

```

/*
    Є кілька множин.
    Необхідно перевірити такі умови: чи збігаються ці множини; перетинаються чи
    не перетинаються?
    Результати подайте у вигляді таблиці (матриці).
*/

#include <iostream>
#include <string>
#include <iomanip>
#include <algorithm>

using namespace std;

int main()
{
    int pluralCount;

    cout << "Input plural amount: ";
    cin >> pluralCount;

    double** plurals = new double*[pluralCount];
    double** pluralsSorted = new double*[pluralCount];
    int* pluralsSize = new int[pluralCount];
    string **pluralCoincide = new string*[pluralCount+1];
    string **pluralIntersecting = new string*[pluralCount+1];

    for (int i = 0; i < pluralCount+1; i++)
    {
        pluralCoincide[i] = new string[pluralCount+1];
        pluralCoincide[i][0] = i;
        pluralCoincide[0][i] = i;

        pluralIntersecting[i] = new string[pluralCount+1];
        pluralIntersecting[i][0] = i;
        pluralIntersecting[0][i] = i;
    }

    pluralCoincide[0][0] = "Plurals";
    pluralIntersecting[0][0] = "Plurals";

    for (int i = 0; i < pluralCount; i++)
    {
        string plural;

        cout << "Input plural number " << i + 1 << " size: ";
        cin >> pluralsSize[i];

        plurals[i] = new double[pluralsSize[i]];
        pluralsSorted[i] = new double[pluralsSize[i]];
    }
}

```

```

    cout << "Input plural in line through a space: ";

    for (int j = 0; j < pluralsSize[i]; j++)
    {
        cin >> plurals[i][j];
        pluralsSorted[i][j] = plurals[i][j];
    }

    sort(pluralsSorted[i], pluralsSorted[i]+pluralsSize[i]);
}

cout << "Convergence of sets: " << endl;

for (int i = 0; i < pluralCount; i++)
{
    for (int j = 0; j < pluralCount; j++)
    {
        if (pluralsSize[i] == pluralsSize[j])
        {
            if (i == j)
            {
                pluralCoincide[i+1][j+1] = "true";
            }
            else
            {
                for (int x = 0; x < pluralsSize[i]; x++)
                {
                    if (pluralsSorted[i][x] != pluralsSorted[j][x])
                    {
                        pluralCoincide[i+1][j+1] = "false";
                        break;
                    }
                }

                if (pluralCoincide[i+1][j+1] != "false")
                {
                    pluralCoincide[i+1][j+1] = "true";
                }
            }
        }
        else
        {
            pluralCoincide[i+1][j+1] = "false";
        }
    }
}

for (int i = 0; i < pluralCount+1; i++)
{

```

```

        for (int j = 0; j < pluralCount+1; j++)
        {
            cout << setw(8) << pluralCoincide[i][j] << " ";
        }
        cout << endl;
    }

    cout << "Intersecting sets: " << endl;

    for (int i = 0; i < pluralCount; i++)
    {
        for (int j = 0; j < pluralCount; j++)
        {
            if (pluralCoincide[i+1][j+1] != "true")
            {
                for (int x = 0; x < pluralsSize[i]; x++)
                {
                    for (int y = 0; y < pluralsSize[j]; y++)
                    {
                        if (pluralsSorted[i][x] == pluralsSorted[j][y])
                        {
                            pluralIntersecting[i+1][j+1] = "true";
                            break;
                        }
                    }
                }
                if (pluralIntersecting[i+1][j+1] == "true")
                {
                    break;
                }
            }

            if (pluralIntersecting[i+1][j+1] != "true")
            {
                pluralIntersecting[i+1][j+1] = "false";
            }

        }
        else
        {
            pluralIntersecting[i+1][j+1] = "true";
        }
    }
}

for (int i = 0; i < pluralCount+1; i++)
{
    for (int j = 0; j < pluralCount+1; j++)
    {
        cout << setw(8) << pluralIntersecting[i][j] << " ";
    }
}

```

```

        cout << endl;
    }
}

```

```

Input plural amount: 3
Input plural number 1 size: 3
Input plural in line through a space: 1 2 3
Input plural number 2 size: 3
Input plural in line through a space: 3 2 1
Input plural number 3 size: 3
Input plural in line through a space: 2 3 1
Convergence of sets:
Plurals      @      @      ♥
    @      true     true     true
    @      true     true     true
    ♥      true     true     true
Intersecting sets:
Plurals      @      @      ♥
    @      true     true     true
    @      true     true     true
    ♥      true     true     true

