sətiri 2-ci mətnin 2-ci sətiri ... 2-ci mətnin <b>n2</b> -ci sətiri	Ən uzun ümumi söz

### T3-2003-03. Cədvəl

Elementləri 0 və ya 1 olan  $N \times N$  ( $N < 100$ ) cədvəli verilmişdir. Həmin cədvəlin sətirində və ya sütununda, yaxud ixtiyari diaqonalında yerləşən və yalnız 1-lərdən ibarət ən uzun ardıcılığın uzunluğunu tapın.

#### Giriş verilənlərinin formatı

$N$  ədədi

1-ci sətir.  $N$  ədəd

2-ci sətir.  $N$  ədəd

...

$N$ -ci sətir.  $N$  ədəd

**Çıxış verilənlərinin formatı**

1-lərdən ibarət ən uzun ardıcılığın uzunluğu çıxışa verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
6 1 1 0 1 0 0 <b>1</b> 0 1 0 0 0 1 <b>1</b> 0 1 1 1 0 0 <b>1</b> 0 0 0 1 1 0 <b>1</b> 0 0 1 0 1 1 0 0	4

**İzahat.** Verilən nümunədə 1-lərdən ibarət axtarılan ardıcılıq  $A(2, 1)$  və  $A(5, 4)$  elementlərini birləşdirən ardıcılıqdır.

**T3-2003-04. Tənlik**

$n < 10000$  natural ədədi üçün  $xy + x + y = n$  tənliyinin ( $0 < x \leq y$ ) tam həllərini tapan proqram yazın. Əgər həllər bir neçədirsə, onda çıxışa onlardan ixtiyari birini, həll yoxdursa, “NO” sözünü verin.

**Giriş verilənləri**

$n$  ədədi

**Çıxış verilənləri**

$x$  və  $y$ -in qiymətləri

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
650	20 30

### T3-2003-05. Sadə ədədlər

$(n, m)$  intervalında yerləşən, birinci və ikinci rəqəmlərinin cəmi üçüncü və dördüncü rəqəmlərinin cəminə bərabər olan bütün sadə ədədləri tapın.

#### Giriş verilənlərinin formatı

$n$  və  $m$  ədədləri

#### Çıxış verilənlərinin formatı

1-ci sadə ədəd

2-ci sadə ədəd

3-cü sadə ədəd

...

Sonuncu sadə ədəd.

### T3-2004-01. Fibonaççi ədədləri

Fibonaççi ədədlər ardıcılığı belə təyin olunur:

$$F_0 = 0; F_1 = 1, F_n = F_{n-1} + F_{n-2}, n > 1.$$

Ardıcılığı mənfi indekslər üçün də müəyyən etmək olar. Məsələn, sıfıra yaxın mənfi indeksli Fibonaççi ədədləri bunlardır:

$n$	...	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	...
$F_n$	...	-8	5	-3	2	-1	1	0	1	1	2	3	5	8	...

Fibonaççi ədədləri içərisində verilmiş rəqəmlə qurtaran ədədlərin sayını tapmaq üçün proqram tərtib edin.

#### Giriş verilənləri (in1.txt faylı)

Bir sətirdə 3 tam ədəd:  $d$  ( $0 \leq d \leq 9$ ),  $a$  və  $b$  ( $-10^6 \leq a \leq b \leq 10^6$ ) ədədləri verilir.

#### Çıxış verilənləri (out1.txt faylı)

Çıxış verilənləri axtarılan ədədlərin sayını bildirən yeganə tam ədəddir. Bu ədəd sonuncu rəqəmi  $d$  olan və indeksləri  $a$  və  $b$  aralığında olan (özləri də daxil olmaqla) Fibonaççi ədədlərinin sayıdır.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1 -1 10	4

### T3-2004-02. Söz zənciri

Söz dedikdə boşluq simvolu daxil olmayan 1-dən 8-dək işarələr ardıcılığı (A-dan Z-dək böyük hərflər) başa düşülür.

$n$  sayda  $A_1, A_2, \dots, A_n$  sözləri daxil edilir. Bu sözləri elə nizamlamaq lazımdır ki, “zəncir” alınsın. Başqa sözlə,  $A_i$  sözünün birinci hərfi özündən əvvəlki sözün sonuncu hərfi ilə eyni olsun,  $A_i$  sözünün sonuncu hərfi isə özündən sonrakı sözün birinci hərfi ilə eyni olsun; uyğun olaraq axırındakı sözün sonuncu hərfi birinci sözün birinci hərfi ilə eyni olmalıdır. “Zəncir”ə təkrarlanma olmadan  $n$  sözün hamısı daxildir. Cavabı YES və ya NO şəklində verin. Əgər belə nizamlama mümkündürsə, söz zəncirinin özünü də çıxışa vermək lazımdır.

#### Giriş verilənləri (in2.txt faylı)

Giriş verilənlər  $n + 1$  sətirdən ibarətdir, burada  $n$  verilən sözlərin sayını bildirir. Birinci sətirdə  $n$  ədədi, sonrakı  $n$  sətirdə isə verilmiş sözlər yazılır.

#### Çıxış verilənləri (out2.txt faylı)

Birinci sətirdə YES və ya NO sözləri verilir. Birinci sətirdə YES sözü verilsə, sonrakı sətirlərdə söz zəncirinin özü də çıxışa verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
6	YES
AB	AB
CA	BD
DC	DC
BC	CB
CB	BC
BD	CA

**T3-2004-03. Maksimumlar cədvəli**

$A(N, N)$  cədvəlinə əsasən  $B(N, N)$  cədvəlini qurmaq lazımdır ( $N \leq 100$ ).  $B(I, J)$  elementi  $A$  cədvəlinin  $A(I, J)$  elementindən keçən diaqonallarının sol tərəfi ilə məhdudlaşan (şəkildə göstərildiyi kimi) elementlərinin maksimumuna bərabərdir.

	1	2	3	4	5	6	7	8	9	10	11
1	*	*	*						-		
2	*	*	*	*				-			
3	*	*	*	*	*		-				
4	*	*	*	*	*	*					
5	*	*	*	*	*		-				
6	*	*	*	*				-			
7	*	*	*						-		
8	*	*								-	
9	*										-
10											
11											

Burada  $B(4, 6)$  elementi  $A$  cədvəlinin yuxarıdakı cədvəldə  $*$  işarəsi ilə qeyd olunmuş elementlərinin maksimumudur.

**Giriş verilənləri** (in3.txt faylından)

Giriş verilənlər  $n+1$  sətirdən ibarətdir. Birinci sətirdə sətir və sütunların sayını bildirən  $n$  ədədi, sonrakı  $n$  sətirdə isə  $A$  cədvəli yerləşir.

**Çıxış verilənləri** (out3.txt faylından)

Çıxış verilənləri  $n$  sətirdən ibarət  $B$  cədvəlidir.

Giriş verilənlərinə nümunə				Çıxış verilənlərinə nümunə			
4				2	7	7	9
2	5	3	4	7	7	9	9
7	2	9	5	4	7	8	9
4	2	1	4	1	8	8	8
1	8	7	2				

### T3-2004-04. Ümumdünya konqresi

2150-ci ildə Bakı Qalaktikalararası Hava Limanı müxtəlif qalaktikalardan gələcək qonaqları Gənc Proqramçıların Qalaktikalararası Olimpiadasına qəbul etməyə hazırlaşır. Gələn kosmik gəmilər monitorların ekranlarında aşağıdakı kimi görünür.

Tip 1	Tip 2	Tip 3	Tip 4	Tip 5

Monitorun ekranında gəmilərin nə tərəfləri, nə də küncləri bir-birinə toxunmur. Monitora 0 və 1-dən ibarət  $N \times N$  ölçülü ( $N \leq 100$ ) cədvəl kimi baxaraq hava limanına yaxınlaşan kosmik gəmilərin tipini müəyyən edən proqram yazın. Burada 0 – boş fəzanı, 1 isə kosmik gəminin elementini göstərir. Gəmilər ekranda 90, 180, 270 dərəcə döna bilər.

Aşağıdakı şəkildə birinci tip 2 gəmi, 2-ci tip 1 gəmi, 3-cü tip 1 gəmi, 4-cü tip 1 gəmi vardır ( $N = 11$ ).

0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	1	0	0	0
0	1	0	0	0	0	1	1	1	0	0
0	0	0	0	0	0	0	1	0	0	0
1	0	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	1	0
1	0	0	0	0	0	1	1	1	1	1
0	0	1	0	1	0	0	0	0	1	0
0	0	1	1	1	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0

#### Giriş verilənləri (in4.txt faylı)

Giriş verilənləri  $N + 1$  sətirdən ibarətdir. 1-ci sətirdə cədvəlin sətir və sütunlarının sayı olan  $N$  ədədi, sonrakı  $N$  sətirdə isə cədvəlin özü yerləşir. Cədvəlin elementləri aralarında boşluq olmadan yazılır.

### Çıxış verilənləri (out4.txt faylı)

Çıxış verilənləri 5 sətirdən ibarətdir. 1-ci sətirdə 1-ci tip gəmilərin sayı, 2-ci sətirdə 2-ci tip gəmilərin sayı və nəhayət, 5-ci sətirdə 5-ci tip gəmilərin sayı yerləşir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
11 000000000000 111000000000 01000001000 01000011100 00000001000 100000000000 111000000010 10000011111 00101000010 00111000000 00101000000	2 1 1 1 0

### T3-2005-01. Zəng

İnformasiyalaşdırma layihəsinə görə 2005/2007-ci illərdə Azərbaycanın bütün məktəblərinə müasir kompüterlər veriləcək. Bu kompüterlərdən təkcə informatika dərslərində deyil, həm də o biri fənlərin tədrisində, məktəbin idarəolunmasında istifadə edilməlidir. Bakı şəhəri məktəblərindən birinin şagirdləri məktəb zəngini idarə edən “Taymer” proqramını hazırlamaq istəyirlər. Məktəbdə zəngin hansı vaxtda vurulmasını müəyyən edən bu proqramı tərtib etməkdə onlara kömək edin.

### Giriş verilənləri (a.in faylı)

Birinci sətirdə cari vaxt HH:MM:SS formatında (aparıcı sıfırlar olmaqla) verilir. Belə ki, HH – 00-dan 23-ə, MM və SS isə 00-dan 60-a qədər qiymətlər ala bilər.

İkinci sətirdə ölçülməli olan vaxt intervalı H:M:S formatında yazılır (burada H, M və S aparıcı sıfırları olmayan, 0-dan  $10^9$ -a qədər ədədlər ola bilər). Əlavə olaraq əgər M = 0 isə (və ya M = 0 və M = 0), onda onlar yazılmaya bilər. Məsələn, 100:60 formatı 100 dəqiqə 60 saniyəni göstərir, onu 101:0 və ya 1:41:0 kimi göstərmək də olar. Tək 42 yazılışı 42 saniyəni göstərir. 100:100:100 isə 100 saat 100 dəqiqə 100 saniyəni göstərir və onu 101:41:40 kimi də ifadə etmək olar.

### Çıxış verilənləri (a.out faylı)

Ekranə səs signalının səsləndiriləcəyi vaxt HH:MM:SS formatında verilir. Bundan başqa, əgər signal cari gündə səslənməyəcəksə, sona +<miqdar>days yazılışı



əlavə edilməlidir. Məsələn, əgər signal növbəti gün səsləndiriləcəksə, onda +1 days yazılışı əlavə edilməlidir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
01:01:01 48:0:0	01:01:01+2 days
01:01:01 58:119	02:01:00
23:59:59 1	00:00:00+1 days

**T3-2005-02. Sıxma** (tut: T3-2014-01)

Yalnız latın əlifbasının baş hərflərindən ibarət simvollar sətrinə baxaq. Məsələn, “AAAABBBBAAAABBC” belə sətirlərdən biridir. Onun uzunluğu 15 simvoldan ibarətdir. Bu sətirdə yalnız hərflər olduğundan ardıcıl gələn eyni hərfləri onların sayını bildirən ədəd və həmin bir hərflə əvəzləmək olar. Ədəd hərfdən öndə yazılır və həmin hərfin təkrarlanma sayını bildirir. Məsələn, yuxarıda verilən sətri “4A3B4A3BC” kimi yazmaq olar və alınan belə sətrin uzunluğu yalnız 9 simvoldan ibarət olacaq. Asanlıqla görmək olar ki, burada “4A3B” qrupu ardıcıl olaraq 2 dəfə təkrarlanır. Bu təkrarlanmanı yazmaq üçün mötərizədən istifadə edək və mötərizənin qarşısında təkrarolunmanın sayını yazaq: “2 (4A3B) C”. Bu ardıcılığın uzunluğu artıq 8-ə bərabərdir. Təkrarlanan qruplar bir neçə səviyyəli də ola bilər. Məsələn, “ABABCABABC” ardıcılığını “2 (2 (AB) C)” kimi yazmaq olar.

İlkin verilən simvollar sətrindən sıxılmış sətri tapan proqram yazın.

**Giriş verilənləri** (B.IN faylı)

Uzunluğu 255 simvolu aşmayan simvollar sətri verilir.

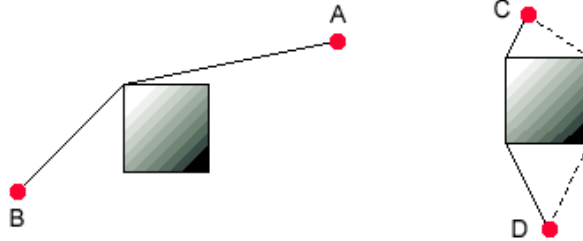
**Çıxış verilənləri** (B.OUT faylı)

Verilmiş sətir sıxılmış formada çıxışa verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
AABCCCC	2AB5C
EEEEEEEEEEEEEEEEEEEEUU	21EU
AAAAAAAAAABABABCCD	10A2 (BA) B2CD

### T3-2005-03. İp

Müstəvi üzərində kvadrat və 2 nöqtə verilir. Göstərilən nöqtələr kvadratin daxilində yerləşə bilməz. Bu nöqtələr kvadratin xaricində və ya onun tərəfləri üzərində ola bilər. Nöqtələr arasında ip çəkmək lazımdır. İp kvadratin daxilindən keçə bilməz, yalnız onun sərhədinə toxuna bilər. İp kifayət qədər nazikdir və onun qalınlığı nəzərə alınmaya bilər.



Kvadratin mərkəzinin  $(X_m, Y_m)$  koordinatları və bir tərəf nöqtəsinin  $(X_t, Y_t)$  koordinatları verilir. Nöqtələr də koordinatları ilə verilir:  $(X_a, Y_a)$ ,  $(X_b, Y_b)$ . Bütün ədədlər:  $X_m, Y_m, X_t, Y_t, X_a, Y_a, X_b, Y_b$  tam ədədlərdir və modulca 10000-i aşmır. Kvadratin tərəfləri koordinat oxlarına paraleldir. Şəkildə uzunluğu minimum olan ipin çəkilməsinə iki misal verilmişdir. Qırıq-qırıq xətlə ipin düzgün olmayan çəkilməsi göstərilmişdir.

Kvadratin ipin düz xətt boyunca çəkilməsinə mane olacağını nəzərə alaraq iki nöqtəni birləşdirən ipin mümkün olan ən qısa uzunluğunu tapan proqram yazın.

#### Giriş verilənləri (C.IN faylı)

Bir sətirdə ədədlər verilir:  $X_m, Y_m, X_t, Y_t, X_a, Y_a, X_b, Y_b$

#### Çıxış verilənləri (C.OUT faylı)

Bir ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
0 0 -1 -1 -1 -2 2 1	4.47

### T3-2005-04. Ardıcılıq

Araşdırılan ardıcılıq ədədlər parçalarının birləşməsindən yaradılır: birinci parça – 1; ikinci parça – 12; üçüncü parça – 123; dördüncü parça – 1234 və s. Nəticədə aşağıdakı rəqəmlər ardıcılığı alınır:

11212312341234512345612345671234567812345678912345678910...

Verilmiş  $N$  ( $1 < N < 10^9$ ) üçün ardıcılıqdakı  $N$ -ci rəqəmi tapan proqram tərtib edin.

#### Giriş verilənləri (D.IN faylı)

Bir sətirdə  $N$  ədədi verilir.

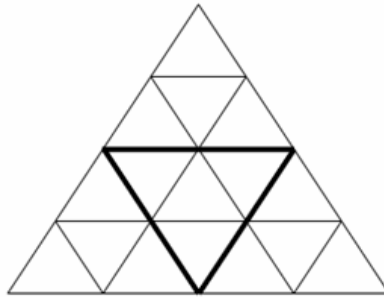
#### Çıxış verilənləri (D.OUT faylı)

Araşdırılan ardıcılıqda  $N$ -ci yerdə duran rəqəm çıxışa verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
10	4

### T3-2006-01. Üçbucaqlıların sayı

Aşağıda verilmiş fiqura (kiçik bərabərtərəfli üçbucaqlılardan təşkil edilmiş böyük bərabərtərəfli üçbucaqlıya) bənzər fiqura baxaq. Şəkildə 4 səviyyədə üçbucaqlılardan təşkil edilmiş fiqur verilmişdir.



Böyük üçbucaqlıda olan bütün üçbucaqlıların sayını müəyyənləşdirən proqram yazmaq lazımdır (təkcə “kiçik” üçbucaqlıları deyil, həm də bütün üçbucaqlıları, o cümlədən qalın xətlə verilmiş üçbucaqlını, bütövlükdə fiqurun özünü də nəzərə almaq lazımdır).

#### Giriş verilənləri

Giriş faylında bir  $N$  ədədi – fiqurdakı səviyyənin sayı yazılmışdır ( $1 \leq N \leq 100000$ ).

#### Çıxış verilənləri

Çıxışa bu fiqurdakı üçbucaqlıların sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1	1
2	5
4	27

### **T3-2006-02. Saatların yoxlanılması**

“MicroHard” şirkətində son vaxtlar əməkdaşların işə müxtəlif vaxtlarda gəlməsi halları birdən-birə artmışdı. Vəziyyəti təhlil edən rəhbərlik belə nəticəyə gəlir ki, buna səbəb əməkdaşların qol saatlarındakı göstəricilərin müxtəlifliyidir. Rəhbər heyətlə əlavə məsləhətləşmədən sonra qərara alınır ki, bütün əməkdaşlar saatlarını eyni vaxta qursun (hansı vaxta qurmağın əhəmiyyəti yoxdur).

Şirkətin bütün əməkdaşları eyni nümunədən olan elektron qol saatından istifadə edirlər. Həmin saatlarda vaxt aşağıdakı formatda göstərilir: HH:MM:SS (burada HH – saat, MM – dəqiqə, SS – saniyə həmişə iki rəqəmlə təsvir olunur və  $00 \leq HH \leq 23$ ,  $00 \leq MM \leq 59$ ,  $00 \leq SS \leq 59$ ). Saatların qurulması iki düymənin köməyi ilə edilir. Birinci düymə redaktə sahəsini aşağıdakı qaydada dəyişir: birinci dəfə basıldıqda saat vaxtın göstərilməsi rejimindən HH sahəsinin redaktəsi rejiminə, ikinci dəfə basıldıqdan sonra MM sahəsinin redaktəsi rejiminə, üçüncü dəfə basıldıqdan sonra isə SS sahəsinin redaktəsi rejiminə keçir və dördüncü dəfə basıldıqdan sonra vaxtın göstərilməsi rejiminə qaydır və proses beləcə təkrarlanır. İkinci düymənin hər dəfə basılması redaktə olunan sahənin bir vahid artmasına səbəb olur (vaxtın göstərilməsi rejimində heç bir dəyişiklik baş vermir). Saniyələr dolduqdan sonra SS sahəsi sıfırlanır və MM bir vahid artır, dəqiqələr dolduqdan sonra MM sahəsi sıfırlanır və HH sahəsi bir vahid artır, saatlar dolduqdan sonra isə, sadəcə, HH sahəsi sıfırlanır.

Hər şey yaxşı olmalı idi, ancaq anadangəlmə tənbel olduqlarından işçilər saatların qurulmasında düymələrin toplam basılma sayını minimallaşdırmaq istəyirlər. Saatların qurulmasından sonra, eləcə də başlanğıcda bütün saatlar vaxtın göstərilməsi rejimində olmalıdır.

Bütün əməkdaşların saatlarını eyni vaxta qurmağa kifayət etməsi üçün düymələrin minimal toplam basılma sayını müəyyən edən proqram tərtib edin.

### **Giriş verilənləri**

Giriş faylının birinci sətirində şirkətin əməkdaşlarının sayını göstərən  $N$  ( $1 \leq N \leq 200$ ) natural ədədi yerləşir. Sonrakı  $N$  sətirdə isə hər bir əməkdaşın qol saatının “HH:MM:SS” formatında vaxt göstəricisi verilir.

### **Çıxış verilənləri**

Çıxış faylında bir ədəd – minimal toplam basılma sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2 08:01:01 07:59:00	7

### T3-2006-03. Sadə olmayan məsələ

$m$  sətir və  $n$  sütundan ibarət düzbucaqlı cədvəl verilmişdir.  $i$ -ci sətir ilə  $j$ -ci sütunun kəsişməsində  $a_{ij}$  tam ədədi yerləşir.

Cədvəlin elə 4 müxtəlif xanasını tapmaq tələb olunur ki, həmin xanaların mərkəzləri tərəfləri cədvəlin tərəflərinə paralel olan düzbucaqlının təpə nöqtələri olsun, bu xanalarda yazılmış ədədlərin cəmi isə maksimum olsun.

1	1	1	1	1
1	2	1	1	1
1	1	1	1	1
1	1	1	3	1
1	1	1	1	1

#### Giriş verilənləri

Birinci sətirdə iki  $m$  və  $n$  natural ədədi yazılır ( $2 \leq m, n \leq 500$ ). Sonrakı sətirlərdə isə cədvəl təsvir edilir – hər birində  $n$  tam ədəd olan ( $10^7 \leq a_{ij} \leq 10^7$ )  $m$  sütun verilir.

