

```

#include "stdafx.h" // előre fordított header
#define _USE_MATH_DEFINES // M_PI a cmath előtt már késő
#include <iostream> // cout nincs .h
#include <iomanip> // i/o manipulativ: pl setw
#include <cmath> // első utasítása: math.h
#include <ctime> // time.h
using namespace std; // std:: helyett

int jegy,**kt; // itt az idő, hogy megismerkedjünk a kétdimenziós tömbbel!
int &mas_jegy=jegy; // a jegy változó másik neve a mas_jegy. ugyanaz a változó
(cím, típus) másik névvel!
time_t ido; // a jó öreg idő.
bool igaz; // c++ definiált alaptípusa. hány bájt hosszú ? majd kiderül.

void egysegugras_nemallapotgepes(double t, double &kimenet) // a kimenetet meg
tudom változtatni, ha referenciát kapok
{
    if (t >= 0 && t < 1) kimenet=1; else kimenet=0; // ebben hol az állapot?
}

// ezt a függvényt loptam a githubról. hf: letesztelni -2..+2-
ig 10000 db szám, eloszlást kiírni.
double gauss()
{
    double x = (double)rand() / RAND_MAX,
           y = (double)rand() / RAND_MAX,
           z = sqrt(-2 * log(x)) * cos(2 * M_PI * y);
    return z;
}

int állapot; // a gép. mindig tudjuk, hogy melyik állapot-ban van
void egysegugras_allapotgepes(double t, double &kimenet)
{
    if (fabs(t) < 1e-10) állapot = 1; // itt az ugrás. VÁLTOZÓBA kerül
    if (fabs(t - 1) < 1e-
10) állapot = 0; // vissza. azért nagyjából legyen ott idő!
    switch (állapot)
    {
    case 0:
        kimenet = 0.1*gauss(); break;
    case 1:
        kimenet = 1 + 0.1*gauss(); break;
    default:
        kimenet = 0;
        break;
    } // kimenet = állapot + 0.1*gauss(); egyszerűen
}

int _tmain(int argc, _TCHAR* argv[])
{
    int i,j;
    double t, dt, y;
    setlocale(LC_ALL, "hun"); // ékezetes kiírás a cout-
ban, hosszú ő és ű ettől nem lesz

```

```

    cout << "az igazság " << sizeof(igaz) << " bájttal hosszú a c++-
ban." << endl; // \n helyett
    // stack overflow-ról:
    // How would you create a pointer to a single bit? You can't. But you ca
n create a pointer to a byte.
    time(&ido); // *time_t volt a függvény paramétere. ez címet jelöl
    cout << "1970 január 1 óta eltelt " << ido << " másodperc" << endl;
    srand((unsigned int)ido);
    cout << "hányasod volt prog alapból? ";
    cin >> jegy; // semmi %i nem kell.
    igaz = true; // előre definiált logikai konstans
    if (jegy < 1) igaz = false; // ki is írhattam volna
    if (jegy > 5) igaz = false; // igaz=(jegy>=1 && jegy <=5);
    if (!igaz) cout << "ez nem is igaz!" << endl;
    kt = new int*[jegy]; // ennek semmi értelme a jegyből gyártani, de legal
abb 2 elemű tömböt készítünk a tömböknek
    // a new nem függvény, hanem műveleti jel. párja a delete műveleti jel
    for (i = 0; i < jegy; i++) kt[i] = new int[jegy]; // kész az n*n-
es mátrix!
    for (i = 0; i < jegy; i++) for (j=0; j<jegy; j++) kt[i][j] = rand(); //
t[i,j] itt nincs, csak .net-ben!
    for (i = 0; i < jegy; i++)
    {
        for (j=0; j<jegy; j++) cout << setw(6) << kt[i][j]; // rand_max 5 jegyű
szám 6 széles mezőben
        cout << endl; // és a soremelés
    }
    for (i = 0; i < jegy; i++) delete kt[i]; // vektorok felszabadítva
    delete kt; // és a tömb is
    mas_jegy = 5; // ezen a referencián
    cout << "szerintem a jegyed " << jegy << " volt" << endl;
    dt = 0.1;
    for (t = -0.2; t < 1.3; t += dt) // próbáljunk meg -1-
től menni ! mi lesz a 0-nál ?
    {
        egysegugras_nemallapotgepes(t, y); //y-t meg tudja változtatni!
        cout << "t=" << setw(4) << setprecision(3) << t // a precision a teljes
pontosságot jelöli
        << " kimenet=" << setw(4) << setprecision(3) << y << endl; // nem a tize
desjegyek számát!
    }
    állapot = 0; // az idők előtti állapot
    cout << scientific; // ezután már csak tudományosan íratunk ki!
    for (t = -0.2; t < 1.3; t += dt) // másold le az előző for-t
    {
        egysegugras_allapotgepes(t, y);
        cout << "t=" << t << " kimenet=" << setprecision(10) << y << endl;
    }
    cout.unsetf(ios_base::floatfield); //így kell kikapcsolni a tudományosat
    cout << "az állapot a végén: " << állapot << endl;
    cin.get(); cin.get();
    return 0;
}

```