```

PS: D:\Денис\_Универ\Дискретні\_Структури\Лабораторія\_1\Лабораторія\_1\_Решення\_завдань\_1\_2\_3\_4\_5\_6\_7\_8\_9\_10\_11\_12\_13\_14\_15\_16\_17\_18\_19\_20\_21\_22\_23\_24\_25\_26\_27\_28\_29\_30\_31\_32\_33\_34\_35\_36\_37\_38\_39\_40\_41\_42\_43\_44\_45\_46\_47\_48\_49\_50\_51\_52\_53\_54\_55\_56\_57\_58\_59\_60\_61\_62\_63\_64\_65\_66\_67\_68\_69\_70\_71\_72\_73\_74\_75\_76\_77\_78\_79\_80\_81\_82\_83\_84\_85\_86\_87\_88\_89\_90\_91\_92\_93\_94\_95\_96\_97\_98\_99\_100\_101\_102\_103\_104\_105\_106\_107\_108\_109\_110\_111\_112\_113\_114\_115\_116\_117\_118\_119\_120\_121\_122\_123\_124\_125\_126\_127\_128\_129\_130\_131\_132\_133\_134\_135\_136\_137\_138\_139\_140\_141\_142\_143\_144\_145\_146\_147\_148\_149\_150\_151\_152\_153\_154\_155\_156\_157\_158\_159\_160\_161\_162\_163\_164\_165\_166\_167\_168\_169\_170\_171\_172\_173\_174\_175\_176\_177\_178\_179\_180\_181\_182\_183\_184\_185\_186\_187\_188\_189\_190\_191\_192\_193\_194\_195\_196\_197\_198\_199\_200\_201\_202\_203\_204\_205\_206\_207\_208\_209\_210\_211\_212\_213\_214\_215\_216\_217\_218\_219\_220\_221\_222\_223\_224\_225\_226\_227\_228\_229\_230\_231\_232\_233\_234\_235\_236\_237\_238\_239\_240\_241\_242\_243\_244\_245\_246\_247\_248\_249\_250\_251\_252\_253\_254\_255\_256\_257\_258\_259\_260\_261\_262\_263\_264\_265\_266\_267\_268\_269\_270\_271\_272\_273\_274\_275\_276\_277\_278\_279\_280\_281\_282\_283\_284\_285\_286\_287\_288\_289\_290\_291\_292\_293\_294\_295\_296\_297\_298\_299\_300\_301\_302\_303\_304\_305\_306\_307\_308\_309\_310\_311\_312\_313\_314\_315\_316\_317\_318\_319\_320\_321\_322\_323\_324\_325\_326\_327\_328\_329\_330\_331\_332\_333\_334\_335\_336\_337\_338\_339\_340\_341\_342\_343\_344\_345\_346\_347\_348\_349\_350\_351\_352\_353\_354\_355\_356\_357\_358\_359\_360\_361\_362\_363\_364\_365\_366\_367\_368\_369\_370\_371\_372\_373\_374\_375\_376\_377\_378\_379\_380\_381\_382\_383\_384\_385\_386\_387\_388\_389\_390\_391\_392\_393\_394\_395\_396\_397\_398\_399\_400\_401\_402\_403\_404\_405\_406\_407\_408\_409\_410\_411\_412\_413\_414\_415\_416\_417\_418\_419\_420\_421\_422\_423\_424\_425\_426\_427\_428\_429\_430\_431\_432\_433\_434\_435\_436\_437\_438\_439\_440\_441\_442\_443\_444\_445\_446\_447\_448\_449\_450\_451\_452\_453\_454\_455\_456\_457\_458\_459\_460\_461\_462\_463\_464\_465\_466\_467\_468\_469\_470\_471\_472\_473\_474\_475\_476\_477\_478\_479\_480\_481\_482\_483\_484\_485\_486\_487\_488\_489\_490\_491\_492\_493\_494\_495\_496\_497\_498\_499\_500\_501\_502\_503\_504\_505\_506\_507\_508\_509\_510\_511\_512\_513\_514\_515\_516\_517\_518\_519\_520\_521\_522\_523\_524\_525\_526\_527\_528\_529\_530\_531\_532\_533\_534\_535\_536\_537\_538\_539\_540\_541\_542\_543\_544\_545\_546\_547\_548\_549\_550\_551\_552\_553\_554\_555\_556\_557\_558\_559\_560\_561\_562\_563\_564\_565\_566\_567\_568\_569\_570\_571\_572\_573\_574\_575\_576\_577\_578\_579\_580\_581\_582\_583\_584\_585\_586\_587\_588\_589\_590\_591\_592\_593\_594\_595\_596\_597\_598\_599\_600\_601\_602\_603\_604\_605\_606\_607\_608\_609\_610\_611\_612\_613\_614\_615\_616\_617\_618\_619\_620\_621\_622\_623\_624