

1. Условие задачи

Для динамического одномерного массива X из N ($0 < N \leq 20$) элементов составить алгоритм и программу (консольное приложение на Си) нахождения суммы ряда с заданной точностью E ($0 < E < 0.1$) для каждого из элементов X

№	Ряд	Контрольная формула
24	$\frac{x^2}{1 \cdot (2 \cdot 1 - 1)} - \frac{x^4}{2 \cdot (2 \cdot 2 - 1)} + \frac{x^6}{3 \cdot (2 \cdot 3 - 1)} - \dots \pm \frac{x^{2i}}{i(2i-1)} \mp \dots$	$2x \cdot \arctg x - 2 \ln \sqrt{1+x^2}$

2. Уточненное условие задачи

Дан вещественный динамический одномерный массив X (кол-во элементов не превышает 20-ти) и бесконечный ряд. Вычислить рекуррентную формулу в виде $\frac{x_i}{x_{i-1}}$ и с помощью нее вычислить сумму данного бесконечного ряда с заданной точностью e (вещественное число < 1).

При X=0.5 контрольная формула принимает значение:

$$2 * 0.5 * \arctg(0.5) - 2 \ln \sqrt{1 + 0.25} = 0.240 \dots$$

3. Рекуррентная формула:

Выведем формулу

$$x_i = (-1)^{i+1} \frac{x^{2i}}{i(2i-1)}, x_{i-1} = (-1)^{(i-1)+1} \frac{x^{2(i-1)}}{(i-1)(2(i-1)-1)} = (-1)^i \frac{x^{2(i-1)}}{(i-1)(2i-3)}$$

$$R = \frac{x_i}{x_{i-1}} = -x^2 * \frac{(i-1)(2i-3)}{i(2i-1)}$$

Проверим её:

$$x_1 = x^2$$

$$x_2 = x_1 * R = x^2 * (-x^2 * \frac{(2-1)(2*2-3)}{2(2*2-1)}) = -\frac{x^4}{2*(2*2-1)};$$

4. Тесты

Исходный массив X: [-0.98, -0.5, 0.1, 0.5, 0.95]

Для точности E=0.1

№	X	Sum(x)	K	F(x)	Sum(x)-F(x)
1	-0.98	0.87	4	0.85	0.0193
2	-0.5	0.24	3	0.24	0.0009
3	0.1	0.01	2	0.01	0.0000
4	0.5	0.24	3	0.24	0.0009
5	0.95	0.82	4	0.80	0.0154

Для точности $E=0.01$

№	X	Sum(x)	K	F(x)	Sum(x)-F(x)
1	-0.98	0.850	8	0.846	0.00346
2	-0.5	0.241	4	0.241	0.00012
3	0.1	0.010	3	0.010	0.00000
4	0.5	0.241	4	0.241	0.00012
5	0.95	0.797	7	0.800	0.00340

Для точности $E=0.001$

№	X	Sum(x)	K	F(x)	Sum(x)-F(x)
1	-0.98	0.8468	18	0.8464	0.000413
2	-0.5	0.2405	5	0.2405	0.000019
3	0.1	0.0100	3	0.0100	0.000000
4	0.5	0.2405	5	0.2405	0.000019
5	0.95	0.8007	14	0.8004	0.000353

Для точности $E=0.0001$

№	X	Sum(x)	K	F(x)	Sum(x)-F(x)
1	-0.98	0.84639	37	0.84643	0.0000435
2	-0.5	0.24051	6	0.24050	0.0000031
3	0.1	0.00998	3	0.00998	0.0000001
4	0.5	0.24051	6	0.24050	0.0000031
5	0.95	0.80042	24	0.80038	0.0000413

5.Код:

```
#include <stdio.h> // printf, scanf
#include <conio.h> // getch
#include <cmath> // fabs, pow, ceil\floor, log, log10, exp, sqrt
#include <stdlib.h> // randomize, rand
#include <windows.h> // SetConsoleOutputCP(1251); SetConsoleCP(1251);
#include <ctime>

//-----
const double xx[7] = {0.00001, -0.99, -1, -0.1, 0.1, 1, 0.99};

double Coeff(double x, int i) //Рекуррентное соотношение
{
    return -(pow(x,2)*(i-1)*(2*i-3)/(i*(2*i-1)));
}

double func(double x) //Контрольная формула
{
    return 2*x*atan(x)-2*log(pow(1+pow(x,2),1.0/2));
}

#pragma argsused
int main() {
    int n, i, k, z;
    double e, sl, sum, f, *x; // четыре простого типа и один – дин.массив(указатель)
    SetConsoleOutputCP(1251);
    srand(time(0)); //randomize

    printf("Введите e=? "); // приглашение – шрифт Lucida Console нужен
    scanf("%lf", &e); // ввод числа с плавающей точкой двойной точности(%lf) e
    if (e < 1e-13 || e > 0.11) {
        printf("Некорректная точность e (0..0.1] \nPress any key");
        getch(); // ожидание нажатия клавиши
        return 0; // выход из функции main
    }
    fflush(stdin); // очищаем буфер (лишние символы после ввода e)

    printf("Введите n=? "); // приглашение
    scanf("%d", &n); // ввод десятичного(%d) n
    if (n < 1 || n > 20) {
        printf("Invalid n [1..20]! \nPress any key");
        getch(); // ожидание нажатия клавиши
        return 0; // выход из функции main
    }
    fflush(stdin);

    x = new double[n]; // выделяем память для n элементов массива

    printf("Введите n=%d значений X из интервала (-1,+1):\n", n); // приглашение
    for (i = 0; i < n; i++) { // ввод x[0]...x[n-1]
        scanf("%lf", &x[i]); // типа long float (%lf)
```

```

if (fabs(x[i]) >= 1) {
    x[i] = xx[rand() % 7]; // одно из семи значений массива xx
    if (fabs(x[i]) == 1) {
        x[i] = x[i] * (rand() % 100) / 100;
        if (x[i] == 0) x[i] = e; // для некоторых вариантов x/=0
    }
    printf("Некорректное значение заменено на %15.10lf\n", x[i]);
}
}

```

```

z = ceil(fabs(log(e) / log(10.0))) + 1; //Вычисление знака после запятой

```

```

printf("e = %*. *lf\n", z + 2, z, e); // вывод e:(z+2):z
printf("N |    X    |   Sum(X)   | K|    F(X)    | |Sum(X)-F(X)|\n");
for (i = 1; i < 80; i++) printf("=");
printf("\n");
for (i = 0; i < n; i++) {

```

```

    // поиск суммы ряда
    sl=pow(x[i],2);
    sum=sl; // первое слагаемое
    k = 2;
    while ((fabs(sl) >= e) && (k != 500)) {
        sl *= Coeff(x[i],k);
        sum += sl;
        k++;
    };

```

```

    f = func(x[i]);

```

```

    printf("%2d| %17.*lf| %17.*lf| %2d| %17.*lf| %17.*lf\n",
        i + 1, z, x[i], z, sum, k, z, f, z + 2, fabs(sum - f));
}

```

```

delete[] x; // освобождение памяти для одномерного дин.массива

```

```

printf("Нажмите любую клавишу - Press any key");
getch();
return 0;
}

```