#### Çıxış verilənləri

Çıxış faylının birinci sətirində seçilmiş elementlərin maksimal cəmi olan  $r$  ədədi, ikinci sətirdə seçilmiş xanadan uyğun olaraq sol yuxarıda və sağ aşağıda yerləşən xananın koordinatları olan dörd natural  $i_1, j_1, i_2, j_2$  ədədi verilir ( $1 \leq i_1 < i_2 \leq m, 1 \leq j_1 < j_2 \leq n$ ). Əgər optimal həll bir neçə olarsa, onlardan ixtiyari birini seçmək lazımdır.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 5 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 3 1 1 1 1 1 1	7 2 2 4 4
5 5 1 -1 -1 -1 -1 -1 -2 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -3 -1 -1 -1 -1 1 -1	0 1 1 5 4

### **T3-2006-04. Çeşidləmə stansiyası**

Böyük dəmir yolu stansiyasının dispetçeri hər gün vaqonların lazım olan yerlərə getməsi üçün çoxlu sayda qatarın vaqonlarının yerini dəyişməli olur. Bunun üçün dispetçer stansiyaya gələn qatarı ixtiyari yerlərdən açar və yaranan bir və ya bir neçə vaqondan ibarət qoşquların ixtiyari qaydada yerini dəyişə bilər. Bir qoşqudakı vaqonların ardıcılığını dəyişmək olmaz, eləcə də bütün qoşqunu elə döndərmək olmaz ki, qoşquda axırda olan vaqon birinci yerə düşsün.

Qatarda vaqonların yerini tələb olunan qaydada dəyişmək üçün vaqonlar arasında hansı minimal sayda birləşmələrin açılmalı olduğunu müəyyənləşdirməkdə dispetçərə kömək etmək lazımdır.

#### **Giriş verilənləri**

Giriş faylının birinci sətirində  $N$  tam ədədi yerləşir ( $1 \leq N \leq 100$ ). İkinci sətirdə 1-dən  $N$ -dək ədədlərin yerdəyişməsi, yəni 1-dən  $N$ -dək bütün natural ədədlər hər hansı ardıcılıqla verilir. Ədədləri bir-birindən boşluq simvolu ayırır. Bu ardıcılıq stansiyaya gələn qatardakı vaqonların nömrəsini göstərir. Tələb olunur ki, stansiyadan yola düşəcək qatardakı vaqonlar öz nömrələrinə uyğun ardıcılıqla düzülsün.

#### **Çıxış verilənləri**

Çıxış faylında yeganə tam ədəd – qatarda vaqonların yerini tələb olunan qaydada dəyişmək üçün vaqonlar arasında açılmalı olan birləşmələrin minimal sayına bərabər olan ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə	Qeyd
4 3 1 2 4	2	3 və 1, 2 və 4 vaqonlarını açmaq lazımdır.
5 5 4 3 2 1	4	5 və 4, 4 və 3, 3 və 2, 2 və 1 vaqonlarını açmaq lazımdır.

### **T3-2006-05. Ədəd**

Onluq say sistemindəki yazılışında yalnız 1, 3, 7 rəqəmləri olan natural ədədlər ardıcılığının bütün elementləri artma sırası ilə nömrələnmişdir. Verilmiş  $N$ -ə görə həmin ardıcılığın  $N$ -ci elementini tapın.

#### **Giriş verilənləri**

Giriş faylının birinci sətirində axtarılan ədədin nömrəsi olan  $N$  ədədi ( $1 \leq N \leq 1000000$ ) verilir.

#### **Çıxış verilənləri**

Çıxışa axtarılıb tapılan bir ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3	7
9	37
13	111

### T3-2007-01. Proqramçılar

Programland ölkəsində  $N$  proqramçı yaşayır. Bütün proqramçılar 1-dən  $N$ -dək nömrələnib və 1 nömrəli proqramçı *Superproqramçıdır*.

Hər səhər proqramçılar (hamısı olmaya da bilər) superkompüterin ətrafında toplaşirlar. Əgər bu toplantıya Superproqramçı gəlsə, o, bütün iştirakçılara bir yeni proqram göstərir. Əgər Superproqramçı toplantıya gəlməyibsə, onda bütün iştirakçılar artıq bildikləri proqramları göstərlər. Sizin tərtib etdiyiniz proqram toplantıda iştirak edən proqramçıların siyahısından Superproqramçının göstərdiyi bütün proqramları bilən proqramçıları müəyyənləşdirməlidir.

#### Giriş verilənləri (a.in faylı)

Birinci sətirdə proqramçıların ümumi sayı və toplantıların sayını göstərən iki  $N$  ( $1 \leq N \leq 100$ ) və  $M$  ( $1 \leq M \leq 50$ ) ədədi verilir.

Sonrakı  $M$  sətirdə hər toplantıya gələn proqramçıların siyahısı verilir. Bu sətirlərin hər biri toplantıya gələn proqramçıların sayı olan  $K$  tam ədədi ilə başlayır. Onun ardı ilə isə aralarında boşluq simvolu olmaqla toplantıda iştirak edən proqramçıların nömrələri olan tam ədədlər verilir.

#### Çıxış verilənləri (a.out faylı)

Çıxış faylının yeganə sətirində Superproqramçının göstərdiyi bütün proqramları bilən proqramçıların nömrələri verilir. Bu nömrələr artma sırası ilə verilməlidir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 3 2 1 2 3 2 3 4 3 4 2 1	1 2 4
8 5 4 1 3 5 4 2 5 6 3 6 7 8 2 6 2 4 2 6 8 1	1 2 6 8
5 3 2 1 3 2 2 1 4 2 1 4 5	1

### T3-2007-02. Simmetrik ardıcılıqlar

Sağdan sola və soldan sağa eyni oxunan ədədlər ardıcılığını *simmetrik ardıcılıq* adlandıraraq. Məsələn, aşağıdakı ardıcılıqlar simmetrikdir:

1 2 3 4 5 4 3 2 1

1 2 1 2 2 1 2 1

Elə program yazın ki, verilmiş ədədlər ardıcılığının simmetrik olması üçün onun sonuna minimal sayda və hansı ədədləri əlavə etməyi müəyyənləşdirsin.

#### Giriş verilənləri (b.in faylı)

Giriş faylında əvvəlcə ilkin ardıcılıqdakı elementlərin sayını bildirən  $N$  ədədi yazılır. Sonra bu ardıcılığın elementləri olan  $N$  sayda ( $1 \leq N \leq 100$ ) ədəd verilir. Ardıcılığın elementləri 1-dən 9-dək natural ədədlərdir.

#### Çıxış verilənləri (b.out faylı)

Birinci sətirdə ardıcılığın sonuna yazılmalı olan ədədlərin  $M$  minimal sayı, ikinci sətirdə isə əlavə ediləcək  $M$  sayda ədəd (hər biri 1-dən 9-dək ədəd ola bilər) yazılır.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
9 1 2 3 4 5 4 3 2 1	0
5 1 2 1 2 2	3 1 2 1
5 1 2 3 4 5	4 4 3 2 1

### T3-2007-03. Ədədin göstərilməsi

Verilmiş  $N$  ədədinə görə qiyməti həmin ədədə bərabər olan ədədi ifadə tərtib edən program yazmaq lazımdır. Bu ifadədə verilmiş  $K$  ədədindən çox olmayan sayda natural ədəd, toplama və vurma əməlləri, eləcə də mötərizələrdən istifadə edilə bilər.

#### Giriş verilənləri (c.in faylı)

Giriş faylının birinci sətirində iki natural ədəd verilir:  $N$  – ifadənin qiyməti ( $1 \leq N \leq 10000$ ) və  $K$  – ifadədə istifadə edilə bilən ədədlərin maksimal sayı ( $1 \leq K \leq 10000$ ).

#### Çıxış verilənləri (c.out faylı)

Çıxış faylının birinci sətirində minimal sayda simvollarla yazılmış ifadənin uzunluğunun ədədi qiyməti, ikinci sətirdə isə ifadənin özü verilir. Əgər həllərin sayı çoxdursa, onda çıxışa onlardan ixtiyari birini verin.



**Qeyd.** İfadənin uzunluğu hesablanarkən bütün simvollar – rəqəmlər, əməl işarələri, mötərizələr nəzərə alınır.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
7 3	5 3+1+3
15 20	2 15
176 1	41 (1+1+1+1) * (1+1+1+1) * (1+1+ (1+1+1) * (1+1+1) )

### T3-2007-04. Yol

Başlanğıcda oyunçu  $N \times M$  düzbucaqlı cədvəlinin sol yuxarı küncdəki xanasında dayanır. Bir gedişdə ona yalnız sağdakı və ya aşağıdakı qonşu xanaya getməyə icazə verilir (soldakı və yuxarıdakı xanaya getmək qadağandır).

Oyunçunun sağ aşağı küncdəki xanaya çatması üçün mümkün olan yolların sayını hesablayın.

**Giriş verilənləri** (d. in faylı)

Bir sətirdə cədvəlin ölçüləri olan  $N$  və  $M$  ədədləri verilir ( $1 \leq N \leq 15$ ,  $1 \leq M \leq 15$ ).

**Çıxış verilənləri** (d. out faylı)

Çıxışa axtarılan yolların sayını göstərən ədəd verilir.

#### Açıqlama

Əgər verilmiş cədvəl 2 sətir və 3 sütundan ibarətdirsə, onda sol yuxarı küncdəki xanadan sağ aşağı küncdəki xanaya getmək üçün aşağıdakı yollar var:

1) aşağı, sağa, sağa; 2) sağa, aşağı, sağa; 3) sağa, sağa, aşağı.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2 3	3
3 3	6

### T3-2007-05. Sola dönmələr

Avtomobilin hərəkət marşrutu sınıq xəttin təpə nöqtələrinin koordinatları şəklində verilmişdir. Avtomobil sınıq xəttin birinci təpə nöqtəsindən hərəkətə başlayır. Sola dönmələrin sayını tapmaq tələb olunur (sınıq xəttin qonşu hissələri bir düz xətt üzərində yerləşmir).

**Giriş verilənləri** (e.in faylı)

Giriş faylının birinci sətiri sınıq xəttin tərəflərinin sayını göstərən bir  $N$  ədədindən ( $1 \leq N \leq 100$ ) ibarətdir. Sonrakı sətirlərdə isə sınıq xəttin təpə nöqtələrinin koordinatlarını göstərən ədədlər cütü verilir.

**Çıxış verilənləri** (e.out faylı)

Çıxış faylına bir ədəd – sola dönmələrin sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 1 1 2 2 3 2 3 3 2 3	2

**T3-2008-01. Yuvarlaq ədədlər**

Ədədin yazılışında yalnız 0 və 5 rəqəmləri olarsa, həmin ədədi *yuvarlaq ədəd* adlandıracağıq. Yuvarlaq ədədləri artan sıra ilə yazsaq, belə bir ardıcılıq alınar: 0, 5, 50, 55, 500, 505, ...

Yuvarlaq ədədlər ardıcılığında  $K$ -cı yerdə duran ədədi tapmaq üçün proqram tərtib edin.

**Giriş verilənləri** (numbers.in faylı)

$K$  ( $0 < K < 500$ ) natural ədədi – ardıcılıqda yuvarlaq ədədin durduğu yerin nömrəsi.

**Çıxış verilənləri** (numbers.out faylı)

Tələb olunan yuvarlaq ədəd ekrana çıxarılır.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2	5
6	505

**T3-2008-02. İki kseroks**

Bu gün səhər münisiflər heyəti olimpiada məsələlərinə daha bir variant əlavə etməyi qərara aldı. Təşkilat Komitəsinin məsul katibi onun şərtlərini bir nüsxədə çap etdi. O, olimpiadanın başlanmasınadək daha  $N$  nüsxə çıxarmalı idi. Onun sərəncamında biri bir vərəqi  $x$  saniyədə, digəri isə  $y$  saniyədə çıxaran iki üz çıxaran aparat – kseroks var. (Kserokslardan həm tək-tək, həm də eyni anda birlikdə istifadə etməyə icazə verilir. Təkcə orijinaldan deyil, həm də kopyadan da üz çıxarmaq olar). Bunun üçün tələb olunan ən az vaxtı hesablayan proqram tərtib edin.

**Giriş verilənləri** (xerox.in faylı)

Giriş faylında bir-birindən boşluq simvolu ilə ayrılmış üç natural ədəd –  $N, x, y$  verilir ( $1 \leq N \leq 2 \times 10^8, 1 \leq x, y \leq 10$ ).

**Çıxış verilənləri** (xerox.out faylı)

Çıxışa bir ədəd –  $N$  sayda kopiyanı almaq üçün lazım olan minimum vaxt (saniyə ilə) verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 1 1	3
5 1 2	4

**T3-2008-03. Yanacaqoldurma məntəqələri**

“Azoil” şirkəti Bakı–Lənkəran magistral yolunda  $N$  sayda yanacaqoldurma məntəqəsi tikmişdir. Hər məntəqənin öz nömrəsi var və məntəqələr tikildiyi zamandan asılı olaraq 1-dən başlayaraq nömrələnib. Bundan başqa, hər məntəqə yolun müəyyən kilometrində yerləşir. Yolda kilometrələr Bakıdan başlayaraq 0-dan nömrələnir. Sonradan məlum olub ki, iqtisadi cəhətdən sərfəli olmadığından məntəqələrdən birini ləğv etmək lazımdır. Sürücülərin rahatlığı üçün elə yanacaqoldurma məntəqəsini qapatmaq lazımdır ki, həmin məntəqənin magistral yol boyunca qonşu məntəqə ilə arasındakı məsafə ən az (minimum) olsun.

Qapadılacaq yanacaqoldurma məntəqəsini müəyyənləşdirmək üçün proqram yazın.

**Giriş verilənləri** (avto.in faylı)

Birinci sətirdə yanacaqoldurma məntəqələrinin  $N$  ( $2 \leq N \leq 10^5$ ) sayı, ikinci sətirdə isə  $N$  sayda müxtəlif  $x_i$  tam ədədləri ( $x_i$  ədədi  $i$  nömrəli məntəqənin yerləşdiyi kilometri göstərir) verilir ( $1 \leq i \leq N$ ) ( $0 \leq x_i \leq 10^9$ ). Sətirdəki ədədlər boşluq simvolu ilə ayrılır.

**Çıxış verilənləri** (avto.out faylı)

Qapadılacaq məntəqənin nömrəsi çıxışa verilməlidir. Bir neçə cavab olduqda onlardan ixtiyari birini vermək olar.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 10 3 7 2 5	2

### T3-2008-04. Artıq yük

Tural həyatında ikinci dəfə idi ki, proqramlaşdırma üzrə dünya olimpiadasına gedirdi. Hava limanında aydın oldu ki, birinci dəfədə olduğu kimi, onun çantasının çəkisi icazə verilən normadan artıqdır.

Turalın çantasındakı  $N$  sayda əşyanın hər birinin öz çəkisi ( $W_i$  kiloqram) və öz dəyəri ( $A_i$  manat) var. Eyni zamanda istənilən əşya üçün aşağıdakı bərabərsizlik ödənilir:

$$W_1 + W_2 + \dots + W_{i-1} \leq W_i$$

Turala bildirildi ki, cəmi çəkisi  $M$ -dən az olmamaqla hansısa əşyaları hava limanında qoymalıdır. Tural minimal itki vermək istəyir, ona görə də hava limanında qoyacağı əşyaların ümumi qiyməti mümkün variantlar içərisində ən az olmalıdır.

Saxlanılan əşyaların mümkün qiymətlərinin minimumunu hesablamaq üçün proqram yazmaq tələb olunur.

#### Giriş verilənləri (baggage.in faylı)

Birinci sətirdə çantadakı əşyaların miqdarını bildirən  $N$  ədədi ( $1 \leq N \leq 30$ ) və artıq kiloqramı göstərən  $M$  ədədi ( $1 \leq M \leq 10^{18}$ ) verilir. İkinci sətirdə  $N$  sayda mənfi olmayan tam ədəd – bütün əşyaların  $W_i$  çəkisi verilir və bu ədədlərin cəmi  $10^{18}$ -i aşmamalıdır. Üçüncü sətirdə  $N$  sayda mənfi olmayan tam ədəd – bütün əşyaların  $A_i$  qiymətləri verilir və bütün ədədlər  $10^9$ -u aşmamalıdır.

#### Çıxış verilənləri (baggage.out faylı)

Çıxış faylında Turalın məcburi olaraq hava limanında qoyacağı əşyaların minimal toplam dəyərini vermək tələb olunur.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 15 5 10 15 30 1 5 3 6	3
3 2 1 2 4 7 6 5	5

### T3-2008-05. Daşınma

Onestreet şəhəri bir düz küçədən ibarətdir. Şəhərdə malların ünvanlara çatdırılması ilə bir şirkət məşğul olur. Rahatlıq üçün çatdırılma ünvanları şirkətin ofisindən olan məsafəni göstərən ədəd şəklində verilir. Müsbət ədədlər bir istiqaməti, mənfi ədədlər isə əks istiqaməti göstərir. Şirkət daşınmanı sifarişlərin verildikləri ardıcılığa uyğun yerinə yetirir.

Şirkətdə iki kuryer işləyir. Günün başlanğıcında sifarişlər onlar arasında bölüşdürülür və onların hər biri öz marşrutu ilə yola düşür. Şirkət sifarişlərin bölüşdürülməsini elə planlaşdırmalıdır ki, axırncı sifərişi yerinə yetirəndən sonra kuryerlərin qət etdikləri toplam məsafə minimum olsun.

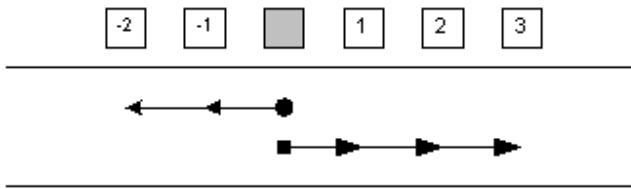
Kuryerlərin ofisdən çatdırılma ünvanlarınadək getdikləri məsafələrin cəminin ən az olmasını tapmaq üçün proqram yazın.

#### Giriş verilənləri (order.in faylı)

Birinci sətirdə sifarişlərin sayını göstərən  $N$  ədədi ( $1 \leq N \leq 10000$ ), sonrakı  $N$  sətirin hər birində bir tam ədəd – ofisdən çatdırılma ünvanınadək olan məsafə verilir. Əgər məsafə müsbətdirsə, çatdırılma ünvanı şirkətin ofisindən şəhərin bir istiqamətində, mənfidirsə, əks istiqamətindədir. Məsafələrin mütləq qiyməti  $10^8$ -i aşmır.

#### Çıxış verilənləri (order.out faylı)

Çıxış faylının yeganə sətirində bir tam ədəd – şirkət işçilərinin hər ikisinin cəmi qət etdikləri mümkün olan ən kiçik məsafə verilir.



Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 1 -1 2 -2 3	5

### T3-2009-01. Kəsir

Məmməd riyaziyyatı çox sevir, ona görə də düzgün ixtisar olunmayan kəsrlərə aid müxtəlif məsələlər düşünüb və həll etməklə o, evdə uzun müddət eksperimentlər aparır. Onu aşağıdakı məsələ çox maraqlandırır: surət və məxrəcinin cəmi  $n$ -ə bərabər olan ən böyük düzgün ixtisar olunmayan kəsri tapmaq. Bu məsələni həll etməkdə ona kömək edin.

#### Giriş verilənləri

Bir tam  $n$  ( $3 < n < 1000$ ) ədədi daxil edilir.

#### Çıxış verilənləri

Çıxışa axtarılan kəsrin surət və məxrəci aralarında boşluq simvolu olmaqla verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
10	3 7
23	11 12

### T3-2009-02. Şahmat

Şəhriyar şahmat oynamağa yeni başlayıb. Xatırladaq ki, şahmatı iki oyunçu oynayır, hər bir oyunçunun başlanğıcda 8 fiquru və 8 piyadası olur. Bu məsələdə piyadalara baxılmayacaq. Atdan başqa heç bir fiqur rəqibin və ya öz fiqurlarının üstündən tullanıb keçə bilməz. Oyunçu öz fiquru olan xanaya gediş edə də bilməz. Rəqibin fiquru olan xanaya gediş zamanı isə rəqibin fiquru şahmat taxtasından uzaqlaşdırılır.

Fiqurlar aşağıdakı qaydada gediş edə bilər:

- şah – üfüqi, şaquli və ya diaqonal istiqamətdə istənilən qonşu xanaya;
- vəzir – üfüqi, şaquli və ya diaqonal istiqamətdə istənilən xanaya;
- top – üfüqi və ya şaquli istiqamətdə istənilən xanaya;
- fil – diaqonal istiqamətdə istənilən xanaya;
- at – “L” hərfi formasında: 1 xana üfüqi və 2 xana şaquli, yaxud əksinə, 1 xana şaquli, 2 xana üfüqi.

Sizə bir ağ və bir qara fiqurun yerləşdiyi xana verilir. Bu fiqurların bir-birini vurub-vurmadığını və vururlarsa, hansının hansını vurduğunu müəyyənləşdirin.

#### Giriş verilənləri

Birinci sətirdə ağ fiqurun, ikinci sətirdə isə qara fiqurun tipi və yerləşdiyi xana verilir. Hər fiqur bir sətirdə uç simvolla verilir. Birinci simvol fiqurun tipini göstərir: “B” – fil, “N” – at, “R” – top, “Q” – vəzir, “K” – şahı göstərir. İkinci

simvol fiqurun üfûqi istiqamətdəki (“a”, “b”, “c”, “d”, “e”, “f”, “g”, “h”), üçüncü simvol isə şaquli istiqamətdəki (“1”-dən “8”-dək ədədlər) yerini göstərir. Fiqurların şahmat lövhəsində müxtəlif damalarda olmasına təminat verilir.

**Çıxış verilənləri**

Ekrana bir söz – məsələnin cavabı verilir.

Əgər fiqurlar bir-birini vura bilmirsə, “NONE” sözü verilir.

Əgər hər iki fiqur bir-birini vura bilirsə, “BOTH” sözü verilir.

Əgər ağ fiqur qara fiquru vurursa, ancaq qara fiqur ağ fiquru vura bilmirsə, “WHITE” sözü verilir.

Əgər qara fiqur ağ fiquru vurursa, ancaq ağ fiqur qara fiquru vura bilmirsə, “BLACK” sözü verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
Ka1 Rg1	BLACK
Qf3 Qh5	BOTH

**T3-2009-03. Palindrom (bax: T3-2013-02)**

Hər iki istiqamətdə eyni cür oxunan söz *palindrom* adlanır. Məsələn “rayar” sözü palindromdur. Söz üçün dövrü sürüşdürməni belə təyin etmək olar: sözün sonundakı hərflərin bir hissəsi (bir hərf də ola bilər) olduqları ardıcılıqla sözün başlanğıcına keçirilir. Məsələn, “rayar” sözü “array” sözünün dövrü sürüşdürməsidir (“ray” hərflərini başlanğıca keçirmək lazımdır).