\_625\_626\_627\_628\_629\_630\_631\_632\_633\_634\_635\_636\_637\_638\_639\_640\_641\_642\_643\_644\_645\_646\_647\_648\_649\_650\_651\_652\_653\_654\_655\_656\_657\_658\_659\_660\_661\_662\_663\_664\_665\_666\_667\_668\_669\_670\_671\_672\_673\_674\_675\_676\_677\_678\_679\_680\_681\_682\_683\_684\_685\_686\_687\_688\_689\_690\_691\_692\_693\_694\_695\_696\_697\_698\_699\_700\_701\_702\_703\_704\_705\_706\_707\_708\_709\_710\_711\_712\_713\_714\_715\_716\_717\_718\_719\_720\_721\_722\_723\_724\_725\_726\_727\_728\_729\_730\_731\_732\_733\_734\_735\_736\_737\_738\_739\_740\_741\_742\_743\_744\_745\_746\_747\_748\_749\_750\_751\_752\_753\_754\_755\_756\_757\_758\_759\_760\_761\_762\_763\_764\_765\_766\_767\_768\_769\_770\_771\_772\_773\_774\_775\_776\_777\_778\_779\_780\_781\_782\_783\_784\_785\_786\_787\_788\_789\_790\_791\_792\_793\_794\_795\_796\_797\_798\_799\_800\_801\_802\_803\_804\_805\_806\_807\_808\_809\_810\_811\_812\_813\_814\_815\_816\_817\_818\_819\_820\_821\_822\_823\_824\_825\_826\_827\_828\_829\_830\_831\_832\_833\_834\_835\_836\_837\_838\_839\_840\_841\_842\_843\_844\_845\_846\_847\_848\_849\_850\_851\_852\_853\_854\_855\_856\_857\_858\_859\_860\_861\_862\_863\_864\_865\_866\_867\_868\_869\_870\_871\_872\_873\_874\_875\_876\_877\_878\_879\_880\_881\_882\_883\_884\_885\_886\_887\_888\_889\_890\_891\_892\_893\_894\_895\_896\_897\_898\_899\_900\_901\_902\_903\_904\_905\_906\_907\_908\_909\_910\_911\_912\_913\_914\_915\_916\_917\_918\_919\_920\_921\_922\_923\_924\_925\_926\_927\_928\_929\_930\_931\_932\_933\_934\_935\_936\_937\_938\_939\_940\_941\_942\_943\_944\_945\_946\_947\_948\_949\_950\_951\_952\_953\_954\_955\_956\_957\_958\_959\_960\_961\_962\_963\_964\_965\_966\_967\_968\_969\_970\_971\_972\_973\_974\_975\_976\_977\_978\_979\_980\_981\_982\_983\_984\_985\_986\_987\_988\_989\_990\_991\_992\_993\_994\_995\_996\_997\_998\_999\_1000

```

Input plural amount: 4
Input plural number 1 size: 3
Input plural in line through a space: 1 2 3
Input plural number 2 size: 2
Input plural in line through a space: 2 2
Input plural number 3 size: 4
Input plural in line through a space: 1 2 3 4
Input plural number 4 size: 4
Input plural in line through a space: 4 3 2 1
Convergence of sets:
Plurals      @      @      ♥      ♦
    @      true     false    false    false
    @      false    true     false    false
    ♥      false    false    true     true
    ♦      false    false    true     true
Intersecting sets:
Plurals      @      @      ♥      ♦
    @      true     true     true     true
    @      true     true     true     true
    ♥      true     true     true     true
    ♦      true     true     true     true

```

```
Input plural amount: 3
Input plural number 1 size: 2
Input plural in line through a space: 1 2
Input plural number 2 size: 3
Input plural in line through a space: 3 4 5
Input plural number 3 size: 1
Input plural in line through a space: 2
```

Convergence of sets:

Plurals	⊖	⊕	♥
⊖	true	false	false
⊕	false	true	false
♥	false	false	true

Intersecting sets:

Plurals	⊖	⊕	♥
⊖	true	false	true
⊕	false	true	false
♥	true	false	true