


```
Terminál
Megnyitás  x
return p;
}

extern void kiir (BINFA_PTR elem);
extern void ratlag (BINFA_PTR elem);
extern void rszoras (BINFA_PTR elem);
extern void szabadit (BINFA_PTR elem);

int
main (int argc, char **argv)
{
    char b;
    int i, egy_e;

    BINFA_PTR gyoker = uj_elem ();
    gyoker->ertekek = '/';
    gyoker->bal_nulla = gyoker->jobb_egy = NULL;
    BINFA_PTR fa = gyoker;
    long max=0;
    while (read (0, (void *) &b, 1))
    {
        // write (1, &b, 1);

        for (i = 0; i < 8; ++i)
        {
            egy_e = b & 0x80;
            if ((egy_e >> 7) == 1)
            {
                //printf ("1");
                if (fa->jobb_egy == NULL)
                {
                    fa->jobb_egy = uj_elem ();
                    fa->jobb_egy->ertekek = 1;
                    fa->jobb_egy->bal_nulla = fa->jobb_egy->jobb_egy = NULL;
                    fa = gyoker;
                }
                else
                {
                    fa = fa->jobb_egy;
                }
            }
            else
            {
                //printf ("0");
                if (fa->bal_nulla == NULL)
                {
                    fa->bal_nulla = uj_elem ();
                    fa->bal_nulla->ertekek = 0;
                    fa->bal_nulla->bal_nulla = fa->bal_nulla->jobb_egy = NULL;
                    fa = gyoker;
                }
                else
                {
                    fa = fa->bal_nulla;
                }
            }
        }
        b <<= 1;
    }
}
```

```
krisztian@krisztian-VirtualBox: ~/Asztal/udprog/udprog-code/source/labor/ziv
.....-0(3)
melyseg=4
altag=3.000000
szoras=0.632456
krisztian@krisztian-VirtualBox:~/Asztal/udprog/udprog-code/source/labor/ziv$ echo "asd" | ./sdd
.....-1(2)
.....-1(1)
.....-0(2)
.....-0(3)
.../(0)
.....-1(3)
.....-0(4)
.....-1(2)
.....-0(3)
.....-0(1)
.....-1(3)
.....-0(2)
.....-0(3)
melyseg=4
altag=3.000000
szoras=0.632456
krisztian@krisztian-VirtualBox:~/Asztal/udprog/udprog-code/source/labor/ziv$
```

```
Terminál
Megnyitás  x
#include <stdio.h>
#include <math.h>
void kiir(double tomb[], int db)
{
    int i=0;
    for(i=0;i<db;i++){
        printf("%.8lf\n",tomb[i]);
    }
}
double tavolsag (double PR[], double PRV[], int n)
{
    int i;
    double osszeg=0;
    for (i = 0; i < n; ++i)
        osszeg += (PRV[i] - PR[i]) * (PRV[i] - PR[i]);
    return sqrt (osszeg);
};
int main(void)
{
    double L[4][4] = {
        {0.0, 0.0, 1.0 / 3.0, 0.0},
        {1.0, 1.0 / 2.0, 1.0 / 3.0, 1.0},
        {0.0, 1.0 / 2.0, 0.0, 0.0},
        {0.0, 0.0, 1.0 / 3.0, 0.0},
    };
    double PR[4] = { 0.0, 0.0, 0.0, 0.0 };
    double PRV[4] = { 1.0 / 4.0, 1.0 / 4.0, 1.0 / 4.0, 1.0 / 4.0 };
    int i, j;
    for (;;)
    {
        for(i=0;i<4;++i)
        {
            PR[i]=0.0;
            for (j=0;j<4;++j)
                PR[i] += (L[i][j]*PRV[j]);
        }
        if (tavolsag (PR, PRV, 4) < 0.00000001)
            break;
        for (i=0;i<4;++i)
            PRV[i]= PR[i];
    }
    kiir (PR, 4);
    return 0;
}
```

```
krisztian@krisztian-VirtualBox: ~/Dokumentumok/prog1 gyak
krisztian@krisztian-VirtualBox:~/Dokumentumok/prog1 gyak$ ./asd
0.09090909
0.54545454
0.27272727
0.09090909
krisztian@krisztian-VirtualBox:~/Dokumentumok/prog1 gyak$
```