Əgər sözün palindrom olan dövrü sürüşdürməsi olarsa, onu *dövrü palindrom* adlandıracağıq. Məsələn, “array” sözü dövrü palindromdur, çünki onun dövrü sürüşdürməsi olan “rayar” palindromdur.

Latın əlifbasının 100-dən artıq olmayan sayda hərflərindən ibarət söz verilmişdir. Bu sözün dövrü palindrom olub-olmadığını yoxlamaq lazımdır.

**Giriş verilənləri**

Girişdə bir sətirdə N sayda ( $N \leq 100$ ) latın hərfindən ibarət bir söz verilir.

**Çıxış verilənləri**

Əgər verilən söz dövrü palindromdursa, ekrana “YES”, əks halda “NO” sözü verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
array	YES
computer	NO
sis	YES

### T3-2009-04. Doğru ardıcillıq

Bir və sıfırlardan ibarət cüt uzunluqlu ardıcillıqda cüt yerdə duran 1-lərin sayı tək yerdə duranların sayı qədərdirsə, belə ardıcillığı *doğru ardıcillıq* adlandıracağıq. Məsələn, “011011” ardıcillığı doğru ardıcillıqdır, “011101” isə doğru ardıcillıq deyil.

Bir və sıfırlardan ibarət tək uzunluqlu hər hansı ardıcillıq verilmişdir. Bu ardıcillıqda bir rəqəmi silməyə icazə verilir. Hansı rəqəmi silmək lazımdır ki, alınan ardıcillıq doğru ardıcillıq olsun? Məsələn, “0111011” ardıcillığında bu məqsədlə soldan ikinci rəqəmi silmək lazımdır.

#### Giriş verilənlərinin formatı

Birinci sətirdə ardıcillıqdakı rəqəmlərin sayını göstərən  $N$  ( $N \leq 200001$ ) tək ədədi, ikinci sətirdə isə sıfır və birlərdən ibarət  $N$  uzunluqlu ardıcillıq verilir.

#### Çıxış verilənlərinin formatı

Ekрана bir ədəd – verilən ardıcillığın doğru ardıcillıq olması üçün silinəcək rəqəmin ardıcillıqdakı nömrəsi çıxarılır. Rəqəmlər verilən ardıcillıqda soldan başlayaraq (1, 2, 3 və s.) nömrələnir. Əgər doğru ardıcillıq almaq mümkün deyilsə, ekrana 0, həll bir neçədirsə, onlardan ixtiyari biri verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
7 0111011	2

### T3-2009-05. Rəqəmlərin sayı

$a^n$  ədədinin yazılışında ən çox təkrar olunan rəqəmin təkrarlanma sayını tapın.

#### Giriş verilənləri

Bir sətirdə iki ədəd –  $a$  ( $0 < a < 10$ ) və  $n$  ( $0 < n < 10001$ ) yerləşir.

#### Çıxış verilənləri

Bir sətirdə  $k$  ( $0 \leq k \leq 9$ ) və  $m$  ədədləri verilir. Belə ki,  $k$  rəqəmi  $a^n$  ədədinin yazılışında maksimum sayda təkrarlanan rəqəm,  $m$  isə onun təkrarlanma sayıdır. Əgər belə rəqəm bir neçədirsə (eyni sayda təkrarlanan varsa), onda çıxışa ən kiçik rəqəm haqqında informasiya verilməlidir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3 17	1 3
7 777	9 90

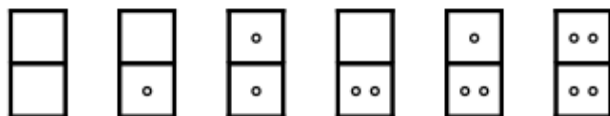
2010-cu ildə informatika üzrə olimpiadanın respublika turunun məsələləri hər sinif üzrə ayrı-ayrılıqda tərtib edilmişdi və mürəkkəblik dərəcəsinə görə hər sinifdə əvvəlki sinifdəkindən bir fərqli məsələ verilmişdi.



### T3-2010-01. Domino

Domino oyununda daşlar iki hissədən ibarət olur və bu hissələrdə qara nöqtələr yerləşə bilər. Hər hissədəki nöqtələrin sayı 0-dan N-dək ola bilər.

Bir hissəsində ən çoxu N nöqtə olan bütün daşlar çoxluğu hər bir hissəsində 0-dan N-dək nöqtə olan və təkrarlanmayan daşlardan ibarətdir. Məsələn, bir hissəsində ən çoxu 2 nöqtə olan bütün daşlar çoxluğu 6 daşdan ibarət olur.



Bir hissəsində ən çoxu N nöqtəsi olan daşların ümumi miqdarını və həmin daşlardakı bütün nöqtələrin cəmini müəyyənləşdirən proqramı tərtib edin.

#### Giriş verilənləri

Bir N ədədi ( $1 \leq N \leq 1000$ ) – bir hissədəki nöqtələrin maksimal sayı daxil edilir.

#### Çıxış verilənləri

Çıxışa bir sətirdə aralarında boşluq simvolu olmaqla iki tam ədəd – tam dəstdəki daşların ümumi sayı və daşların üzərindəki bütün nöqtələrin sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2	6 12
3	10 30
15	136 2040

#### Açıqlama.

Bir hissədəki maksimal miqdarda nöqtələrin sayı 3 olduqda tam dəstdəki daşlar belə olar:

[0|0], [0|1], [0|2], [0|3], [1|1], [1|2], [1|3], [2|2], [2|3], [3|3].

**T3-2010-02. Ədədlərdən böyüyü**

İbtidai sinif şagirdi Məmməd ədədləri müqayisə etməkdə çətinlik çəkir. Onun bu problemi başa düşməsinə kömək etmək üçün müəllim yazı lövhəsində iki üçrəqəmli ədəd yazır və Məmməddən onları müqayisə etməyi xahiş edir. Ancaq o, ədədləri adi qayda ilə soldan sağa deyil, əksinə, sağdan sola müqayisə etməyi və böyük ədədi də sağdan sola yazmağı xahiş edir. Məsələn, 734 və 893 ədədləri üçün Məmməd 437 ədədini yazmalıdır. Çünki 437 ədədinin sağdan sola oxunuşu olan 734 ədədi ikinci ədədin həmin oxunuşundan (398-dən) böyükdür. Məmmədin icra etməli olduğu məsələnin proqramını yazın.

**Giriş verilənləri**

Rəqəmlərindən heç birinin sıfır olmadığı iki **A** və **B** üçrəqəmli tam ədədi daxil edilir.

**Çıxış verilənləri**

Çıxışa məsələdə deyilən qaydada müqayisə etməklə **A** və **B** ədədlərindən böyüyü olan ədəd verilir. Bu ədədin rəqəmləri əks ardıcılıqla yazılmalıdır.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
734 893	437
221 231	132
237 839	938

**T3-2010-03. Təklidlər ölkəsi**

Sindbad özünün növbəti səyahətində Təklidlər ölkəsinə düşür. Bu ölkənin sakinləri cüt ədədlərin nə olduğunu bilmirlər. Belə ki, “2 alma” əvəzinə onlar “3 minus 1 alma” deyirlər. Bu ölkədə artan ədədlər ardıcılığı aşağıdakı kimidir: 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 31, ... Bu ardıcılıqda araşdırma aparmaq üçün Sindbada kömək edin.

Verilmiş **N** nömrəsinə görə ( $1 \leq N \leq 2000000000$ ) bu ardıcılığın **N**-ci həddini tapan proqram yazın.

**Giriş verilənləri**

Bir **N** ədədi daxil edilir.

**Çıxış verilənləri**

Ardıcılığın **N**-ci həddi çıxışa verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5	9
18	55
26	111

**T3-2010-04. Kosmik gəlmələr**

Məmməd məşhur “Kosmik işğalçılar” oyunu üçün öz variantını yazdı. Oyun aşağıdakı kimidir: Yer kürəsinə kosmik işğalçıların gəmiləri hücum edir. Onlar ekranın yuxarı hissəsində sıra-sıra düzülür. Oyunçu ekranın aşağı qırağında hər hansı bir sütunda yerləşən lazerli topu idarə edir. Bir hərəkətlə oyunçu topu sola və ya sağa hərəkət etdirə bilər, yaxud şaquli olaraq yuxarıya atəş açmağa bilər. Əgər oyunçu atəş açırsa, gəlmələrin topun yerləşdiyi sütunda ən yaxındakı gəmisi məhv edilir.



Orijinal oyundan fərqli olaraq Məmmədin variantında gəlmələrin gəmiləri yerində dayanır və atəş açmağa bilmir. Ona görə də oyunçu heç zaman uduza bilməz. Minimum sayda hərəkətlər vasitəsilə gəlmələrin bütün gəmilərini məhv etməkdə Məmmədə kömək edin.

### Giriş verilənlərinin formatı

Birinci sətirdə iki  $n$  və  $p$  ( $1 < n < 100$ ,  $1 < p < n$ ) ədədləri – sütunların sayı və başlanğıcda topun yerləşdiyi sütunun nömrəsi olan ədəd verilir. İkinci sətir  $n$  sayda  $a_i$  ( $1, 2, \dots, a_n$ ) ədədlərindən ibarətdir:  $a_i$  – gəlmələrin  $i$ -ci sütundakı gəmilərinin sayını göstərir ( $1 < i < 100$ ).

### Çıxış verilənlərinin formatı

Çıxışa bir ədəd – gəlmələrin bütün gəmilərini məhv etmək üçün zəruri olan hərəkətlərin minimal sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 4 5 3 4 1 2	20

### T3-2010-05-IX. Ardıcılıqların medianı

Tutaq ki,  $N$  sayda mənfi olmayan tam ədəddən ibarət ardıcılıq verilmişdir.  $N$  tək olduqda artma və ya azalma sırası ilə nizamlanmış bu ardıcılığın uclarından bərabər uzaqlıqda olan elementi ardıcılığın *medianı* adlandıracağıq (aydındır ki, nizamlanmış ardıcılıqda bu elementin nömrəsi, nömrələr 1-dən başlayaraq sayılarsa,  $(N + 1) / 2$  olar).  $N$  cüt olduqda isə bu ardıcılığın medianı olaraq həmin ardıcılıq nizamlanıqda  $N / 2$  və  $(N / 2) + 1$  nömrəli yerlərdə duran iki elementin ədədi ortasını götürəcəyik. İlkin ardıcılıq nizamlanmış olmaya da bilər.

Verilmiş ardıcılığa görə onun medianını tapmaq üçün proqram tərtib edin.

### Giriş verilənlərinin formatı

Birinci sətirdə ardıcılığın uzunluğu olan  $N$  ədədi verilir. İkinci və sonra gələn hər sətirdə bir element olmaqla ardıcılığın öz elementləri yerləşir. Ardıcılığın uzunluğu 1-dən 250 000-dək ola bilən tam ədəddir. Ardıcılığın hər bir elementi özü də daxil olmaqla 0-dan  $(2^{31} - 1)$ -dək tam ədəddir.

### Çıxış verilənlərinin formatı

Çıxışa medianın qiyməti 0.1 dəqiqliklə verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 3 6 5	4.5

### T3-2010-05-X. Yeşiklər

$N$  sayda eyni əşya və  $M$  sayda nömrələnmiş yeşiklər var. Bütün əşyaları yeşiklərə yerləşdirməyin mümkün olan bütün üsullarının sayını müəyyənləşdirən proqram tərtib edin. Üsullar o zaman fərqli sayılır ki, heç olmasa bir yeşikdəki əşyaların sayı fərqli olur.

#### Giriş verilənləri

İki  $N$  və  $M$  ədədləri ( $M \geq 1$ ,  $N \geq 1$ ,  $M + N \leq 60$ ) daxil edilir.

#### Çıxış verilənləri

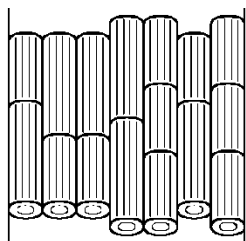
Çıxışa bir ədəd – axtarılan ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2 2	3

### T3-2010-05-XI. Körpüsalma

Yazda rayonlarda şiddətli yağışlar olur. Bu zaman, adətən, sadəcə, tullanmaqla keçilən kiçik Quruçay çayı daşaraq sahillərini aşır. Daşan çayı keçmək üçün qışdan yandırılmamış qalan ağac tirlərdən üzən körpü hazırlamaq planlaşdırılır. Qalan bütün ağac tirlər eyni yoğunluğa malikdir. Bununla yanaşı,  $a$  uzunluğunda  $x$  sayda,  $b$  uzunluğunda isə  $y$  sayda tir var. Qurulacaq körpü  $l$  sıradan, hər sıra isə bir və ya bir neçə tirdən ibarət olmalıdır. Tirləri mişarlamaq olmur, belə ki, axırıncı mişar çay daşqını zamanı axıb getmişdir.

Baş mühəndis körpünü maksimum enlikdə qurmaq istəyir. Həm də bu zaman körpünün eni tirlərin sırasının minimum eni ilə müəyyən olunur. Məsələn, əgər körpünün uzunluğu üçün 7 sıra tir atılmalıdırsa və uzunluğu 3 olan 6 tir və uzunluğu 2 olan 10 tir varsa, onda qurulacaq körpünün maksimal eni 5 olar.



#### Giriş verilənlərinin formatı

Giriş verilənləri 5 natural ədəddən ibarətdir:  $x$ ,  $a$ ,  $y$ ,  $b$  və  $l$ . Bütün ədədlər 150-ni aşmır. Ağac tirlərin ümumi sayı  $l$ -dən kiçik deyil.

### Çıxış verilənlərinin formatı

Çıxışa bir ədəd – körpünün maksimal mümkün eni verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
6 3 10 2 7	5
10 7 20 9 25	9

### T3-2011-01. Hasarın rənglənməsi

V sinif şagirdlərinə N hissədən ibarət olan hasarı rəngləmək tapşırıldı. Sınıfın sərəncamında üç rəng – ağ, mavi və sarı rəngdə boya var. Qonşu hissələri eyni rəngdə olmamaq şərti ilə hasarın verilmiş hissəsini neçə mümkün üsulla rəngləmək olar?

#### Giriş verilənləri

Bir sətirdə bir N ( $1 \leq N \leq 50$ ) ədədi verilir.

#### Çıxış verilənləri

Çıxışa bir ədəd – mümkün rəngləmə variantlarının sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3	12

### T3-2011-02. N-rəqəmli ədəd

Rəqəmlərinin cəmi onların hasilinə bərabər olan N-rəqəmli ədədlərin sayını tapın. Verilmiş N ədədi ( $N < 10$ ) üçün belə ədədlər arasından ən kiçiyini çıxışa verin.

#### Giriş verilənləri

Bir N ədədi daxil edilir.

#### Çıxış verilənləri

Çıxışa iki ədəd – axtarılan ədədlərin sayı və onlar arasında ən kiçik ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1	10 0

### T3-2011-03. Bölənlər

Verilmiş  $N$  natural ədədinə görə  $N!$  ( $N$  faktorial) ədədinin bölənlərinin sayını hesablamaq lazımdır. Məsələn,  $N = 4$  olduqda  $N! = 4 \cdot 3 \cdot 2 \cdot 1 = 24$ . Bu ədədin aşağıdakı bölənləri var: 1, 2, 3, 4, 6, 8, 12, 24. Beləliklə, axtarılan say 8-ə bərabərdir.

#### Giriş verilənləri

Bir  $N$  ( $1 \leq N \leq 45$ ) ədədi daxil edilir.

#### Çıxış verilənləri

Çıxışa bir  $N$  ədədi – tapılmış bölənlərin sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4	8

### T3-2011-04. Kibritlər

Müstəvi üzərində hər tərəfi bir ədəd kibrit çöpü olan  $N$  sayda kvadrat düzəltmək üçün hansı minimal miqdarda kibrit çöpü lazımdır? Kibrit çöplərini qırmaq, bir-birinin üzərinə qoymaq olmaz. Kvadratların təpələri kibrit çöplərinin uclarının birləşdiyi nöqtə, tərəfləri isə kibrit çöplərinin özləri olmalıdır.

Düzəldiləcək  $N$  sayda kvadrat üçün lazım olan kibrit çöplərinin minimal sayını tapan proqram yazın.

#### Giriş verilənləri

Bir  $N$  ( $1 \leq N \leq 10^9$ ) ədədi daxil edilir.

#### Çıxış verilənləri

Çıxışa bir ədəd – verilmiş sayda kvadratların düzəldilməsi üçün lazım olan kibritlərin minimal sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4	12

### T3-2011-05. Parça və düzbucaqlı

$(X_1, Y_1)$ ,  $(X_2, Y_2)$  parçası ilə diaqonallarının uc nöqtələri  $(X_3, Y_3)$ ,  $(X_4, Y_4)$  olan, tərəfləri isə koordinat oxlarına paralel olan düzbucaqlı kəsişirmi? Düzbucaqlı və parçanın, heç olmasa, bir ortaq nöqtəsi varsa, onlar kəsişir.

#### Giriş verilənləri

Birinci sətirdə dörd tam ədəd – parçanın uc nöqtələrinin  $X_1, Y_1, X_2, Y_2$  koordinatları aralarında bir boşluq simvolu olmaqla verilir. İkinci sətirdə dörd tam ədəd – düzbucaqlının diaqonallarından birinin uc nöqtələrinin  $X_3, Y_3, X_4, Y_4$  koordinatları aralarında bir boşluq simvolu olmaqla verilir. Bütün koordinatlar tam ədədlərdir və mütləq qiymətcə 50-ni aşmır.

#### Çıxış verilənləri

Çıxışa bir ədəd – parça və düzbucaqlı kəsişdikdə 1, əks halda isə 0 verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1 3 4 -1 -1 4 3 -2	1

### T3-2012-01. Viruslar bayramı

Bir yay günü planetimizə üzərində yer elminə məlum olmayan viruslar olan meteorit düşür. Meteorit sıx atmosfer qatından keçərkən cəmi  $N$  sayda ( $0 < N < 300$ ) virus sağ qalır. Yer kürəsi virusların çox xoşlarına gəldiyindən onlar burada qalıb yaşamağı qərarlaşıdırırlar.

Hər virus bir gündə  $M$  sayda nəsil verir və onların hər biri sonsuz uzun ömür sürür (məsələn, Yer kürəsinə 3 virus uçub gəlmişdirsə, artıq növbəti gün onların sayı  $3+3M$  olacaq). Virusların sayını göstərən ədədin rəqəmlərinin kvadratları cəmi  $X$ -ə qalıqsız bölünürsə, onda viruslar üçün “X” bayramı olur.

$K$  ( $K < 100$ ) gündə virusların neçə “X” bayramı qeyd edəcəyini müəyyənləşdirən proqram tərtib edin.

#### Giriş verilənləri

Bir sətirdə aralarında boşluq simvolu olan  $N, M, K, X$  tam ədədləri verilir.

#### Çıxış verilənləri

Çıxışa bir ədəd – bayram günlərinin sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2 2 3 4	1



### T3-2012-02. Oxlar

Yalnız “>”, “<” və “-” simvollarından ibarət ardıcillıq verilmişdir. Bu ardıcillıqda gizlənmiş *oxların* sayını tapmaq tələb olunur. Ox dedikdə “>>-->” və “<--<<” şəklində olan altşərtlər nəzərdə tutulur.

#### Giriş verilənləri

Bir sətirdə boşluq işarəsi olmadan “>”, “<” və “-” simvollarından ibarət sətir verilir. Sətirdəki simvolların sayı 250-dən çox deyil.

#### Çıxış verilənləri

Çıxışa axtarılan – oxların miqdarını göstərən ədəd verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
<<<<>>--><--<<--<>>--><<<<<	4

### T3-2012-03. Roma ədədləri

Roma say sistemində verilmiş iki **A** və **B** natural ədədinin cəmini hesablamaq lazımdır. Cavab da Roma say sistemində verilməlidir. Bu say sistemində ədədlər latın əlifbasının baş hərfləri vasitəsilə yazılır:

M = 1000, D = 500, C = 100, L = 50, X = 10, V = 5, I = 1 (bütün ədədlər 2000-i aşmır).

#### Giriş verilənləri

Bir sətirdə Roma say sistemində yazılmış və aralarında “+” işarəsi olan iki ədəd verilir.

#### Çıxış verilənləri

Çıxışa verilmiş ədədlərin cəmi Roma say sistemində verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
VII+II	IX

### T3-2012-04. Fibonaççi sətirləri

*Fibonaççi sətirləri* aşağıdakı kimi təyin olunur:

- Birinci Fibonaççi sətiri “a”-ya bərabərdir.
- İkinci Fibonaççi sətiri “bc”-yə bərabərdir.
- ....
- $(n + 2)$ -ci Fibonaççi sətiri  $(n + 1)$ -ci və  $n$ -ci sətirlərinin birləşməsidir.

Məsələn, ilk beş Fibonaççi sətiri aşağıdakı şəkildə olar:

a  
bc  
abc  
bcabc  
abcabcabc

Sətrin nömrəsini və simvolun bu sətirdəki mövqeyini bilməklə onun hansı simvol olduğunu müəyyənləşdirmək lazımdır.

#### Giriş verilənləri

Bir sətirdə aralarında boşluq simvolu olmaqla 2 tam ədəd –  $K$  və  $P$  ədədləri ( $0 < K \leq 10^8$ ), ( $0 < P \leq 10^8$ ) verilir. Burada  $K$  Fibonaççi sətirinin nömrəsi,  $P$  isə axtarılan simvolun sətirdəki yeri – mövqeyidir.

#### Çıxış verilənləri

Çıxışa axtarılan üç “a”, “b” və “c” simvollarından biri verilir. Əgər  $K$ -cı sətirin ( $K \leq 10^8$ )  $P$ -ci mövqeyində simvol yoxdursa, çıxışa “No solution” məlumatı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
20 46	B

### T3-2012-05. Keçilməz meşə

Müşahidəçi durduğu yerdən meşə istiqamətində yalnız ağacları görürsə, yəni meşənin arxasında heç nə görünmürsə, belə meşə *keçilməz* hesab olunur. Sizə meşənin xəritəsi və müşahidəçinin dayandığı nöqtənin koordinatları verilib. Meşənin həmin müşahidəçiyə görə keçilməz olub-olmadığını müəyyənləşdirmək lazımdır. Qeyd edək ki, əgər meşədə yalnız bir ağac varsa, onda o, keçilməz meşə deyil. Meşənin xəritəsində bütün ağaclar dairə ilə təsvir olunub. Ancaq meşədə bir yerə toplanmış ağaclar da ola bilər (xəritədə belə ağacların təsvirləri kəsişir), eləcə də bir ağac başqasının daxilində ola bilər. Müşahidəçinin dayandığı nöqtə heç bir ağacın daxilində və ya sərhədində deyil.

#### Giriş verilənləri

Birinci sətirdə ağacların sayını göstərən  $N$  ( $1 \leq N \leq 50000$ ) tam ədədi verilir. İkinci sətirdə müşahidəçinin koordinatları olan iki ədəd verilir. Sonrakı  $N$  sətirdə ağacları təsvir edən ədədlər üçlüyü verilir. İlk iki ədəd ağacın təsviri olan dairənin mərkəzinin koordinatları, 3-cüsü isə onun radiusudur. Bütün koordinatlar dəqiq verilir, mütləq qiymətcə 100000-i aşmır və onluq vergüldən sonra iki rəqəm olmaqla həqiqi ədədlərdir.

#### Çıxış verilənləri

Çıxışa meşə keçilməzdirsə YES, əks halda isə NO məlumatı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
4 0 0 2 2 2 -2 2 2 -2 -2 2 2 -2 2	YES
2 10 10 0 0 1 0.5 0 2	NO

### **T3-2013-01. Təpələr və düzənliklər**

Hündürlüklərin ölçülməsi nəticəsində alınan tam ədədlər sırası verilmişdir. Bu sırada ardıcıl yerləşən eyni ədədlərdən ibarət ardıcılıqların içərisindən ən uzun ardıcılığı tapan proqram tərtib edin. Əgər eyni uzunluqlu bir neçə belə ardıcılıq olarsa, onda ən birinci ardıcılığı seçmək lazımdır.

#### **Giriş verilənlərinin formatı**

Birinci sətirdə ölçmələrin sayını göstərən  $N$  ədədi ( $1 \leq N \leq 100$ ) yerləşir. İkinci sətirdə aralarında boşluq simvolu olmaqla hündürlüklərin ölçülməsinin nəticələri olan və  $(-10000, 10000)$  aralığında yerləşən  $N$  sayda tam ədəd verilir.

#### **Çıxış verilənlərinin formatı**

Çıxış verilənləri iki ədəddir: tapılan ardıcılığın uzunluğu və eyni ədədlərdən ibarət həmin ardıcılığın birinci elementinin nömrəsi.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
10 7 5 5 5 -4 -4 -4 2 5 5	3 2

### **T3-2013-02. Palindrom (bax: T3-2009-03)**

Hər iki istiqamətdə eyni cür oxunan söz *palindromdur*.

İstənilən sözdə minimum sayda hərfi silməklə həmin sözü palindroma çevirən proqram yazın. Söz dedikdə latın əlifbasının adi (kiçik) hərflərindən ibarət ardıcılıq nəzərdə tutulur.

#### **Giriş verilənlərinin formatı**

Aralarında boşluq olmamaqla latın əlifbasının kiçik hərflərindən ibarət söz (255 simvoldan çox olmamaqla) daxil edilir.

#### **Çıxış verilənlərinin formatı**

Çıxışa bir ədəd – sözün palindroma çevrilməsi üçün silinməsi lazım olan simvolların minimum sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
qwerrewtq	1

**T3-2013-03. Axırncı ədəd**

1-dən N-dək nömrələnmiş N sayda adam çevrə boyunca düzülmüşdür. Birincidən başlayaraq onları saymağa başlayırlar, hər üçüncü adam çevrədən çıxır. Sayma axırncı adam qalanadək davam etdirilir. Məsələn, əgər  $N = 8$  isə, onda çevrədən ardıcıl olaraq 3, 6, 1, 5, 2, 8, 4 nömrəli adamlar çıxacaq, axıra 7 nömrəli adam qalacaq.

Axırncı qalan adamın nömrəsini müəyyənləşdirən proqram yazmaq tələb olunur.

**Giriş verilənlərinin formatı**

Yeganə bir N ( $1 \leq N \leq 108$ ) ədədi daxil edilir.

**Çıxış verilənlərinin formatı**

Çıxışa bir ədəd – axırncı qalan adamın nömrəsi verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
8	7

**T3-2013-04. Yaxşı elementlər**

N sayda tam ədəddən ibarət A ardıcılığı verilmişdir. Əgər bu ardıcılıqdakı *i*-ci element özündən əvvəl gələn elementlərdən düzəldilmiş üç ədədin cəminə bərabər olarsa (cəmdəki ədədlərin hər üçü eyni elementin qiymətinə də bərabər ola bilər, deməli, ardıcılığın bir elementi cəmə bir dəfədən çox daxil ola bilər), onda onu *yaxşı element* adlandıracağıq. Verilmiş ardıcılıqda neçə yaxşı element olduğunu müəyyənləşdirin.

**Giriş verilənlərinin formatı**

Birinci sətirdə A ardıcılığının elementlərinin sayı olan bir N ədədi ( $1 \leq N \leq 5000$ ) verilir. İkinci sətirdə aralarında boşluq simvolu olmaqla A ( $-100\,000 \leq A_i \leq 100\,000$ ) ardıcılığını təşkil edən N sayda tam ədəd yerləşir.

**Çıxış verilənlərinin formatı**

Çıxışa bir ədəd – yaxşı elementlərin sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
2 1 3	1
6 1 2 3 5 7 10	4
3 -1 2 0	1

### **T3-2013-05. Üçqatçı icraçı**

Üçqatçı icraçının komandalar sistemi iki komandadan ibarətdir:

- 1 çıxmaq
- 3-ə vurmaq

Birinci komanda ədədi 1 vahid azaldır, ikincisi isə ədədi 3-ə vurur. Üçqatçının proqramı bu komandalar ardıcılığıdır. Əgər Üçqatçıya ardıcıl olaraq üçdən artıq “1 çıxmaq” komandası verilərsə, onda o, möhkəm qızır və korlanır. N ədədini M-ə çevirən neçə belə proqram var?

#### **Giriş verilənlərinin formatı**

Bir sətirdə aralarında boşluq olmaqla N və M ədədləri daxil edilir.

#### **Çıxış verilənlərinin formatı**

Çıxışa bir ədəd – N ədədini M-ə çevirən belə proqramların sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
5 12	4

### **T3-2014-01. Sətrin açılışı (bax: T3-2005-02)**

Latın əlifbasının baş hərflərindən ibarət sətri gözdən keçirək. Məsələn: AAAABCCCCDDDD. Bu sətrin uzunluğu 14-ə bərabərdir. Sətirdəki ardıcıl təkrarlanan simvollar həmin simvolların təkrarlanma sayını göstərən ədəd və bu simvolun özü ilə əvəz edilir. Yuxarıda verilmiş sətir 4AB5C4D kimi göstərilə bilər. Təsvir edilən metodu *sətirlərin qablaşdırılması* adlandıracağıq.

Qablaşdırılmış sətir əsasında ilkin sətri bərpa edən proqramı tərtib edin.

#### **Giriş verilənləri**

Giriş verilənləri bir qablaşdırılmış sətirdən ibarətdir. Sətrin uzunluğu 80-i aşmır.

#### **Çıxış verilənləri**

Çıxışa bərpa edilmiş sətir verilir. Bu zaman sətir hər birinin uzunluğu 40 simvola bərabər olan alt sətirlərə bölünür (sonuncu sətir istisna olmaqla).

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3A4B7D	AAABBBBDDDDDDDD
95AB	AA AA AAAAAAAAAAAAAAAAAAAAAB

### T3-2014-02. Vahidlərin ƏBOB-u

Onluq say sistemində yalnız vahidlərdən ibarət olan iki natural ədəd verilmişdir. Birinci ədəddə düz  $N$  sayda 1 (bir), ikinci ədəddə isə  $M$  sayda 1 (bir) var. Bu ədədlərin ƏBOB-nu (ən böyük ortaq bölənini) tapmaq tələb olunur. (İki  $a$  və  $b$  ədədinin ƏBOB-u elə ən böyük  $c$  ədədinə deyilir ki, həm  $a$  ədədi, həm də  $b$  ədədi həmin  $c$ -yə qalıqsız bölünür.)

#### Giriş verilənləri

İki  $N$  və  $M$  tam ədədi ( $1 \leq N, M \leq 2000$ ).

#### Çıxış verilənləri

$N$  və  $M$  ədədlərinin ƏBOB-u.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
1 1	1
2 2	11

### T3-2014-03. Cəm

$x$  natural ədədi verilib. Həmin ədədin dörd natural ədədin cəmi şəklində təqdimədmə üsullarının sayını tapın:

$x = a + b + c + d$ , burada  $a \leq b \leq c \leq d$ .

#### Giriş verilənləri

Bir tam  $x$  ədədi ( $1 \leq x \leq 1500$ ) daxil edilir.

#### Çıxış verilənləri

Çıxışa bir ədəd – təqdimədmə üsullarının sayı verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
3	0
5	1

### T3-2014-04. Ariadnanın ipi

Minotavrın olduğu labirintdən çıxmaq üçün Ariadna Teseyə bir yumaq ip verir. Teseyin labirintdəki marşrutunu daxil etməklə dalanlara girmədən və ilgək yaratmadan onların labirintdən geriye çıxma biləcəkləri ən qısa yolu tapan program yazın.

#### Giriş verilənləri

N, S, W, E hərflərindən ibarət və uzunluğu 250-dən çox olmayan sətirlə Teseyin marşrutu verilir.

Hərflər bunları göstərir: N – bir “addım” şimala, S – bir “addım” cənuba, W – bir “addım” qərbə, E isə bir “addım” şərqə.

#### Çıxış verilənləri

Giriş verilənlərinə analoji olan geriye tapılmış yol çıxışa verilir. Əgər tapılan marşrut bir neçədirsə, onda seçimi aşağıdakı ardıcılığa uyğun etmək lazımdır: N, E, S, W.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
EENNESWSSWE	NWW

### T3-2014-05. Hərflər üzrə tullanma

Latın əlifbasının baş hərflərindən təşkil olunmuş N sayda simvollar zənciri verilmişdir. Ən çoxu K simvol tullanmaqla zəncirin birinci simvolundan axırını simvolunadək getmək lazımdır. Tullanma nəticəsində simvol dəyişmərsə, yəni əvvəlki ilə eyni olan hərfin üzərinə tullanılırsa, belə tullanmanın qiyməti 0-a, başqa simvolun üzərinə tullanılırsa, qiyməti 1-ə bərabər olur.

Birinci simvoldan axırını simvola çatmaq üçün qiyməti ən az olan keçidi hesablayan program yazın.

#### Giriş verilənləri

Birinci sətirdə iki tam ədəd verilir: simvollar zəncirinin uzunluğu – N ( $2 \leq N \leq 10^5$ ) və maksimal tullanma uzunluğu – K ( $1 \leq K < N$ ). İkinci sətirdə N sayda hərfdən ibarət zəncir yerləşir.

#### Çıxış verilənləri

Çıxışa bir ədəd – yerdəyişmənin minimum qiyməti verilir.

Giriş verilənlərinə nümunə	Çıxış verilənlərinə nümunə
10 2 ABABBCACBC	2



## 6.2. HƏLL NÜMUNƏLƏRİ

Bu paraqrafda müxtəlif illərdə olimpiadanın respublika turuna düşmüş bəzi məsələlərin həlli verilir. Şərtləşməyə görə, proqramın adı həmin proqramın hansı ilin hansı məsələsinin həlli olduğunu göstərir. Məsələn, T3\_2005\_01 adı proqramın 2005-ci ildə respublika turunda (T3) verilmiş 1-ci (01) məsələnin həlli olduğu deməkdir. Şagirdlərin müxtəlif proqramlaşdırma dillərindən istifadə etdiyini nəzərə alaraq proqram mətnləri **Pascal** və ya **C++** dilində təqdim olunur. Bəzi proqram kodları isə hər iki dildə verilir.

T3-2005-01.

Zəng

**Program T3\_2005\_01;**

```
type
  tt = array[1..3] of string;
  mas = array[1..3] of integer;
var
  t: tt;
  i, p, d, k, k2, k1: integer;
  s1, s2, s: string;
  x, y, su: mas;

procedure init(var x: mas; var l: string);
var
  nl, k, i: integer;
  t: tt;

begin
  for i := 1 to 3 do
    t[i] := '';

  nl := length(l);
  k := 3;
  for i := nl downto 1 do
    if l[i] = ':' then dec(k)
      else t[k] := l[i] + t[k];
  for i := 1 to 3 do
    if t[i] = '' then x[i] := 0
      else val(t[i], x[i], k);
end;

begin
  assign(input, 'a.in');
  reset(input);
```

```

readln(s1);
readln(s2);
close(input);

init(x, s1);
init(y, s2);
p := 0;

for i := 3 downto 2 do begin
    su[i] := x[i] + y[i] + p;
    p := su[i] div 60;
    su[i] := su[i] mod 60;
end;

su[1] := x[1] + y[1] + p;
d := su[1] div 24;
su[1] := su[1] mod 24;

for i := 1 to 3 do begin
    str(su[i], t[i]);
    if length(t[i]) = 1 then
        t[i] := '0' + t[i];
end;

assign(output, 'a.out');
rewrite(output);
write(t[1] + ':' + t[2] + ':' + t[3]);

if d > 0 then
    write('+', d, ' days');
writeln;
close(output);
end.

```

T3-2005-02.

Ardıcılıq

#### **Program T3\_2005\_02;**

```

var
    A: array[0..9] of byte;
    n, i, l, k, j, m, d: integer;

begin
    assign(input, 'd.in');
    reset(input);
    readln(n);
    close(input);

    j := 2;
    i := 1;

```

```

while (j < n) do begin
    inc(i);
    for m := 1 to i do begin
        l := m;
        repeat
            inc(j);
            l := l div 10;
        until l = 0;
        if j > n then break;
    end;
end;
for i := 1 to j - n - 1 do
    m := m div 10;

d := m mod 10;
assign(output, 'd.out');
rewrite(output);
writeln(d);
close(output);
end.

```

T3-2007-01.

Programçılar

#### **Program T3\_2007\_01;**

```

var
    a: array[1..100] of set of 1..50;
    b: array[1..100] of integer;
    bb: boolean;
    i, n, j, k, e, nn: integer;

begin
    for i := 1 to n do
        a[i] := [];

    assign(input, 'a.in');
    reset(input);
    readln(n, e);

    k := 0;
    for i := 1 to e do begin
        read(nn);
        fillchar(b, sizeof(b), 0);
        bb := false;
        for j := 1 to nn do begin
            read(b[j]);
            if b[j] = 1 then bb := true;
        end;
        if bb then begin
            inc(k);
        end;
    end;
end.

```

```

        for j := 1 to nn do
            a[b[j]] := a[b[j]] + [k];
        end
    else begin
        for j := 2 to nn do
            a[b[1]] := a[b[1]] + a[b[j]];
        for j := 2 to nn do
            a[b[j]] := a[b[1]];
        end;
    end;
end;

close(input);
assign(output, 'a.out');
rewrite(output);

if (a[1] <> []) then
    write(1)
else
    write(0);

for i := 2 to n do
    if (a[1] <> []) and (a[i] = a[1]) then
        write(' ', i);

writeln;
close(output);
end.

```

T3-2007-04.

Yol

#### **Program T3\_2007\_04;**

```

var
    A: array[1..100, 1..100] of longint;
    i, j, n, m: integer;

begin
    assign(input, 'd.in');
    reset(input);
    readln(n, m);
    close(input);

    for j := 2 to m do
        A[1, j] := 1;
    for i := 2 to n do
        A[i, 1] := 1;
    for i := 2 to n do
        for j := 2 to m do
            A[i, j] := A[i-1, j] + A[i, j-1];
        end;
    end;
end.

```

```

    assign(output, 'd.out');
    rewrite(output);
    writeln(A[n, m]);
    close(output);
end.

```

**T3-2007-05.**

**Sola dönmələr**

**Program T3\_2007\_05;**

```

var
    n, i, k: integer;
    x1, y1, x2, y2, x3, y3, d, a1, a2, b1, b2: real;

begin
    k := 0;
    assign(input, 'e.in');
    reset(input);
    readln(n);

    if n > 2 then begin
        readln(x1, y1);
        readln(x2, y2);
        for i := 3 to n do begin
            readln(x3, y3);
            a1 := x2 - x1;
            a2 := y2 - y1;
            b1 := x3 - x2;
            b2 := y3 - y2;
            d := a1 * b2 - a2 * b1;

            if d > 0 then inc(k);

            x1 := x2;
            y1 := y2;
            x2 := x3;
            y2 := y3;
        end;
    end;
    close(input);
    assign(output, 'e.out');
    rewrite(output);
    writeln(k);
    close(output);
end.

```

**Program T3\_2008\_01;**

```
var
    k, q: longint;
    s: string;

begin
    assign(input, 'a09.in');
    reset(input);
    readln(k);
    close(input);

    dec(k);
    s := '';
    repeat
        q := k mod 2;
        if q = 1 then
            s := '5' + s
        else
            s := '0' + s;

        k := k div 2;
    until k = 0;

    assign(output, 'a09.out');
    rewrite(output);
    writeln(s);
    close(output);
end.
```

**Program T3\_2008\_02;**

```
var
    n, x, y, k, m, t, t1: longint;

begin
    assign(input, 'xerox.in');
    reset(input);
    readln(n, x, y);
    close(input);

    if x < y then
        t1 := x
    else
        t1 := y;
```

```

m := n - 1;
k := round(m * y / (x + y));

if k * x > (m - k) * y then
    t := k * x + t1
else
    t := (m - k) * y + t1;

assign(output, 'xerox.out');
rewrite(output);
writeln(t);
close(output);
end.

```

**T3-2008-03.**

**Yanacaqdoldurma məntəqələri**

**Program T3\_2008\_03;**

```

var
    a: array[1..100000, 1..2] of longint;
    i, j, n, d, min, imin: longint;

procedure sort(l, r: longint);
var
    c, x, y, i, j: longint;
begin
    i := l;
    j := r;
    x := a[(l+r)div 2, 1];

    repeat
        while a[i, 1] < x do inc(i);
        while x < a[j, 1] do dec(j);

        if i <= j then begin
            if a[i, 1] > a[j, 1] then begin
                y := a[i, 1];
                a[i, 1] := a[j, 1];
                a[j, 1] := y;
                c := a[i, 2];
                a[i, 2] := a[j, 2];
                a[j, 2] := c;
            end;
            inc(i);
            dec(j);
        end;
    until i > j;
    if l < j then sort(l, j);
    if i < r then sort(i, r);
end;

```

```

end;

begin
  assign(input, 'avto.in');
  reset(input);
  readln(n);

  for i := 1 to n do begin
    read(a[i, 1]);
    a[i, 2] := i;
  end;

  sort(1, n);
  min := 999999999;
  for i := n downto 2 do begin
    d := a[i, 1] - a[i-1, 1];
    if d <= min then begin
      min := d;
      imin := a[i, 2];
    end;
  end;
  assign(output, 'avto.out');
  rewrite(output);
  writeln(imin);
  close(output);
end.

```

T3-2008-04.

Artıq yük

#### **Program T3\_2008\_04;**

```

var
  a, b: array[1..50] of int64;
  s, s1, m, min, k: int64;
  i, j, n, l: longint;

begin
  assign(input, 'baggage.in');
  reset(input);
  readln(n, m);

  for i := 1 to n do
    read(a[i]);
  readln;
  for i := 1 to n do
    read(b[i]);
  close(input);

  min := 99999999999;
  k := round(exp(ln(2)*n)) - 1;

```



```

for i := 1 to k do begin
    s := 0;
    s1 := 0;
    l := i;
    for j := 1 to n do begin
        if odd(l) then begin
            s := s + a[j];
            s1 := s1 + b[j];
        end;

        if s1 >= min then break;
        l := l div 2;
        if l = 0 then break;
    end;
    if (s >= m) and (s1 < min) then min := s1;
end;
assign(output, 'baggage.out');
rewrite(output);
writeln(min);
close(output);
end.

```

T3-2009-01.

Køsr

```

#include <cstdlib>
#include <iostream>

#define nn 100000

void mult(char x[], int *nx, int a, int n);
using namespace std;

int main(int argc, char *argv[])
{
    char x[nn];
    long y[10] = {0};
    int a, n, nx;

    cin >> a >> n;

    for (int i=0; i<nn; i++)
        x[i] = 0;

    x[1] = a;
    nx = 1;
    mult(x, &nx, a, n);

    for (int i=nx; i>0; i--)
        cout << (int)x[i];
    cout << endl << nx << endl;
}

```

```

    for(int i=nx; i>0; i--)
        y[x[i]]++;

    int imax = 0;
    for(int i=1; i<10; i++)
    {
        if (y[i] > y[imax])
            imax = i;
    }

    cout << imax << " " << y[imax] << endl;
    system("PAUSE");
    return EXIT_SUCCESS;
}

void mult(char x[], int *nx, int a, int n)
{
    int m=*nx, q, p;
    for (int k=1; k<n; k++) {
        p = 0;
        for (int i=1; i<=m; i++) {
            q = x[i] * a + p;
            x[i] = q % 10;
            p = q / 10;
        }

        while (p != 0){
            m++;
            x[m] = p % 10;
            p = p / 10;
        }
    }
    *nx = m;
}

```

T3-2009-02.

Şahmat

```

#include <cstdlib>
#include <iostream>

using namespace std;

int main()
{
    int n, a;
    cin >> n;

    a = (n & 1) ? n / 2 : n / 4 * 2 - 1;
    cout << a << " " << n-a << endl;
}

```

```
//    system("PAUSE");
return EXIT_SUCCESS;
}
```

T3-2009-03.

Palindrom

```
#include <stdio>
#include <stdlib>

bool is(char *a, char *b)
{
    int ax = a[1] - 'a', ay = a[2] - '1', bx = b[1] - 'a', by =
b[2] - '1';
    int dx = bx - ax, dy = by - ay;

    return
        (a[0] == 'B' && abs(dx) == abs(dy)) ||
        (a[0] == 'N' && dx * dx + dy * dy == 5) ||
        (a[0] == 'R' && dx * dy == 0) ||
        (a[0] == 'Q' && (abs(dx) - abs(dy)) * dx * dy == 0) ||
        (a[0] == 'K' && abs(dx) <= 1 && abs(dy) <= 1);
}

int main( void )
{
    char sa[9], sb[9];
    scanf("%s%s", sa, sb);
    bool fa, fb;

    fa = is(sa, sb);
    fb = is(sb, sa);
    if (fa && fb)
        puts("BOTH");
    else if (fa)
        puts("WHITE");
    else if (fb)
        puts("BLACK");
    else
        puts("NONE");

    return 0;
}
```

**Program T3\_2009\_04;**

```
const
    maxn = 200001;
var
    v, a, b: array [0..maxn + 1] of longint;
    i, n, ans: longint;
    s: string;
    ch: char;

begin
    readln(n);

    for i := 1 to n do begin
        read(ch);
        v[i] := ord(ch) - ord('0');
    end;

    a[0] := 0;
    for i := 1 to n do begin
        if i mod 2 = 1 then
            a[i] := a[i - 1] + v[i]
        else
            a[i] := a[i - 1] - v[i];
    end;

    b[n + 1] := 0;
    for i := n downto 1 do begin
        if i mod 2 = 1 then
            b[i] := b[i + 1] + v[i]
        else
            b[i] := b[i + 1] - v[i];
    end;

    ans := -1;
    for i := 0 to n - 1 do begin
        if a[i] = b[i + 2] then begin
            ans := i + 1;
            break;
        end;
    end;
    writeln(ans);
end.
```

```
function isPaly(s : string) : boolean;

var
  i : integer;
  n : integer;

begin
  n := length(s);
  for i := 1 to n do begin
    if (s[n - i + 1] <> s[i]) then begin
      result := false;
      exit;
    end;
  end;
  result := true;
end;

var
  s : string;
  n : integer;
  i : integer;
  t : string;
  j : integer;

begin
  readln(s);
  n := length(s);
  s := s + s;
  for i := 1 to n do begin
    t := '';
    for j := 0 to n - 1 do begin
      t := t + s[i + j];
    end;
    if (isPaly(t)) then begin
      writeln('YES');
      halt(0);
    end;
  end;
  writeln('NO');
end.
```

**C++**

```
#include <iostream>
#include <math.h>

using namespace std;

int main()
{   int n;
    cin >> n;
    cout << 3*(long long)pow(2, n-1) << endl;
    return 0;
}
```

**Pascal**

**Program T3\_2011\_01;**

```
var
    n,i: integer;
    p: int64;

begin
    p := 1;
    readln(n);
    for i := 1 to n - 1 do
        p := p * 2;
    writeln(3*p);
end.
```

## C++

```
#include <iostream>

using namespace std;

int main()
{   int a[] = {0,10,1,6,12,40,30,84,224,144};
    long long b[] =
{0,0,22,123,1124,11125,111126,1111127,11111128,111111129};
    int n;

    cin >> n;
    cout << a[n] << " " << b[n] << endl;

    return 0;
}
```

## Pascal

**Program** T3\_2011\_02;

```
const
    a: array[0..9] of integer = (0, 10, 1, 6, 12, 40, 30, 84,
224, 144);
    b: array[0..9] of longint = (0, 0, 22, 123, 1124, 11125,
11126, 1111127, 11111128, 111111129);

var
    n: integer;

begin
    readln(n);
    writeln(a[n], ' ', b[n]);
end.
```

C++

```
#include <iostream>

using namespace std;

long long A[50];

void prime(int a) {
    int l = 1;
    while (a != 1) {
        l++;
        while (a % l == 0) {
            a /= l;
            A[l]++;
        }
    }
}

int main(int argc, char *argv[])
{
    long long S = 1;
    int N;

    cin >> N;

    for (int i = 2; i <= N; i++)
        prime(i);

    for (int i = 2; i <= N; i++)
        S *= (A[i] + 1);

    cout << S << endl;

    return 0;
}
```

C++

```
#include <iostream>
#include <math.h>

using namespace std;
```



```

int main()
{
    long n, k, m, l;

    cin >> n;

    k = (long)sqrt(n);
    m = n - k * k;
    l = 2 * (k + 1) * k;
    if (m > 0)
        l += 2 * m + 1;
    if (m > k)
        l++;

    cout << l << endl;

    return 0;
}

```

T3-2011-05.

Parça və düzbucaqlı

C++

```

#include <iostream>
int zpq(int , int , int , int );

using namespace std;

int main()
{
    int x1, y1, x2, y2, x3, y3, x4, y4;
    int p, ans = 0, z;
    int a, b, c, d, a1, b1, c1;
    int u1 = 0, u2 = 0;

    cin >> x1 >> y1 >> x2 >> y2;
    cin >> x3 >> y3 >> x4 >> y4;

    p = zpq( x1,y1,x2,y2);

    if (x3 < x4) {
        a = x3;
        b = x4;
    }
    else {
        a = x4;
        b = x3;
    }
}

```

```

if (y3 < y4){
    c = y3;
    d = y4;
}
else {
    c = y4;
    d = y3;
}

if ((x1 >= a && x1 <= b && y1 >= c && y1 <= d) ||
    (x2 >= a && x2 <= b && y2 >= c && y2 <= d))
{
    cout << 1 << endl;
    return 0;
}
switch(p) {
    case 0:
        if (y1 > y2) {z = y1; y1 = y2; y2 = z;}
        if (y1 > d || y2 < c || x1 < a || x1 > b )
            ans = 0;
        else
            ans = 1;
        break;

    case 1:
        if (x1 > x2) {z = x1; x1 = x2; x2 = z;}
        if (y1 > d || y1 < c || x1 > b || x2 < a)
            ans = 0;
        else
            ans = 1;
        break;

    case 2:
        if (x1 >= a && x1 <= b && y1 >= c && y1 <= d)
            ans = 1;
        else
            ans = 0;
        break;

    case 3:
        a1 = y1 - y2;
        b1 = x2 - x1;
        c1 = x1 * (y2 - y1) - y1 * (x2 - x1);
        ans = 1;
        double yy = (-c1 - a1 * a) / ((double)b1);
        if (y1 < y2){
            u1 = y1;
            u2 = y2;
        }
        else {
            u1 = y2;
            u2 = y1;
        }
    }
}

```

```

    }
    if (yy >= c && yy <= d && yy >= u1 && yy <= u2)
        break;
    yy = (-c1 - a1 * b) / ((double)b1);
    if (yy >= c && yy <= d && yy >= u1 && yy <= u2)
        break;
    double xx = (-c1 - b1 * c) / ((double)a1);
    if (x1 < x2){
        u1 = x1;
        u2 = x2;
    }
    else {
        u1 = x2;
        u2 = x1;
    }
    if (xx >= a && xx <= b && xx >= u1 && xx <= u2)
        break;
    xx = (-c1 - b1 * d) / ((double)a1);
    if (xx >= a && xx <= b && xx >= u1 && xx <= u2)
        break;
    ans = 0;
    break;
}
cout << ans << endl;
return 0;
}

int zpq(int u1, int v1, int u2, int v2) {
    if (u1 == u2 && v1 != v2) return 0;
    if (u1 != u2 && v1 == v2) return 1;
    if (u1 == u2 && v1 == v2) return 2;

    return 3;
}

```

T3-2012-01.

Viruslar bayramı

```

#include <iostream>
#include <math.h>

double ugol(double x1, double y1, double x2, double y2);
void sortt(int n);
double *ugli;
char *rl;
using namespace std;

int main()
{
    int n, j = 0;
    double ug, x, y, xi, yi, ri;
}

```

```

cin >> n;

ugli = new double [2*n];
rl = new char [2*n];

cin >> x >> y;

for(int i=0; i<n; i++)
{
    cin >> xi >> yi >> ri;
    ug = ugol(x,y,xi,yi); //cout<<ug<<endl;
    double d = sqrt(pow(x-xi, 2) + pow(y-yi, 2));
    double hug = asin(ri/d);
    ugli[j] = ug - hug;
    ugli[j+1] = ug + hug;
    rl[j] = '0';
    rl[j+1] = '1';
    j += 2;
}
int m = j;
sortt(m);
int k = 0;

for (j=0; j<m-1; j++)
    cout << rl[j] << " " << ugli[j] << ",";

for (j=0; j<m-1; j++)
{
    if (rl[j] == '0')
        k++;
    else
        k--;
    if (k == 0)
    {
        if (ugli[j+1]-ugli[j] > 0.005)
        {
            cout << "NO\n";
            return 0;
        }
    }
}

cout << "YES\n";
return 0;
}

double ugol(double x1, double y1, double x2, double y2)
{
    double u = x2 - x1, v = y2 - y1;
    double cosa = u / sqrt(u*u+v*v);
    double ug = acos(cosa);

```

```

    if (y2 < y1)
        ug = 2 * M_PI - ug;

    return ug;
}

void sortt(int n)
{
    int i, j;
    double c;
    char s;
    for (i = 1; i < n; i++)
    {
        if (ugli[i-1] > ugli[i]) {
            c = ugli[i];
            ugli[i] = ugli[i-1];
            ugli[i-1] = c;
            s = rl[i];
            rl[i] = rl[i-1];
            rl[i-1] = s;
        }
        for (j = i-1; j > 0 && ugli[j] <= ugli[j-1]; j--) {
            c = ugli[j];
            ugli[j] = ugli[j-1];
            ugli[j-1] = c;
            s = rl[j];
            rl[j] = rl[j-1];
            rl[j-1] = s;
        }
    }
}

```

T3-2012-02.

Oxlar

```

#include <iostream>
#define n 1000000000

using namespace std;

int main()
{
    char ss[3][3]={{'a','0','0'},{'b','c','0'},{'a','b','c'}};
    char *s;
    int i, j, k, p;

    s = new char [n];

    cin >> k >> p;

    if (k > 40) {
        if (k%2 == 1)
            k = 39;
    }
}

```

```

        else
            k = 40;
    }
    if (k < 3) {
        if (p > 3) {
            cout << "No solution" << endl;
            return 0;
        }

        char c = ss[k-1][p-1];
        if (c == '0') {
            cout << "No solution" << endl;
            return 0;
        }
        cout << c << endl;
        return 0;
    }

    s[n-1] = 'c';
    s[n-2] = 'b';
    s[n-3] = 'a';
    int i1 = n - 3, i2 = n - 2;

    for (i=4; i<=k; i++) {
        for (j=i1-1; j>=i1+i2-n; j--)
            s[j] = s[n+j-i1];

        i2 = i1;
        i1 = j+1;
    }
    if (i1+p-1 >= n){

        cout << "No solution\n";
        return 0;
    }
    cout << s[i1+p-1] << endl;
    return 0;
}

```

T3-2012-03.

Roma ədədləri

```

#include <iostream>
#include <stdlib.h>
#include <string>

```

```

using namespace std;

```

```

int numb(string s, int k1, int k2);
int kod(char c);
char kod1(int c);

```

```

string sum(int n);
string charnum(int m, int k);

int main()
{
    string s;
    int z;

    cin >> s;

    int i, l = s.length();

    for (i = 0; i < l && s[i] != '+'; i++);

    z = numb(s, 0, i) + numb(s, i+1, l);

    cout << sum(z) << endl;

    return 0;
}

int kod(char c)
{
    switch (c) {
        case 'M' : return 1000;
        case 'D' : return 500;
        case 'C' : return 100;
        case 'L' : return 50;
        case 'X' : return 10;
        case 'V' : return 5;
        case 'I' : return 1;
    }

    return 0;
}

int numb(string s, int k1, int k2)
{
    int i = k1, n = 0, a, b;
    a = kod(s[k1]);

    for (; i < k2; i++) {
        b = kod(s[i+1]);

        if (a < b)
            n += -a;
        else
            n += +a;
        a = b;
    }
    n += b;

    return n;
}

```

```

string sum(int n)
{
    int k = 0, p, d = 1000;
    string s = "";

    for (int i=0; i<4; i++) {
        p = n / d;
        if (p != 0)
            s += charnum(p, k);
        k += 2;
        n %= d;
        d /= 10;
    }
    return s;
}

string charnum(int m, int k)
{
    char ss[] = {'M','D','C','L','X','V','I'};
    string s;
    int i;

    if (m < 4) {
        for (i = 0; i < m; i++)
            s += ss[k];
        return s;
    }
    if (m == 4) {
        s += ss[k];
        s += ss[k-1];
        return s;
    }
    if (m == 5) {
        s += ss[k-1];
        return s;
    }
    if (m < 9) {
        s += ss[k-1];
        for (i=0; i<m-5; i++)
            s += ss[k];
        return s;
    }
    if (m == 9) {
        s += ss[k];
        s += ss[k-2];
        return s;
    }

    return s;
}

```



```

char kod1(int c)
{
    switch (c)
    {
        case 1000 : return 'M';
        case 500  : return 'D';
        case 100  : return 'C';
        case 50   : return 'L';
        case 10   : return 'X';
        case 5    : return 'V';
        case 1    : return 'I';
    }

    return '0';
}

```

T3-2012-04.

Fibonaççi satirleri

```

#include <iostream>
#include <string.h>

using namespace std;

int strr(string s, int i);
int strl(string s, int i);

int main()
{
    string s;
    int k = 0;

    cin >> s;
    int n=s.length();
    for (int i=0; i<n-4; i++) {
        if (s[i] == '>')
            k += strr(s, i);
        else
            if (s[i] == '<') k += strl(s, i);
    }
    cout << k << endl;

    return 0;
}

int strr(string s, int i)
{
    if (s[i+1] == '>' && s[i+2] == '-' && s[i+3] == '-' &&
s[i+4] == '>')
        return 1;
    else
        return 0;
}

```

```

int strl(string s, int i)
{
    if (s[i+1] == '-' && s[i+2] == '-' && s[i+3] == '<' &&
s[i+4] == '<')
        return 1;
    else
        return 0;
}

```

T3-2012-05.

Keçilməz meşə

```

#include <iostream>
using namespace std;
int main()
{
    long long n, m, s, m1;
    int x, k, d = 0, u;
    cin >> n >> m >> k >> x;
    s = n;
    for (int i=1; i<=k; i++) {
        m1 = s;
        u = 0;
        while (m1 != 0) {
            int q = m1 % 10;
            u += q * q;
            m1 /= 10;
        }
        if (u%x == 0)
            d++;
        s *= 1 + m;
    }
    cout << d << endl;
    return 0;
}

```

T3-2014-01.

Sətrin açılışı

Pascal

**Program T3\_2014\_01;**

```

var
    s: string[80];
    num: string[10];
    i, j, n, l: integer;

```

```

begin
  readln(s);
  l := length(s);
  i := 1;
  while (i <= l) do begin
    if (s[i] >= 'A') and (s[i] <= 'Z') then write(s[i])
    else begin
      num := '';
      while (s[i] >= '0') and (s[i] <= '9') do begin
        num := num + s[i];
        inc(i);
      end;
      val(num, n);
      for j := 1 to n do write(s[i]);
    end;
    i := i + 1;
  end;
end.

```

## C++

```

#include <iostream>
#include <cstring>

using namespace std;

int main()
{
  char s[81];
  int i = 0, n, l, j;
  cin >> s;
  l = strlen(s);
  while(i < l) {
    if (s[i] >= 'A' && s[i] <= 'Z')
      cout<<s[i++];
    else {
      n = 0;
      while(s[i] >= '0' && s[i] <= '9')
        n = n * 10 + (s[i++] - 48);
      for(j = 0; j < n; j++)
        cout << s[i];

      i++;
    }
  }
  cout << endl;
  return 0;
}

```

**Pascal****Program T3\_2014\_02;**

```

var
    n, m, i, nod: integer;

begin
    readln(n, m);
    while (n <> 0) and (m <> 0) do
        if n > m then n := n mod m
            else m := m mod n;

        nod := n + m;
        for i := 1 to nod do
            write(1);
        writeln;
    end.

```

**C++**

```

#include <iostream>

using namespace std;
int main()
{
    int n, m, i;
    cin >> n >> m;
    while(n != 0 && m != 0)
        if(n > m)
            n = n % m;
        else
            m = m % n;

    int nod = n + m;
    for (i = 0; i < nod; i++)
        cout << 1;
    cout << endl;

    return 0;
}

```

## Pascal

**Program T3\_2014\_03;**

```

var
    n, a, b, c, d, k: integer;

begin
    readln(n);
    k := 0;
    for a := 1 to n div 4 + 1 do
        for b := a to (n-a) div 3 + 1 do begin
            for c := b to (n-a-b) div 2 + 1 do begin
                d := n - a - b - c;
                if (d < c) then break;
                inc(k);
            end;
        end;
    end;

    writeln(k);
end.

```

## C++

```

#include <iostream>

using namespace std;

int main()
{
    int n, a, b, c, d, k = 0;
    cin >> n;
    int n1 = n/4 + 1;
    for (a=1; a<=n1; a++)
        for (b=a; b<=(n-a)/3+1; b++) {
            for (c=b; c<=(n-a-b)/2+1; c++) {
                d = n - a - b - c;
                if (d < c) break;
                k++;
            }
        }

    cout << k << endl;

    return 0;
}

```

C++

```

#include <iostream>
#include <cstring>
#define M 401
using namespace std;

int main()
{
    short a[M][M] = {{0}},
    h[4][2] = {{-1,0},{0,1},{1,0},{0,-1}};
    char s[M/2], sh[4] = {'N','E','S','W'};
    int i, j, i1, j1, k;

    cin >> s;
    int L = strlen(s);
    i1 = j1 = i = j = M/2;
    a[i][j] = 1;
    for (k=0; k<L; k++)
    {
        switch (s[k])
        {
            case 'N': i1--; break;
            case 'E': j1++; break;
            case 'S': i1++; break;
            case 'W': j1--; break;
        }
        if (a[i1][j1] == 0)
            a[i1][j1] = a[i][j] + 1;
        i = i1;
        j = j1;
    }

    while (a[i][j]!=1)
    {
        for(k=0; k<4; k++)
        {
            i1 = i + h[k][0];
            j1 = j + h[k][1];
            if (a[i][j] == a[i1][j1] + 1)
            {
                cout << sh[k];
                i = i1;
                j = j1;
                break;
            }
        }
    }
    cout << endl;
}

```

```

    return 0;
}

```

T3-2014-05.

Hərflər üzrə tullanma

Pascal

**Program T3\_2014\_05;**

```

var
  s: array[1..100000] of char;
  c: char;
  M: array[1..100000] of integer;
  km1, km2, i, j, k, n: integer;

begin
  readln(n, k);
  for i := 1 to n do
    read(s[i]);

  M[n] := 0;
  for i := n - 1 downto n - k do
    if (s[i] <> s[n]) then M[i] := 1
      else M[i] := 0;

  for i := n - k - 1 downto 1 do begin
    c := s[i];
    km1 := n;
    km2 := n;
    for j := i + 1 to i + k - 1 do
      if (c <> s[j]) then begin
        if (M[j] < km1) then
          km1 := M[j];
        end
        else begin
          if (M[j] < km2) then
            km2 := M[j];
          end;
        if (km2 < km1 + 1) then
          M[i] := km2
        else
          M[i] := km1 + 1;
        end;
      writeln(M[1]);
    end.

```

## C++

```
#include <iostream>
#include <stdio.h>

using namespace std;

int main()
{   int n, k;
    char s[100001], c;
    int M[100000], km1, km2, i, j;

    scanf("%d%d", &n, &k);
    scanf("%s", s);

    M[n-1] = 0;
    for (i = n-2; i >= n-k-1; i--)
        if (s[i] != s[n-1])
            M[i] = 1;
        else
            M[i] = 0;
    for (i = n-k-2; i >= 0; i--)
    {
        c = s[i];
        km1 = km2 = n;
        for(j = i+1; j < i+k; j++)
            if (c != s[j])
            {
                if (M[j] < km1) km1 = M[j];
            }
            else
            {
                if (M[j] < km2) km2 = M[j];
            }

        if (km2 < km1 + 1)
            M[i] = km2;
        else
            M[i] = km1 + 1;
    }
    printf("%d\n", M[0]);

    return 0;
}
```



### 6.3. HƏLLƏRİN TESTLƏNMƏSİ VƏ QIYMƏTLƏNDİRİLMƏSİ SİSTEMİ

Olimpiada iştirakçılarının respublika turunda həll etdikləri məsələlərin yoxlanılması və qiymətləndirilməsini, eləcə də mükafata layiq yerlərin müəyyənləşdirilməsini olimpiadanın münisflər heyəti həll edir. Bu proses, onun düzgün və obyektiv aparılması çətin və məsuliyyətli bir işdir. Olimpiadaların keçirildiyi ilk illərdə həllər “əllə” yoxlanırdı və kifayət qədər çox vaxt və əmək tələb edirdi. Ona görə də nəticələr bir və ya bir neçə gündən sonra elan edilirdi.

Kompüter şəbəkələrinin meydana çıxması və bu texnologiyanın geniş tətbiqi ilə əlaqədar olaraq olimpiada məsələlərinin həllərinin yoxlanılması işi də asanlaşdı. Son illər isə bu işlər tamamilə avtomatlaşdırılmışdır. Belə ki, hər bir olimpiada məsələsi üçün hazırlanmış testlər vasitəsilə həllər avtomatik yoxlanılır və iştirakçının topladığı balların əsasında onun tutduğu yer müəyyənləşir. Əlbəttə, həllərin yoxlanılması üçün tərtib edilən testlərin özünün üzərinə müəyyən şərtlər qoyulmalıdır. Yəni elə testlər hazırlanmalıdır ki, məsələnin həllində istifadə edilən bütün mümkün məntiqi variantların hamısını yoxlamağa imkan versin. Avtomatlaşdırılmış sistemdə şagirdlərin aldığı nəticələr hazır cavablarla müqayisə olunmaqla təhlil edilir.

Hər bir məsələ üçün ən azı 5 test hazırlanır. Bunlar aşağıdakı testlərdir:

- Sadə testlər.
- Xüsusi halları əhatə edən testlər.
- Ümumi testlər (kifayət qədər təsadüfi halları əhatə edir).
- Maksimum ölçülü testlər (giriş verilənlərinin daha böyük qiymətlərindən istifadə etməklə alqoritmin düzgünlüyünün yoxlanılması üçün).
- Antievristik testlər.

Hər məsələ üçün testlər hazır olduqdan sonra iştirakçıların həllərinin yoxlanılması üçün qiymətləndirmə metodikası işlənir:

- Hər bir məsələnin tam həlli üçün maksimal bal müəyyənləşdirilir.
- Maksimal bal həmin məsələ üçün testlər arasında paylanır.
- Hər məsələnin həlli hansı testdən keçirsə, ona müvafiq bal verilir və testlər üzrə həmin məsələyə görə yekun bal hesablanır. Məsələ, ümumiyyətlə, həll edilməmişdirsə, həmin məsələ sıfır balla qiymətləndirilir.
- Bütün məsələlər üzrə yekun ballar cəmlənərək iştirakçının ümumi yekun balı hesablanır.

Nümunə üçün aşağıda 2011-ci ildə keçirilmiş respublika olimpiadası işlərinin yoxlanılması üçün istifadə olunan bir forma verilib. Göründüyü kimi, 033 kodla

şifrlənmiş iştirakçı məsələlərə uyğun olaraq 23, 30, 38, 28 və 30 bal, cəmi isə 149 bal qazanıb.

İştirak- çının kodu	Məsələnin kodu	Testlər üzrə ballar/ İştirakçının qazandığı ballar					Məsələ üzrə maksimal bal	İştirak- çının balı
		1	2	3	4	5		
<b>033</b>	T3-2011-01	2	3	7	8	10	30	
		2	3	0	8	10		<b>23</b>
	T3-2011-02	5	7	8	10	15	45	
		5	7	8	10	0		<b>30</b>
	T3-2011-03	5	5	8	12	20	50	
		5	5	8	0	20		<b>38</b>
	T3-2011-04	5	5	7	8	10	35	
		5	5	0	8	10		<b>28</b>
	T3-2011-05	5	5	10	10	10	40	
		5	5	10	10	0		<b>30</b>
	İştirakçının yekun balı							<b>149</b>

Hər bir iştirakçı üçün yekun bal hesablandıqdan sonra onlar azalma sırası ilə düzülür və qaliblər müəyyənləşdirilir. Sonra isə şifrlər açılaraq qaliblərin adları elan edilir.

Nümunədə verildiyi kimi, hər bir məsələnin həlli çətinlik dərəcəsinə görə müəyyən maksimal balla qiymətləndirilir. Məsələnin çətinlik dərəcəsinə münisiflər heyəti müəyyənləşdirir. Burada məsələlərin çətinlik dərəcəsinə iki yanaşma mümkündür:

- münisiflər heyəti üzvlərinin subyektiv fikirlərinə əsasən qabaqcadan müəyyənləşdirmə;
- iştirakçılar tərəfindən məsələlərin həlledilmə dərəcəsindən asılı olaraq sonda müəyyənləşdirmə.

Hazırda olimpiada məsələlərinin çətinlik dərəcəsinin müəyyənləşdirilməsi üçün müxtəlif metodikalar işlənib praktikada tətbiq edilir. Bəzən isə bütün məsələlər eyni balla qiymətləndirilir.

Bu metodikanın təkmilləşdirilməsi, eləcə də müxtəlif psixoloji amillərin nəzərə alınması yolu ilə obyektivliyi daha çox təmin etmək məqsədilə hər il bu iş informatika üzrə olimpiadanın münisiflər heyəti tərəfindən təkmilləşdirilir.



## VII BÖLÜM

### ƏLAVƏLƏR

#### 7.1. RESPUBLİKA İNFORMATİKA OLİMPİADALARININ QALİBLƏRİ (1989-2014)

**A**

**İllər üzrə**

Yer	Adı, soyadı, atasının adı	Oxuduğu məktəb	Sınıf
<b>1989</b>			
I	Məhərrəmov Musa Əli oğlu	Bakı şəhəri 23 saylı məktəb	IX
<b>1990</b>			
I	Babayev Vüqar Həbib oğlu	Bakı şəhəri 82 saylı məktəb	XI
<b>1991</b>			
I	Qəhrəmanova Xumar Şirzad qızı	Fizika, riyaziyyat və informatika təmayüllü respublika internat məktəbi	XI
I	Gül Qasım Elvin oğlu	Bakı şəhəri 134 saylı məktəb	XI
III	Şərifov Çingiz Talib oğlu	Bakı şəhəri 20 saylı məktəb	XI
<b>1992</b>			
Ölkədə yaranmış ictimai-siyasi durumla əlaqədar olaraq respublika turu keçirilməyib.			
Yer	Adı, soyadı, atasının adı	Oxuduğu məktəb	Sınıf
<b>1993</b>			
I	Sultanov Ənvər Çingiz oğlu	Fizika, riyaziyyat və informatika təmayüllü respublika internat məktəbi	X
<b>1994</b>			
I	Sultanov Ənvər Çingiz oğlu	Fizika, riyaziyyat və informatika təmayüllü lisey	XI
II	Bəxşi Rasim Rafiq oğlu	H.b.Mahmudbəyov adına 2 saylı texniki humanitar lisey	X
III	Ləmbəranski Surxay Saday oğlu	H.b.Mahmudbəyov adına 2 saylı texniki humanitar lisey	X

1995			
13-17 mart hadisələri ilə bağlı olaraq respublika turu keçirilməyib.			
Yer	Adı, soyadı, atasının adı	Oxuduğu məktəb	Sınıf
1996			
I	Əliyev Aqil Əkbər oğlu	Bakı şəhəri 287 saylı məktəb	XI
1997			
II	Nəstun Denis Nikolayeviç	Bakı şəhəri 264 saylı məktəb	X
III	Yusifov Elşad Barat oğlu	Sumqayıt şəhəri Texniki təmayüllü özəl lisey	XI
1998			
I	Nəstun Denis Nikolayeviç	Bakı şəhəri 264 saylı məktəb	XI
III	Rüstəmov Nicat Azər oğlu	Sumqayıt Özəl Türk liseyi	XI
III	Əsgərov Aslan Babək oğlu	“Zəngi” liseyi	XI
1999			
I	Lyapın Timur Ramiz oğlu	Bakı şəhəri 160 saylı məktəb	XI
I	Məmmədov Murad Şaiq oğlu	Bakı Özəl Türk liseyinin Mingəçevir filialı	X
II	Kərimov Rüstəm Zəkəriyyə oğlu	Bakı Özəl Türk liseyinin Mingəçevir filialı	XI
II	Vəliyev Emin Elxan oğlu	Sumqayıt Özəl Türk liseyi	X
III	Məmmədov Ramil Rauf oğlu	Bakı Özəl Türk liseyi	X
III	Rəhimov Vüqar Eldar oğlu	Bakı Özəl Türk liseyi	XI
2000			
I	Məmmədov Seymur Vəli oğlu	Bakı Özəl Türk liseyi	XI
I	Məmmədov Murad Şaiq oğlu	Bakı Özəl Türk liseyinin Mingəçevir filialı	XI
II	Əfəndiyev Məzahir Tahir oğlu	Şəki Azərbaycan-Türkiyə özəl liseyi	XI
III	Məmmədov Ramil Rauf oğlu	Bakı Özəl Türk liseyi	XI
2001			
I	Bağirov Ramin Əbdüləhəd oğlu	Lənkəran şəhəri 1 saylı məktəb	XI
I	Davudov Rəcəb Murad oğlu	Bakı Özəl Türk liseyi	XI
II	Səttarov Elçin İmanqulu oğlu	Bakı Özəl Türk liseyinin Lənkəran filialı	XI
III	Əzizov Tural Abdulsəid oğlu	Bakı Özəl Türk liseyi	XI
III	Abdullayev Kamandar Elbrus oğlu	Şəki Azərbaycan-Türkiyə özəl liseyi	X
2002			
I	Quliyev Teymur Tofiq oğlu	Bakı Özəl Türk liseyi	IX
I	Əhmədov Fərid Rahim oğlu	Fizika, riyaziyyat və informatika təmayüllü lisey	X
I	İbrahimov Maqsud Üzeyir oğlu	H.b.Mahmudbəyov adına 2 saylı texniki humanitar lisey	XI
II	Muxtarov Rüstəm Allahverdi oğlu	Bakı Özəl Türk liseyi	X
II	Həsənov Tofiq Nazim oğlu	Bakı Özəl Türk liseyi	IX
III	Fərəcov Amal Zahir oğlu	Sumqayıt Özəl Türk liseyi	IX
III	Əliyev İsa Atamoğlu oğlu	Sumqayıt Özəl Türk liseyi	X

Yer	Adı, soyadı, ata adı	Oxuduğu məktəb	Sinif
<b>2003</b>			
I	Əliyev Şahin Məhəmməd oğlu	Bakı Özəl Türk liseyinin Ağdaş filialı	XI
I	Əhmədov Fərid Rahim oğlu	Fizika, riyaziyyat və informatika təmayüllü lisey	XI
I	Quliyev Teymur Tofiq oğlu	Bakı Özəl Türk liseyi	X
II	Şiriyev Elvin Əlirza oğlu	Bakı Özəl Türk liseyinin Lənkəran filialı	XI
II	Məmmədov Teymur Məmməd oğlu	Şəki Azərbaycan-Türkiyə özəl liseyi	XI
II	Əliyev Ramil Nurəddin oğlu	Bakı şəhəri A.Hüseynzadə adına 20 saylı məktəb-lisey	X
III	Babayev Fərid Əbasət oğlu	Bakı Özəl Türk liseyinin Quba filialı	XI
III	Verdiyev Anar Rövşən oğlu	Gəncə şəhəri 1 saylı məktəb	IX
<b>2004</b>			
I	Həsənov Vaqif Nazim oğlu	Bakı Özəl Türk liseyi	X
I	Həsənov Tofiq Nazim oğlu	Bakı Özəl Türk liseyi	XI
I	Həşimov Rüstəm Rəsul oğlu	Bakı Özəl Türk liseyinin Lənkəran filialı	VIII
II	Neymanov Bəxtiyar İslam oğlu	Bakı Avropa liseyi	X
II	Quliyev Teymur Tofiq oğlu	Bakı Özəl Türk liseyi	XI
II	Şeydayev Pərviz Eyvaz oğlu	Bakı Özəl Türk liseyinin Quba filialı	IX
III	Əliyev Adil Elçin oğlu	Bakı şəhəri 83 saylı məktəb-lisey	X
III	Verdiyev Anar Rövşən oğlu	Gəncə şəhəri 1 saylı məktəb	X
III	Kərimov Nurlan Ramazan oğlu	Bakı Özəl Türk liseyinin Dədə Qorqud filialı	IX
III	Məmmədli Türkel Nazim qızı	Sumqayıt Özəl Türk liseyi	X
III	Bayramov Fuad Tərlən oğlu	Bakı Özəl Türk liseyinin Quba filialı	X
<b>2005</b>			
I	Səlimov Aqşin Səxavət oğlu	Bakı şəhəri 134 saylı məktəb	IX
I	Abbasov Əfqan Hikmət oğlu	Şəki Azərbaycan-Türkiyə özəl liseyi	X
I	Həsənov Vaqif Nazim oğlu	Bakı Özəl Türk liseyi	XI
II	Həşimov Rüstəm Rəsul oğlu	Bakı Özəl Türk liseyinin Lənkəran filialı	IX
II	Mahmudzadə Cavid Namiq oğlu	Bakı şəhəri A.Hüseynzadə adına 20 saylı məktəb-lisey	VIII
II	Kərimov Nurlan Ramazan oğlu	Fizika, riyaziyyat və informatika təmayüllü lisey	X

Yer	Adı, soyadı, ata adı	Oxuduğu məktəb	Sinif
II	Şeydayev Pərviz Eyvaz oğlu	Bakı Özəl Türk liseyinin Quba filialı	X
II	Bayramov Fuad Tərən oğlu	Bakı Özəl Türk liseyinin Quba filialı	XI
II	Neymanov Bəxtiyar İslam oğlu	Bakı Avropa liseyi	XI
III	Əliyev Ayxan Arif oğlu	Bakı Özəl Türk liseyi	IX
III	Fərəcullayev Şamil Əliheydər oğlu	Bakı Özəl Türk liseyi	IX
III	Bəylərov Cəfər Camal oğlu	Şərur Türk liseyi	X
III	Məmmədov Məmməd Bəxtiyar oğlu	Bakı şəhəri 164 saylı məktəb	XI
III	Məmmədli Türkel Nazim qızı	Sumqayıt Özəl Türk liseyi	XI
III	Maqsudov Bəhrüz Azər oğlu	Bakı şəhəri 23 saylı məktəb	XI
<b>2006</b>			
I	Mahmudzadə Cavid Namiq oğlu	Bakı şəhəri A.Hüseynzadə adına 20 saylı məktəb-lisey	IX
I	Səlimov Aqşin Səxavət oğlu	Beynəlxalq Təhsil Kompleksi (Bakı şəhəri 134 saylı məktəb)	X
I	Abbasov Əfqan Hikmət oğlu	Şəki Azərbaycan-Türkiyə özəl liseyi	XI
II	Əliyev Ayxan Arif oğlu	Bakı Özəl Türk liseyi	X
II	Həsənzadə Toğrul Həsən oğlu	Sumqayıt Özəl Türk liseyi	X
II	Kərimov Nurlan Ramazan oğlu	Fizika, riyaziyyat və informatika təmayüllü lisey	XI
II	Cəfərova Türkan Turan qızı	Bakı şəhəri 264 saylı məktəb	XI
III	Rüstəmli Vüqar Oqtay oğlu	Bakı Özəl Türk liseyi	IX
III	Hüseynzadə Cavid Urfan oğlu	Bakı şəhəri 134 saylı məktəb	X
III	Qasımov Fərid Etibar oğlu	“Elitar” gimnaziya	XI
III	Əliyev Rəşad Əbülfəz oğlu	Bakı şəhəri 147 saylı texniki-humanitar lisey	XI
III	Şeydayev Pərviz Eyvaz oğlu	Bakı Özəl Türk liseyinin Quba filialı	XI
<b>2007</b>			
I	Əliyev Ayxan Arif oğlu	Bakı Özəl Türk liseyi	XI
I	Mahmudzadə Cavid Namiq oğlu	Bakı şəhəri A.Hüseynzadə adına 20 saylı məktəb-lisey	X
II	Orucov Abid İlham oğlu	Bakı Özəl Türk liseyi	IX
II	Səlimov Aqşin Səxavət oğlu	Bakı şəhəri 134 saylı məktəb	XI
III	Rəşidzadə İsfəndiyar Asif oğlu	Bakı Özəl Türk liseyinin Dədə Qorqud filialı	IX

Yer	Adı, soyadı, ata adı	Oxuduğu məktəb	Sinif
III	Rüstəmli Turan Eldar oğlu	Bakı şəhəri “İntellekt” məktəb-liseyi (6 saylı məktəb)	X
<b>2008</b>			
I	Neymanov Tural İslam oğlu	Bakı Özəl Türk liseyi	IX
I	Abbasov Yaşar Akif oğlu	Bakı Özəl Türk liseyi	X
I	Mahmudzadə Cavid Namiq oğlu	Bakı şəhəri A.Hüseynzadə adına 20 saylı məktəb-lisey	XI
II	Rüstəmli Turan Eldar oğlu	Bakı şəhəri “İntellekt” məktəb-liseyi (6 saylı məktəb)	XI
II	Hüseynzadə Suel İmran oğlu	“Elitar” gimnaziya	X
II	Fərmanov Fərman İqbal oğlu	Bakı Özəl Türk liseyi	IX
II	Axundzadə Elkin Azər oğlu	Bakı şəhəri 147 saylı texniki-humanitar lisey	XI
III	Səlimli Asim Muxtar oğlu	Bakı şəhəri 134 saylı məktəb	XI
III	İbayev Sərxan Ələddin oğlu	Sumqayıt Özəl Türk liseyi	IX
III	Səlimov Anar Elmar oğlu	Bakı şəhəri 264 saylı məktəb-lisey	IX
III	Kərimli Taleh Valeh oğlu	Bakı şəhəri 100 saylı məktəb	X
III	Cəfərli Sənan İlham oğlu	Bakı Özəl Türk liseyinin Quba filialı	X
III	Yəhyayev Emin Borisoviç	Bakı Avropa liseyi	IX
III	Quliyev Əyyub Tahir oğlu	Bakı Özəl Türk liseyinin Ağdaş filialı	IX
<b>2009</b>			
I	Fərmanov Fərman İqbal oğlu	Bakı Özəl Türk liseyi	X
I	Neymanov Tural İslam oğlu	Bakı Özəl Türk liseyi	X
I	Abbasbəyli Zaur Azər oğlu	Bakı Özəl Türk liseyinin Mingəçevir filialı	IX
II	İbayev Sərxan Ələddin oğlu	Sumqayıt Özəl Türk liseyi	X
II	Qədimbəyli Sahil Mikayıl oğlu	Bakı Özəl Türk liseyinin Dədə Qorqud filialı	IX
III	Abbasov Yaşar Akif oğlu	Bakı Özəl Türk liseyi	XI
III	Hüseynzadə Suel İmran oğlu	“Elitar” gimnaziya	XI
III	Səbzizəyev Valeh Aqşin oğlu	Bakı şəhəri 291 saylı “Ekologiya” liseyi	X
III	Muradlı Polad Güloğlan oğlu	H.Əliyev adına Müasir Təhsil Kompleksi	X
III	İsayeva Səbinə Zaur qızı	Bakı şəhəri 83 saylı məktəb-lisey	IX
III	Quluzadə Toğrul Nazim oğlu	Sumqayıt şəhəri Təbiət elmləri təmayüllü gimnaziya	IX
III	Gülməmmədov Tural Vüqar oğlu	Bakı Özəl Türk liseyi	VIII
<b>2010</b>			
I	Salahlı Azad Qəzənfər oğlu	Bakı Özəl Türk liseyi	X
I	Fərmanov Fərman İqbal oğlu	Bakı Özəl Türk liseyi	XI
I	Gülməmmədov Tural Vüqar oğlu	Bakı Özəl Türk liseyinin Dədə Qorqud filialı	IX

Yer	Adı, soyadı, atasının adı	Oxuduğu məktəb	Sınıf
I	Süleymanov Allahverdi Bəhruz oğlu	Bakı Avropa liseyi	VIII
II	Hacıyev Elşən Əkbər oğlu	Bakı Özəl Türk liseyi	IX
II	İsayeva Səbinə Zaur qızı	Bakı şəhəri akademik Z.Əliyeva adına məktəb	X
II	Abbasbəyli Zaur Azər oğlu	Bakı Özəl Türk liseyinin Mingəçevir filialı	X
II	Neymanov Tural İslam oğlu	Bakı Özəl Türk liseyi	XI
III	Vəliyev Əli Salman oğlu	Bakı şəhəri 134 saylı məktəb	XI
III	Qədimbəyli Sahil Mikayıl oğlu	Bakı Özəl Türk liseyinin Dədə Qorqud filialı	X
III	Qaziyev Rəşid Mübariz oğlu	Sumqayıt Özəl Türk liseyi	VIII
III	İbayev Sərxan Ələddin oğlu	Sumqayıt Özəl Türk liseyi	XI
<b>2011</b>			
I	Abbasbəyli Zaur Azər oğlu	Bakı Özəl Türk liseyinin Mingəçevir filialı	XI
I	Gülməmmədov Tural Vüqar oğlu	Bakı Özəl Türk liseyinin Dədə Qorqud filialı	X
I	Salahlı Azad Qəzənfər oğlu	Bakı Özəl Türk liseyi	XI
I	Qaziyev Rəşid Mübariz oğlu	Sumqayıt Özəl Türk liseyi	IX
II	Məhəmmədli Məhəmmədəli Əliağa oğlu	Sumqayıt şəhəri 12 saylı tam orta məktəb	IX
II	Nağızadə Həsən Əli oğlu	Fizika, riyaziyyat və informatika təmayüllü lisey	X
II	Kərimli Əli İdrak oğlu	Türk Dəyanət Vəqfinin Bakı Türk liseyi	X
III	Vəliyev Məhəmməd Tağı oğlu	Sumqayıt şəhəri Təbiət elmləri təmayüllü gimnaziya	XI
III	Əhmədov Elmi Fəxrəddin oğlu	Bakı Özəl Türk liseyinin Mingəçevir filialı	X
III	Hacıyev Elşən Əkbər oğlu	Bakı Özəl Türk liseyi	X
III	Tahirov Ceyhun Firuz oğlu	Bakı şəhəri akademik Z.Əliyeva adına məktəb	IX
III	Kərimova Səlminaz Əjdər qızı	Neftçala rayonu Qədimkənd tam orta məktəbi	XI
<b>2012</b>			
I	Qaziyev Rəşid Mübariz oğlu	Sumqayıt Özəl Türk liseyi	X
I	Gülməmmədov Tural Vüqar oğlu	Bakı Özəl Türk liseyinin Dədə Qorqud filialı	XI
I	Əlizadə Tahir Tokay oğlu	Bakı Özəl Türk liseyi	IX
I	Əhmədov Elmi Fəxrəddin oğlu	Bakı Özəl Türk liseyinin Mingəçevir filialı	XI



Yer	Adı, soyadı, atasının adı	Oxuduğu məktəb	Sinif
I	Kərimli Əli İdrak oğlu	Türk Dəyanət Vəqfinin Bakı Türk liseyi	XI
III	Nağızadə Həsən Əli oğlu	Fizika, riyaziyyat və informatika təmayüllü lisey	XI
III	Paşayev Sənan Natiq oğlu	Bakı Özəl Türk liseyi	VIII
III	Bilalov Zaur Səid oğlu	Bakı Özəl Türk liseyi	IX
III	Məhəmmədli Məhəmmədəli Əliağa oğlu	Sumqayıt şəhəri 12 saylı məktəb	X
III	Qasimov Elmin Vahid oğlu	Bakı şəhəri 264 saylı məktəb-lisey	X
III	Ömərov Samir Şakir oğlu	Bakı Dövlət Universitetinin "Gənc istedadlar" liseyi	X
III	Məmmədov Xəyal Elşən oğlu	Bakı şəhəri 201 saylı məktəb	XI
III	İsmayılzadə Elşad Şükür oğlu	İmişli şəhəri 2 saylı məktəb	XI
<b>2013</b>			
I	Qaziyev Rəşid Mübariz oğlu	Sumqayıt Özəl Türk liseyi	XI
I	Əlizadə Tahir Tokay oğlu	Bakı Özəl Türk liseyi	X
I	Paşayev Sənan Natiq oğlu	Bakı Özəl Türk liseyi	IX
II	Süleymanov Allahverdi Bəhrüz oğlu	Bakı Avropa liseyi	XI
II	Talıbov Murad Vəli oğlu	Sumqayıt Özəl Türk liseyi	X
II	Şirinov Əliağa Faiq oğlu	Sumqayıt Özəl Türk liseyi	IX
<b>2014</b>			
I	İbrahimli Fətex Xaqani oğlu	Bakı Özəl Türk liseyinin Dədə Qorqud filialı	IX
I	Şirinov Əliağa Faiq oğlu	Sumqayıt Özəl Türk liseyi	X
I	Əlizadə Tahir Tokay oğlu	Bakı Özəl Türk liseyi	XI
II	Məmmədov İbrahim Nasir oğlu	Bakı Özəl Türk liseyi	IX
II	Cəbrayılzadə Elgün Ayaz oğlu	Bakı Özəl Türk liseyi	IX
II	Paşayev Sənan Natiq oğlu	Bakı Özəl Türk liseyi	IX
<b>Yer</b>	<b>Adı, soyadı, ata adı</b>	<b>Oxuduğu məktəb</b>	<b>Sinif</b>
III	Ağayev Rüstəm Nemət oğlu	Bakı Özəl Türk liseyinin Dədə Qorqud filialı	VIII
III	Eyyubov Kamal Əmirxan oğlu	Türk Dəyanət Vəqfinin Bakı Türk liseyi	IX
III	Süleymanlı Samir Azər oğlu	Şəki Azərbaycan-Türkiyə özəl liseyi	IX

№	Qalıbların						
	soyadı	adı	atasının adı	qalib olduğu illər			
1	Abbasbəyli	Zaur	Azər	2009	2010	2011	
2	Abbasov	Əfqan	Hikmət	2005	2006		
3	Abbasov	Yaşar	Akif	2008	2009		
4	Abdullayev	Kamandar	Elbrus	2001			
5	Ağayev	Rüstəm	Nemət	2014			
6	Axundzadə	Elkin	Azər	2008			
7	Babayev	Fərid	Əbasət	2003			
8	Babayev	Vüqar	Həbib	1990			
9	Bağırov	Ramin	Əbdüləhəd	2001			
10	Bayramov	Fuad	Tərən	2004	2005		
11	Bəxşi	Rasim	Rafiq	1994			
12	Bəylərov	Cəfər	Camal	2005			
13	Bilalov	Zaur	Səid	2012			
14	Cəbrayılzadə	Elgün	Ayaz	2014			
15	Cəfərli	Sənan	İlham	2008			
16	Cəfərova	Türkan	Turan	2006			
17	Davudov	Rəcəb	Murad	2001			
18	Eyyubov	Kamal	Əmirxan	2014			
19	Əfəndiyev	Məzahir	Tahir	2000			
20	Əhmədov	Fərid	Rahim	2002	2003		
21	Əhmədov	Elmi	Fəxrəddin	2011	2012		
22	Əliyev	İsa	Atamoğlu	2002			
23	Əliyev	Şahin	Məhəmməd	2003			
24	Əliyev	Aqil	Əkbər	1996			
25	Əliyev	Adil	Elçin	2004			
26	Əliyev	Ayxan	Arif	2005	2006	2007	
27	Əliyev	Rəşad	Əbülfəz	2006			
28	Əliyev	Ramil	Nurəddin	2003			
29	Əlizadə	Tahir	Tokay	2012	2013	2014	

№	Qaliblərin						
	soyadı	adı	atasının adı	qalib olduğu illər			
30	Əsgərov	Aslan	Babək	1998			
31	Əzizov	Tural	Abdulsəid	2001			
32	Fərəcov	Amal	Zahir	2002			
33	Fərəcullayev	Şamil	Əliheydər	2005			
34	Fərmanov	Fərman	İqbal	2008	2009	2010	
35	Gül	Qasım	Elvin	1991			
36	Gülməmmədov	Tural	Vüqar	2009	2010	2011	2012
37	Hacıyev	Elşən	Əkbər	2010	2011		
38	Həsənov	Tofiq	Nazim	2004			
39	Həsənov	Vaqif	Nazim	2004	2005		
40	Həsənzadə	Toğrul	Həsən	2006			
41	Həşimov	Rüstəm	Rəsul	2004	2005		
42	Hüseynzadə	Suel	İmran	2008	2009		
43	Hüseynzadə	Cavid	Urfan	2006			
44	İbayev	Sərxan	Ələddin	2008	2009	2010	
45	İbrahimli	Fateh	Xaqani	2014			
46	İbrahimov	Maqsud	Üzeyir	2002			
47	İsayeva	Səbinə	Zaur	2009	2010		
48	İsmayılzadə	Elşad	Şükür	2012			
49	Kərimli	Taleh	Valeh	2008			
50	Kərimli	Əli	İdrak	2011	2012		
51	Kərimov	Rüstəm	Zəkəriyyə	1999			
52	Kərimov	Nurlan	Ramazan	2004	2005	2006	
53	Kərimova	Səlminaz	Əjdər	2011			
54	Ləmbəranski	Surxay	Saday	1994			
55	Lyapın	Timur	Ramiz	1999			
56	Mahmudzadə	Cavid	Namiq	2005	2006	2007	2008
57	Maqsudov	Bəhrüz	Azər	2005			
58	Məhəmmədli	Məhəmmədəli	Əliağa	2011	2012		
59	Məhərrəmov	Musa	Əli	1989			
60	Məmmədli	Türkel	Nazim	2004	2005		

№	Qaliblərin						
	soyadı	adı	atasının adı	qalib olduğu illər			
61	Məmmədov	Murad	Şaiq	1999	2000		
62	Məmmədov	İbrahim	Nasir	2014			
63	Məmmədov	Seymur	Vəli	2000			
64	Məmmədov	Ramil	Rauf	1999	2000		
65	Məmmədov	Teymur	Məmməd	2003			
66	Məmmədov	Məmməd	Bəxtiyar	2005			
67	Məmmədov	Xəyal	Elşən	2012			
68	Muradlı	Polad	Güloğlan	2009			
69	Muxtarov	Rüstəm	Allahverdi	2002			
70	Nağızadə	Həsən	Əli	2011	2012		
71	Neştun	Denis	Nikolayeviç	1997	1998		
72	Neymanov	Bəxtiyar	İslam	2004	2005		
73	Neymanov	Tural	İslam	2008	2009	2010	
74	Orucov	Abid	İlham	2007			
75	Paşayev	Sənan	Natiq	2012	2013	2014	
76	Qasimov	Fərid	Etibar	2006			
77	Qasimov	Elmin	Vahid	2012			
78	Qaziyev	Rəşid	Mübariz	2010	2011	2012	2013
79	Qədimbəyli	Sahil	Mikayıl	2009	2010		
80	Qəhrəmanova	Xumar	Şirzad	1991			
81	Quliyev	Teymur	Tofiq	2002	2003	2004	
82	Quliyev	Əyyub	Tahir	2008			
83	Quluzadə	Toğrul	Nazim	2009			
84	Ömərov	Samir	Şakir	2012			
85	Rəhimov	Vüqar	Eldar	1999			
86	Rəşidzadə	İsfəndiyar	Asif	2007			
87	Rüstəmli	Vüqar	Oqtay	2006			
88	Rüstəmli	Turan	Eldar	2007	2008		
89	Rüstəmov	Nicat	Azər	1998			
90	Salahlı	Azad	Qəzənfər	2010	2011		
91	Səbzizəyev	Valeh	Aqşin	2009			


№	Qaliblərin						
	soyadı	adı	atasının adı	qalib olduğu illər			
92	Səlimov	Aqşin	Səxavət	2005	2006	2007	
93	Səlimli	Asim	Muxtar	2008			
94	Səlimov	Anar	Elmar	2008			
95	Səttarov	Elçin	İmanqulu	2001			
96	Sultanov	Ənvər	Çingiz	1993	1994		
97	Süleymanlı	Samir	Azər	2014			
98	Süleymanov	Allahverdi	Bəhrüz	2010	2013		
99	Şeydayev	Pərviz	Eyvaz	2004	2005	2006	
100	Şərifov	Çingiz	Talib	1991			
101	Şirinov	Əliağa	Faiq	2013	2014		
102	Şiriyev	Elvin	Əlirza	2003			
103	Tahirov	Ceyhun	Firuz	2011			
104	Talıbov	Murad	Vəli	2013			
105	Verdiyev	Anar	Rövşən	2003	2004		
106	Vəliyev	Emin	Elxan	1999			
107	Vəliyev	Əli	Salman	2010			
108	Vəliyev	Məhəmməd	Tağı	2011			
109	Yəhyayev	Emin	Borisoviç	2008			
110	Yusifov	Elşad	Barat	1997			

**1-ci Respublika İnformatika Olimpiadasının (1989) qalibi,  
2-ci Ümumittifaq İnformatika Olimpiadasının bürünc medalçısı**



**Məhərrəmov Musa Əli oğlu**

## Beynəlxalq İnformatika Olimpiadalarının medalçıları

1		<b>Seyidli Elnur Fəttah oğlu</b>	(1996, bürünc)
2		<b>Neştun Denis Nikolayeviç</b>	(1997, bürünc)
3		<b>Həsənov Tofiq Nazim oğlu</b>	(2003, gümüş)
4		<b>İbrahimov Maqsud Üzeyir oğlu</b>	(2004, bürünc)
5		<b>Salahlı Azad Qəzənfər oğlu</b>	(2011, bürünc)

## 7.2. SƏNƏDLƏR

Азәрбајчан ССР  
ХАЛГ ТƏҺСИЛИ  
НАЗИРЛИЈИ

Бақы шəһ., Һөкүмәт еви. Тел.: 93-70-83.



МИНИСТЕРСТВО  
НАРОДНОГО ОБРАЗОВАНИЯ  
Азербайджанской ССР

370016, г. Баку, Дом правительства. Тел.: 93-70-83.

Ә М Р

П Р И К А З

01

199... ил/г.

Бақы шəһ.  
г. Бақы

№ 05

Азәрбајчан ССР Халг Тəһсили Назирлијинин  
1245 №-ли, 06.12.89-чу ил тарихли әмринә  
әлавә һаггында

"Мәктәблиләрин республика фəнн олимпиадалары галиблəринин али мөктəблərə имтаһансыз гəбул едилмəsi һаггында Назирлијин 819 №-ли 12 ијул 1989-чу ил тарихли әмринә әлавә һаггында" 1245 №-ли 06.12.89-чу ил тарихли әмринә әлавә олараг мтəј-јөн едилсин ки, информатика вә һесаблама техникасы фəнни үзрə олимпиадада биринчи јери тутан шакирдлəрин али мөктəблərə гəбулу заманы һəмин фəнн физика вә ријазийјат фəнлəринә мұвафиг һесаб олунур.

НАЗИР



Р. Б. ФЕЛЗУЛЛАЈЕВ

ПОДПИСАНЫ  
ВЕРНО



Республика ташкилат комитәси информатика  
фәнн үзрә республика олимпиадасынын галиби  
физика, ривазият в информатика тәмајулу Республика  
интернат мәктәбин XI синиф шакирди  
Гүлрәманова Хунар Ширзад ғызини  
I дәрәчәли дипломла тәлтиф едир

Тәшкилат комитәсинин

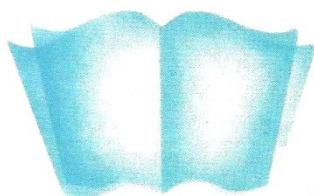
Мүнсифлар пәјәтинин



март 1991 ил №18

сәдри  
Халк тәһсилат министрлиги

# ФӘХРИ ФӘРМАН



Азәрбајҹан Республикасы *Мәһсил*  
Назирлијинин 22 апрел 1994 *чи ил* \_\_\_\_\_

тарихли 248 №-ли *апрел ил* Информатика үзрә  
Республика олимпиадасы вә компютер программалары  
мусабигәсинин галиби *И. Махмудбәјов* адна техника-ку  
манитар елими итсәјимин *х синиф* шәһәрди  
*Бахши Расим Рафиг* оьау

Фәхри Фәрманла тәлтиф едиле.



*Мәһсил* /*И. Тәсәков*/.  
22 *апрел* 1994 *ил* № \_\_\_\_\_



**AZƏRBAYCAN RESPUBLİKASI TƏHSİL NAZİRİNİN  
ƏMRİ**

№ 1008, Bakı şəh., Xətai prospekti, 49. Tel.: 496-06-47

№

04 02 2008 il

*A. Mahmudov*  
*Rəhbərlik*  
*05.02.08*  
**Dünya fənn olimpiadaları iştirakçılarının  
dərş məşğələlərindən azad edilməsi barədə**

Azərbaycan Respublikası Prezidentinin 17 aprel 2006-cı il tarixli, 1414 nömrəli Sərəncamı ilə təsdiq edilmiş «Xüsusi istedadla malik olan uşaqların (gənclərin) yaradıcılıq potensialının inkişafı üzrə Dövlət Proqramı»nın 5.14-cü bəndinin icrası ilə bağlı

**ƏMR EDİRƏM:**

1. Dünya fənn olimpiadalarında iştirak üçün seçilmiş istedadlı şagirdlərin hazırlıq səviyyəsinin yüksəldilməsinə daha əlverişli və səmərəli şərait yaratmaq məqsədilə onların hər ilin noyabr və aprel aylarında dərş məşğələlərindən azad olunması məqsəduyğun hesab edilsin.
2. Fizika, riyaziyyat, kimya, biologiya və informatika üzrə fənn olimpiadalarının münisflər heyətlinin sədrləri (M.Murquzov, R.Quliyev, V.Abbasov, Ə.Fərəcov, R.Mahmudov) soyadı, adı, atasının adı, oxuduğu məktəb və sinif göstərilməklə dünya fənn olimpiadalarında iştirakı nəzərdə tutulan şagirdlər barədə hər ilin mart və oktyabr aylarında Nazirliyə məlumat versinlər.
3. İctimaiyyətlə əlaqə şöbəsi (B.Hüseynzadə) əmrin «Azərbaycan müəllimi» qəzetində və «Təhsil xəbərləri» məcmuəsində dərc olunmasını təmin etsin.
4. Əmrin icrasına nəzarət nazir müavini İ.İsgəndərova həvalə olunsun.



Misir Mərdanov

**"Məktəblilərin fənn olimpiadalarının təşkili və keçirilməsi Qaydaları"nın  
təsdiq olunması barədə  
Azərbaycan Respublikası Təhsil nazirinin əmri**

İstedadlı şagirdlərin aşkara çıxarılmasında əhəmiyyətli rol oynayan fənn olimpiadalarının təşkili və keçirilməsi ilə bağlı işləri obyektiv və şəffaf qaydada həyata keçirmək məqsədilə "Azərbaycan Respublikasının Təhsil Nazirliyi haqqında Əsasnamə"nin 8.14-cü və 13.5-ci bəndlərini rəhbər tutaraq

**ƏMR EDİRƏM:**

1. "Məktəblilərin fənn olimpiadalarının təşkili və keçirilməsi Qaydaları" təsdiq edilsin (əlavə olunur).

2. Təhsilin inkişafı proqramları şöbəsi (E.Əmrullayev) 10 gün ərzində Respublika təşkilat komitəsinin tərkibinə dair təklifləri təqdim etsin.

3. Naxçıvan MR Təhsil Nazirliyi, Bakı Şəhəri üzrə Təhsil İdarəsi, Gəncə Şəhər Təhsil İdarəsi, rayon (şəhər) təhsil şöbələri, Təhsil Nazirliyinə tabe olan ümumi təhsil müəssisələrinin direktorları bu "Qaydalar"dan irəli gələn zəruri tədbirləri həyata keçirsinlər.

4. İqtisadiyyat şöbəsi (R.Orucov) fənn olimpiadaların təşkili, keçirilməsi, qaliblərin təltif olunması, münəfiqlər heyəti üzvlərinin, müəllimlərin mükafatlandırılması ilə əlaqədar xərclərin smetasını hazırlayıb təsdiq üçün Nazirliyin rəhbərliyinə təqdim etsin.

5. İnformasiya şöbəsi (E.Əliyev) bu əmrin "Azərbaycan müəllimi" qəzetində dərc edilməsini, aidiyyəti struktur qurum və bölmələrə çatdırılmasını və Nazirliyin internet saytında yerləşdirilməsini təmin etsin.

6. Əmrin icrasına nəzarət nazir müavini Ceyhun Bayramova həvalə edilsin.

***Mikayıl CABBAROV,***  
***Azərbaycan Respublikasının Təhsil naziri***  
***12 dekabr 2014-cü il***

Azərbaycan Respublikası Təhsil nazirinin  
12 dekabr 2014-cü il tarixli 1256 nömrəli  
əmrinə əsasən təsdiq edilmişdir

## **Məktəblilərin fənn olimpiadalarının təşkili və keçirilməsi QAYDALARI**

### **1. Ümumi müddəalar**

1.1. “Məktəblilərin fənn olimpiadalarının təşkili və keçirilməsi Qaydaları” (bundan sonra – Qaydalar) Azərbaycan Respublikası Təhsil Nazirliyi haqqında Əsasnamənin 8.14-cü bəndinə uyğun olaraq hazırlanmışdır.

1.2. Fənn olimpiadalarının keçirilməsində əsas məqsəd şagirdlərdə keyfiyyətli təhsil alma, biliklərə marağı artırmaq və özünə inam hisslərini gücləndirmək, ayrı-ayrı fənlər üzrə istedadı və yaradıcı təfəkkürü ilə fərqlənənləri aşkar etməklə onların potensial imkanlarının inkişafını təşkil etmək istiqamətində tədbirlər müəyyənləşdirmək, şagirdlərin ixtisas və peşə seçiminə hazırlanması üçün zəmin yaratmaqdan ibarətdir.

1.3. Fənn olimpiadalarının yekunlarına görə, şagirdlərin əldə etdiyi nəticələr müvafiq fənn müəllimlərinin, məktəbin, yerli təhsili idarəetmə orqanının fəaliyyətinin qiymətləndirilməsində əsas meyarlardan biri kimi nəzərə alınır.

1.4. Ümumi təhsil müəssisələrinin rəhbərləri, yerli təhsili idarəetmə orqanları istedadlı şagirdlərin müəyyənləşdirilməsi və fənn olimpiadalarına hazırlığını təmin etmək üçün birbaşa məsuliyyət daşıyırlar.

1.5. Fənn olimpiadaları IX-XI sinif şagirdləri cəlb olunmaqla Azərbaycan dili və ədəbiyyat, tarix, coğrafiya, riyaziyyat, fizika, kimya, biologiya, informatika fənləri üzrə təşkil edilir. Fənn müəllimi və məktəb psixoloqunun təqdimatı əsasında xüsusi istedadı ilə seçilən aşağı sinif şagirdləri də fənn olimpiadalarına cəlb oluna bilər.

## **2. Fənn olimpiadalarının təşkili**

2.1. Fənn olimpiadalarının şəffaf, obyektiv və tələb olunan səviyyədə keçirilməsini təmin etmək məqsədilə Yerli və Respublika təşkilat komitələri və hər fənni əhatə etməklə apelyasiya komissiyaları yaradılır.

2.2. Təşkilat komitələri fənn olimpiadalarının bu Əsasnaməyə uyğun qaydada keçirilməsi üçün lazımi tədbirlər görürlər.

2.3. Yerli təşkilat komitəsi rayon (şəhər) təhsil şöbəsi (idarəsi) müdirinin, Respublika təşkilat komitəsi isə Azərbaycan Respublikasının təhsil nazirinin əmri ilə yaradılır.

2.4. Yerli təşkilat komitəsinin tərkibi rayon (şəhər) təhsil şöbəsinin (idarəsinin) müvafiq məsul işçiləri, qabaqcıl müəllimlər, valideynlərin nümayəndələrindən ibarət olur. Yerli təşkilat komitəsi 5 (beş) nəfərdən ibarət olmaqla yaradılır. Yerli təşkilat komitəsinin sədri rayon (şəhər) təhsil şöbəsinin (idarəsinin) müdiri olur.

2.5. Yerli təşkilat komitələrinin vəzifələri:

2.5.1. İstedadlı şagirdlərin aşkara çıxarılması və onlarla fənn olimpiadalarına hazırlıq işlərinin aparılmasını, rayon (şəhər) səviyyəsində dərs ili boyu müxtəlif müsabiqələr, test yoxlamaları və digər formalarda bilik yarışları keçirilməsini təmin edir;

2.5.2. Olimpiadaların rayon (şəhər) mərhələsinin keçirilməsi məqsədilə iştirakçılar üçün sual kitabçalarını və cavab kartlarını müsabiqə günü təmin edir;

2.5.3. Rayon (şəhər) mərhələsinin cavab kartlarını yoxlanılması üçün Respublika təşkilat komitəsinə göndərir.

2.6. Respublika təşkilat komitəsi Təhsil Nazirliyinin məsul əməkdaşlarından ibarət tərkibdə formalaşır. Respublika təşkilat komitəsinin səlahiyyət müddəti 1 (bir) il, tərkibi isə 7 (yeddi) nəfərdən ibarət olmaqla Azərbaycan Respublikası təhsil nazirinin əmri ilə yaradılır. Respublika təşkilat komitəsinə təhsil nazirinin müavini sədrlik edir.

2.7. Olimpiadaların yekun nəticələrindən narazı qalmış şagirdlərin yazı işlərinə yenidən baxılması üçün Respublika təşkilat komitəsi tərəfindən hər fənn üzrə 3 (üç) nəfər mütəxəssisdən ibarət apelyasiya komissiyası yaradılır.

2.8. Respublika təşkilat komitəsinin vəzifələri:

2.8.1. Rayon (şəhər) və respublika mərhələlərinin suallarının tərtib edilməsi, şagirdlərin yazı işlərinin yoxlanılması məqsədi ilə fənlər üzrə respublika münisflər heyətlərinin tərkibinə namizədləri müəyyənləşdirib təhsil nazirinə təqdim edir və onların səmərəli fəaliyyəti üçün lazımi şərait yaradır. Münisflər

heyətinin səlahiyyət müddəti 1 (bir) il, tərkibi isə hər bir fənn üzrə 5 (beş) nəfərdən ibarət olmaqla təhsil nazirinin əmri ilə təsdiqlənir;

2.8.2. Olimpiadanın respublika mərhələsində hazırlıq işlərinin vaxtında başa çatdırılmasını təmin edir, olimpiadaların şəffaf, ədalətli və aşkarlıq prinsiplərinə uyğun keçirilməsi üçün tədbirlər görür;

2.8.3. Olimpiada qaliblərinin rəğbətləndirilməsi üçün təkliflər hazırlayır və mükafatlandırılmasını təşkil edir;

2.8.4. Olimpiadaların son nəticələrini ümumiləşdirərək təhlil edir və yekun hesabatı hazırlayır;

2.8.5. Fənn olimpiadalarının təşkili və keçirilməsi prosedurunun təkmilləşdirilməsi barədə təkliflər verir.

### **3. Fənn olimpiadalarının keçirilməsi**

3.0.1. Ümumi təhsil müəssisələrində fənn olimpiadalarına hazırlıq dərsləri ərzində aparılır, müntəzəm olaraq sınaq yoxlamaları, monitorinqlər keçirilir və bu proseslərdə fənn olimpiadalarının rayon (şəhər) mərhələsi iştirakçıların tərkibi müəyyənləşdirilir.

3.0.2. Fənn olimpiadaları 2 (iki) mərhələdən – rayon (şəhər) və respublika mərhələlərindən ibarətdir. Respublika mərhələsi öz növbəsində iki turdan – yarımfinal və finaldan ibarət olmaqla keçirilir.

#### **3.1. Fənn olimpiadalarının rayon (şəhər) mərhələsi**

3.1.1. Fənn olimpiadalarının rayon (şəhər) mərhələsi yanvar ayının 2-ci bazar günü təşkil edilir.

3.1.2. Bu mərhələdə suallar siniflər üzrə respublika fənn olimpiadalarının münisiflər heyəti tərəfindən məktəb proqramına uyğun qapalı test formatında hazırlanır.

3.1.3. Rayon (şəhər) mərhələsində şagirdlərə fənlərdən və siniflərdən asılı olaraq 30-50 sual təqdim edilir və imtahan üçün 3 (üç) saat vaxt ayrılır.

3.1.4 Təhsil Nazirliyi tabeliyindəki lisey və gimnaziyaların, özəl məktəblərin şagirdləri rayon (şəhər) mərhələsində ərazi üzrə yerləşdikləri rayonda (şəhərdə) iştirak edirlər.

3.1.5. Ümumi təhsil müəssisələrinin, Təhsil Nazirliyinin tabeliyindəki lisey və gimnaziyaların, özəl məktəblərin direktorları rayon (şəhər) mərhələsində iştirak edəcək şagirdlərin adlı siyahısını dekabr ayının 20-dək müvafiq Yerli təşkilat komitəsinə təqdim edirlər. Bu turda iştirakçıların sayına məhdudiyyət qoyulmur.

3.1.6. Yerli təşkilat komitəsi rayon (şəhər) turunun cavab kartlarını ən gec imtahan keçirilən tarixdən 1 (bir) gün sonra Respublika Təşkilat Komitəsinə çatdırılmasını təmin edir.

### **3.2. Fənn olimpiadalarının respublika mərhələsinin yarımfinal turu**

3.2.1. Respublika mərhələsinin yarımfinal turu mart ayının 2-ci bazar günü Respublika Təşkilat Komitəsi tərəfindən müəyyən edilmiş mərkəzlərdə keçirilir. Bu mərkəzlər Respublikanın müxtəlif bölgələrində yerləşən ümumi təhsil müəssisələrindən seçilir.

3.2.2. Respublika mərhələsinin yarımfinal turunda iştirak edəcək şagirdlər rayon (şəhər) mərhələsinin nəticələrinə əsasən Respublika təşkilat komitəsinin qərarı ilə müəyyənləşdirilir. Rayon (şəhər) mərhələlərində hər bir fəndən sualların ən azı 60%-nə düzgün cavab vermək şərtlə hər sinif üzrə ilk 3 yeri tutmuş şagird yarımfinal mərhələsində iştirak etmək hüququ əldə edir. Gəncə və Sumqayıt şəhərlərindən hər sinif üzrə ilk 5 yeri tutmuş şagird, Bakı şəhərindən isə hər sinif üzrə ilk 20 yeri tutmuş şagird yarımfinal mərhələsində iştirak etmək hüququ əldə edir.

3.2.3. Respublika təşkilat komitəsi olimpiadaların rayon (şəhər) mərhələsinin nəticələrinə əsasən olimpiadaların respublika mərhələsinin yarımfinal turunda iştirak edəcək şagirdlərin adlı siyahısını rayon (şəhər) təhsil şöbələrinə (idarələrinə) bir ay əvvəl bildirir.

3.2.4. Bu mərhələnin yarımfinal turunda suallar siniflər üzrə respublika fənn olimpiadalarının münisiflər heyəti tərəfindən məktəb proqramına uyğun və nisbətən çətinləşdirilmiş qapalı test formatında hazırlanır.

3.2.5. Respublika mərhələsinin yarımfinal turunda şagirdlərə fənlərdən asılı olaraq 30-40 sual təqdim edilir və cavablandırma üçün 3 (üç) saat vaxt ayrılır.



3.2.6. Respublika təşkilat komitəsi yarımfinal turunun cavab kartlarının oxunmasını, qiymətləndirilməsini və nəticələrin Yerli təşkilat komitələrinə çatdırılmasını təmin edir.

### **3.3. Fənn olimpiadalarının respublika mərhələsinin final turu**

3.3.1. Respublika mərhələsinin final turu aprel ayının 2-ci bazar günü keçirilir.

3.3.2. Respublika mərhələsinin final turu iki yaş qrupu üzrə keçirilir:

- Aşağı yaş qrupu ( IX siniflər və aşağı)

- Yuxarı yaş qrupu (X-XI siniflər)

3.3.3. Respublika mərhələsinin final turunda iştirak edəcək şagirdlər yarımfinal turunun nəticələrinə əsasən Respublika təşkilat komitəsinin qərarı ilə müəyyənləşdirilir. Respublika mərhələsinin yarımfinal turunda hər fəndən sualların ən azı 70%-nə düzgün cavab vermək şərtilə ən yüksək nəticə göstərmiş aşağı yaş qrupu üzrə 20 şagird, yuxarı yaş qrupu üzrə isə 40 şagird final mərhələsində iştirak etmək hüququ əldə edir.

3.3.4. Final turunun sualları fənn olimpiadalarının münəvver heyəti tərəfindən yaş qruplarına uyğun olmaqla açıq tipli suallar formatında hazırlanır.

3.3.5. Respublika mərhələsinin final turunda şagirdlərə fənlərdən və yaş qruplarının xüsusiyyətləri nəzərə alınmaqla 5-7 sual, Azərbaycan dili və ədəbiyyatı fənni üzrə 3 sual Azərbaycan dili, 2 sual isə ədəbiyyatdan olmaqla ümumilikdə 5 sual, tarixi fənni üzrə isə 3 sual Azərbaycan tarixi, 2 sual isə ümumi tarixdən olmaqla ümumilikdə 5 sual təqdim edilir və imtahan üçün 3 (üç)-4 (dörd) saat vaxt ayrılır.

3.3.6. Respublika mərhələsinin final turunun nəticələri imtahan başa çatdıqdan 10 gün sonra elan edilir.

3.3.7 Nəticələr məlum olduqdan sonra apelyasiya günü müəyyən edilir. Nəticələrindən narazı qalmış iştirakçılar öz işlərinə yenidən baxılması üçün Respublika təşkilat komitəsinə müraciət edirlər. Belə müraciətlərə apelyasiya komissiyası yenidən baxır və onların verdiyi yekun rəy Respublika təşkilat komitəsi tərəfindən nəzərə alınır.

#### **4. Olimpiada iştirakçılarının mükafatlandırılması**

4.1. Olimpiadaların respublika mərhələsinin yarımfinal turunun qalibləri Yerli təşkilat komitələri, final turunun qalibləri isə Respublika təşkilat komitəsi tərəfindən diplom, tərifnamə, qiymətli hədiyyələr və pul mükafatı ilə təltif olunurlar.

4.2. Fənn olimpiadalarının təşkili və keçirilməsində iştirak edən münisflər heyəti üzvləri, habelə qalib şagirdlərin müəllimləri müvafiq qaydada mükafatlandırılırlar.

#### **5. Fənn olimpiadalarının maliyyələşdirilməsi**

5.1. Fənn olimpiadalarının rayon (şəhər) mərhələsi, habelə respublika mərhələsi iştirakçılarının ezamiyyə xərcləri yerli təhsil orqanları tərəfindən ödənilir.

5.2. Olimpiadaların respublika mərhələsinin təşkili, keçirilməsi, qaliblərin təltif olunması, münisflər heyəti üzvlərinin, müəllimlərin mükafatlandırılması ilə əlaqədar bütün xərclər Təhsil Nazirliyi tərəfindən ödənilir.

### İstifadə və tövsiyə olunan ədəbiyyat

1. *Calallı İ.* Beynəlxalq İnformatika Olimpiadaları / Bakı: “Bakınəşr”, 2012, 336 səh.
2. *Mahmudzadə R., Əliyev A.* İnformatika üzrə olimpiada məsələlərinin həlli / Fizika, riyaziyyat və informatika jurnalı, Bakı: 2002, №1, səh.19-23.
3. *Арсак Ж.* Программирование игр и головоломок / Ж.Арсак. – М.: Наука, 1990.
4. *Ахо А.* Структуры данных и алгоритмы / А.Ахо, Дж.Хопкрофт, Дж.Ульман. – М.: Издательский дом «Вильямс», 2000.
5. *Ахо А.* Построение и анализ вычислительных алгоритмов / А.Ахо, Дж.Хопкрофт, Дж.Ульман; пер. с англ. – М.: Мир, 1979.
6. *Бентли Д.* Жемчужины творчества программистов / Д.Бентли; пер. с англ. – М.: Радио и связь, 1990.
7. *Ван Тассел Д.* Стил, разработка, эффективность, отладка и испытание программ / Д.Ван Тассел. – М.: Мир, 1985.
8. *Вирт Н.* Алгоритмы и структуры данных / Н.Вирт; пер. с англ. – М.: Мир, 1989.
9. *Джонстон Г.* Учисеь программировать / Г.Джонстон. – М.: Финансы и статистика, 1989.
10. *Долинский М.С.* Алгоритмизация и программирование на Turbo Pascal: от простых до олимпиадных задач: учебное пособие / М.С.Долинский. – СПб.: Питер Принт, 2004.
11. *Златопольский Д.М.* Программирование: типовые задачи, алгоритмы, методы / Д.М.Златопольский – М.: БИНОМ. Лаборатория знаний, 2007.
12. *Кирюхин В.М.* Методика решения задач по информатике: международные олимпиады / В.М.Кирюхин, С.М.Окулов. – М.: Бином. Лаборатория знаний, 2007.
13. *Кирюхин В.М.* Всероссийская олимпиада школьников по информатике в 2006 году / В.М. Кирюхин, М.С.Цветкова. – М.: АПК и ППРО, 2006.
14. *Кирюхин В.М.* Информатика. Международные олимпиады. Вып. 1 / В.М.Кирюхин. – М.: Просвещение, 2009.
15. *Кнут Д.* Искусство программирования для ЭВМ. Т. 1-3 / Д.Кнут. – М.; СПб.; Киев: Вильямс, 2000.
16. *Кормен Т.* Алгоритмы: построение и анализ / Т.Кормен, Ч.Лейзерсон, Р.Ривест. – М.: МЦНМО, 1999.
17. *Кристофидес Н.* Теория графов: алгоритмический подход / Н. Кристофидес. – М.: Мир, 1978.
18. *Липский В.* Комбинаторика для программистов / В.Липский. – М.: Мир, 1988.
19. *Меньшиков Ф.В.* Олимпиадные задачи по программированию / Ф.В.Меньшиков. – СПб.: Питер, 2006.
20. *Окулов С.М.* Программирование в алгоритмах / С.М.Окулов. – М.: БИНОМ. Лаборатория знаний, 2013.
21. *Окулов С.М.* Основы программирования / С.М.Окулов. – М.: БИНОМ. Лаборатория знаний, 2012.

22. *Окулов С.М.* Информатика: развитие интеллекта школьников / С.М.Окулов. – М.: Бином. Лаборатория знаний, 2005.
23. *Окулов С.М.* Дискретная математика: теория и практика решения задач по информатике / С.М.Окулов. – М.: Бином. Лаборатория знаний, 2008.
24. *Рейгольд Э.* Комбинаторные алгоритмы: теория и практика / Э.Рейгольд, Ю.Нивергельт, Н.Део. – М.: Мир, 1980.
25. *Скиена С.С.* Олимпиадные задачи по программированию: руководство по подготовке к соревнованиям / С.С.Скиена, М.А.Ревилла. – М.: Кудиц-образ, 2005.
26. *Уэзерелл Ч.* Этюды для программистов / Ч.Уэзерелл. – М.:Мир, 1982.
27. *Шень А.* Программирование: теоремы и задачи / А.Шень. – М.: МЦНМО, 1995.

# BAŞLIQLAR

<b>Ön söz</b> .....	3
<b>I BÖLÜM. Məktəblilər arasında informatika olimpiadalarının tarixi</b>	
1.1. İnformatika ümumtəhsil məktəblərində .....	5
1.2. Respublika İnformatika Olimpiadaları .....	7
1.3. Beynəlxalq İnformatika Olimpiadaları .....	18
<b>II BÖLÜM. Respublika İnformatika Olimpiadalarının təşkili və keçirilmə qaydaları</b>	
2.1. Olimpiadaların keçirilmə mərhələləri .....	20
2.2. Olimpiadanın rəhbər orqanları .....	21
2.3. Olimpiada iştirakçılarının tərkibi .....	23
2.4. Olimpiadanın keçirilmə qaydaları və qalıbların müəyyənləşdirilməsi .....	24
<b>III BÖLÜM. Olimpiada məsələlərinin məzmunu və olimpiadaya hazırlıq üçün metodik tövsiyələr</b>	
3.1. Olimpiada məsələlərinə qoyulan tələblər .....	27
3.2. Olimpiada məsələlərinin müasir məzmunu .....	29
3.3. Olimpiada məsələlərinin həlli metodikası .....	30
<b>IV BÖLÜM. Olimpiadanın məktəb turuna hazırlıq üçün məsələlər</b>	
4.1. Məsələlər .....	33
4.2. Həllər .....	46
<b>V BÖLÜM. Olimpiadanın ikinci (rayon, şəhər) turunun məsələləri</b>	
5.1. Məsələlərin şərtləri .....	103
5.2. Həll nümunələri .....	133
<b>VI BÖLÜM. Olimpiadanın yekun (respublika) turunun məsələləri</b>	
6.1. Məsələlərin şərtləri .....	145
6.2. Həll nümunələri .....	193
6.3. Həllərin testlənməsi və qiymətləndirilməsi sistemi .....	225
<b>VII BÖLÜM. Əlavələr</b>	
7.1. Respublika İnformatika Olimpiadalarının qalıbları (1989-2014) .....	227
7.2. Sənədlər .....	240
<b>İstifadə və tövsiyə olunan ədəbiyyat</b> .....	251



**Ramin Əlinəzım oğlu Məhmudzadə**

e-mail:  
rmahmudzade@gmail.com

31 avqust 1935-ci ildə Bakı şəhərində, görkəmli ədəbiyyatşünas alim Əli Nazimin ailəsində doğulub. İbtidai təhsilini Ağstafa və Gəncədə, orta təhsilini isə Ukraynanın İvano-Frankovsk şəhərində alıb.

1953-cü ildə Dnepropetrovsk Dağ-Mədən İnstitutunun fizika fakültəsinə qəbul olunub. 1955-ci ildə həmin ali məktəbdə təhsilini yarımçıq qoyaraq Azərbaycana qayıdıb və 1956-cı ildə Azərbaycan Dövlət Universitetinin mexanika-riyaziyyat fakültəsinə daxil olub. 1958-ci ildə göndərişlə təhsilini Leningrad Dövlət Universitetində davam etdirib. 1961-ci ildə həmin təhsil ocağını uğurla başa vurub.

1961-ci ildən 1968-ci ilədək Azərbaycan Dövlət Universitetinin mexanika-riyaziyyat fakültəsində assistent işləyib. 1968-1973-cü illərdə Azərbaycan EA Nəzəri Kimya Problemləri İnstitutunda şöbə müdiri olub. 1973-cü ildə Bakı Dövlət Universitetinə qayıdaraq tətbiqi riyaziyyat və kibernetika fakültəsində dərs deyib. 1981-ci ildə müdafiə edərək fizika-riyaziyyat elmləri namizədi alimlik dərəcəsi alıb. Baş müəllim, dosent, kafedra müdiri vəzifələrində işləyib.

1970-ci illərin əvvəlində məktəblilər üçün ilk proqramlaşdırma sinifləri yaradıb (Bakı şəhəri 164 və 134 saylı orta məktəblərdə). 1980-ci illərdə indiki Bakı Dövlət Universitetində “Gənc proqramçılar məktəbi” yaradıb və 5 il həmin məktəbin işinə rəhbərlik edib. Bakı “Bilik” Cəmiyyətinin nəzdində “İstehsalatda riyazi üsullar” ictimai universitetinin rektoru olub (1985-1990).

1982-ci ildə tələbələr arasında proqramlaşdırma üzrə keçirilən ilk Ümumittifaq olimpiadasında Azərbaycan komandasına rəhbərlik edib. 1989-cu ildə məktəblərdə kompüterlərin tətbiqi ilə bağlı Ümumittifaq pilot layihəsinin Azərbaycan üzrə rəhbəri olub. Azərbaycan komandası 1988 və 1989-cu illərin Ümumittifaq informatika olimpiadalarında onun başçılığı altında iştirak edib. 1990-cı ildən məktəblilər arasında informatika üzrə respublika olimpiadalarında münisiflər heyətinin sədridir. 1993-cü ildən Beynəlxalq İnformatika Olimpiadalarında Azərbaycan komandasının rəhbəridir. 2003-cü ildən Proqramlaşdırma üzrə Beynəlxalq Tələbə Olimpiadasının (ACM/ICPC) regional 1/4 finalına başçılıq edir. 2007-ci ildən tələbələr arasında proqramlaşdırma üzrə keçirilən olimpiadaların münisiflər heyətinin sədridir.

1999-cu ildən Təhsil Nazirliyi Elmi-Metodik Şurasının “İnformatika və kompüter texnologiyası” bölməsinin sədridir. Ümumtəhsil məktəbləri üçün “İnformatika” dərsliklərinin müəlliflərindəndir.

2005-ci ildə “Şöhrət” ordeni ilə təltif edilib.





**İsmayıl Calal oğlu Sadıqov**

e-mail:

ismayil.sadigov@gmail.com

1962-ci ilin mayın 24-də Vedibasər mahalının Xalisə kəndində doğulub.

1976-1978-ci illərdə Bakı şəhər 1 nömrəli fizika-riyaziyyat təmayüllü məktəbdə orta, 1978-1983-cü illərdə isə Bakı Dövlət Universitetində mexanika-riyaziyyat ixtisası üzrə ali təhsil alıb.

1983-2006-cı illərdə Azərbaycan SSR Dövlət Maddi-Texniki Təchizat Komitəsi, Azərbaycan Dövlət Dəmir Yolu, Azərbaycan Respublikasının Rabitə Nazirliyi sistemlərində İKT sahəsi üzrə müxtəlif vəzifələrdə çalışıb. 2006-cı ildən Azərbaycan Respublikasının Təhsil Nazirliyində şöbə müdürüdür.

Avtomatlaşdırma və proqramlaşdırma üzrə bir çox layihələrin müəllifi və rəhbəri olub (“ARBAY: Mühəsibatlıq”, “ARBAY: Kadrlar”, “ARBAY: Əmək haqqı”, “ARBAY: Mallar”, “KADRLAR: Analitik-İnformasiya Sistemi”, “Bakı Telefon Şəbəkəsinin Billinq Sistemi”, “Defektoskop: Relslərdə zədələrin aşkarlanması”, “TUTSAQ: Analitik-İnformasiya Sistemi”, “TİMS – Ümumtəhsil” və b.).

Azərbaycanda ilk multimedia soraqçası – “Azərbaycan: Kim Kimdir...” kompakt-diskinin (1998), “Kompüter. Bunu öyrənməyə nə var ki!..” (2005-2011), “Beynəlxalq İnformatika Olimpiadaları” (2012) kitablarının, “Mətnlərin mürəkkəbliyi və onun qiymətləndirilməsi yolları” monoqrafiyasının müəllifidir.

Ümumtəhsil məktəblərinin 1-11-ci sinifləri üçün “İnformatika” dərsliklərinin (2006-2014) müəlliflərindəndir.

Beynəlxalq İnformatika Olimpiadalarında (2007-2014) Azərbaycan komandasının rəhbərlərindən olub.

Türkiyə Bilişim Derneğinin “Türk Dünyası Ortak Bilişim Terimleri Çalışma Grubu”nda Azərbaycanı təmsil edir.

“Rabitə ustası” fəxri adına (2008) və “Tərəqqi” medalına (2009) layiq görülüb.





**Akif Fərrux oğlu Əliyev**

e-mail:  
akif\_aliev@mail.ru

20 oktyabr 1955-ci ildə Beyləqan rayonunun Əlinəzərli kəndində doğulub.

Orta təhsilini Beyləqan rayonunda alıb. 1972-1977-ci illərdə Bakı Dövlət Universitetində mexanika-riyaziyyat ixtisası üzrə təhsil almışdır.

Orta məktəbdə müəllim, Kibernetika İnstitutunda proqramçı-mühəndis vəzifələrində işləmişdir. 1987-ci ildən Təhsil Nazirliyi sistemində metodist, Təhsil Problemləri İnstitutunda baş metodist və İnformatlaşdırma şöbəsinin müdiri vəzifəsində işləmişdir. 2004-2014-cü illərdə Təhsil Nazirliyində baş məsləhətçi vəzifəsində çalışmışdır.

Azərbaycan Respublikasında Təhsil İslahatı layihəsi üzrə İnformatika kurikulumunu hazırlayan işçi qrupunun üzvü olmuşdur. Təhsil Nazirliyinin nəzdindəki Elmi-Metodik Şuranın İnformatika bölməsinin elmi katibi, informatika fənni üzrə respublika olimpiadası münisiflər heyətinin məsul katibidir. “Fizika, riyaziyyat və informatikanın tədrisi” jurnalının redaksiya heyətinin üzvüdür.

İnformatika üzrə bir sıra proqram, metodik vəsait və dərsliklərin müəllifidir.

Təhsil Nazirliyinin həyata keçirdiyi bir çox layihələrdə, o cümlədən ümumtəhsil məktəblərinin pedaqoji heyəti üçün “İKT-nin təhsilə tətbiqi” üzrə təlimlərdə, riyaziyyat fənn müəllimləri üçün yeni kurikulum üzrə təlimlərdə təlimçi kimi iştirak etmişdir.

