

## 1) Параметрическая математическая модель

1) Параметрическая математическая модель

$$g(t) = g_0 + g_1 t + g_2 t^2 + \dots + g_n t^n \quad - \text{полиномиальное воздействие}$$

$g_i$  - коэффициенты полинома,  $i = 0, 1, \dots, n$   
 $n$  - степень полинома  
 $n+1$  - порядок полиномиального воздействия

$$G(t) = \begin{cases} g_0(t), & t \in [t_{\text{нач}_0}, t_{\text{кон}_0}] \\ g_1(t), & t \in [t_{\text{нач}_1}, t_{\text{кон}_1}] \\ \vdots \\ g_n(t), & t \in [t_{\text{нач}_n}, t_{\text{кон}_n}] \end{cases}$$

$G(t)$  - результирующая функция  
 $t_{\text{нач}_i}$  - начало  $i$ -ого интервала  
 $t_{\text{кон}_i}$  - конец  $i$ -ого интервала

2) Алгоритм компьютерной модели

## 2) Алгоритм компьютерной модели

1. Считать количество интервалов (intervals)
2. Считать время начала и время конца  $i$ -ого интервала и его степень (массив begin\_arr - для начал интервалов и end\_arr для концов, degree\_arr - массив степеней полиномов)
3. Считать коэффициенты полинома для  $i$ -ого интервала (массив coef\_arr)
4. Посчитать значения функций на каждом интервале (g\_values)

## 3) Компьютерная модель в среде MATLAB

```

function [] = new()
    interval_number = input('Введите количество интервалов: ');

    begin_arr = zeros(1, interval_number);
    end_arr = zeros(1, interval_number);
    degree_arr = zeros(1, interval_number);
    coef_arr = cell(1, interval_number);

    for i = 1:interval_number
        disp(['Введите данные для интервала ', num2str(i)]);

        begin_arr(i) = input(['Начало временного интервала ', num2str(i), ': ']);
        end_arr(i) = input(['Конец временного интервала ', num2str(i), ': ']);
        degree_arr(i) = input(['Степень полинома для интервала ', num2str(i), ': ']);

        coef_arr{i} = zeros(1, degree_arr(i) + 1);
        for j = 1:degree_arr(i) + 1
            coef_arr{i}(j) = input(['Коэффициент g', num2str(j-1), ' для интервала ', num2str(i), ': ']);
        end
    end

    t = linspace(min(begin_arr), max(end_arr), 1000);
    g_values = zeros(size(t));

    for k = 1:interval_number
        mask = (t >= begin_arr(k)) & (t <= end_arr(k));
        g_values(mask) = compute_poly(t(mask), begin_arr(k), degree_arr(k), coef_arr{k});
    end

    plot(t, g_values);
    xlabel('Время t');
    ylabel('Значение g(t)');
    title('График функции g(t)');
    grid on;
end

function result = compute_poly(t, begin_point, degree, coefficients)
    result = coefficients(1);
    for i = 2:degree+1
        result = result + coefficients(i) * power((t - begin_point), (i-1));
    end
end

```

Пример входных данных:

Введите количество интервалов: 3  
 Введите данные для интервала 1  
 Начало временного интервала 1: 0  
 Конец временного интервала 1: 3  
 Степень полинома для интервала 1: 1  
 Коэффициент  $g_0$  для интервала 1: 2  
 Коэффициент  $g_1$  для интервала 1: 3  
 Введите данные для интервала 2  
 Начало временного интервала 2: 3  
 Конец временного интервала 2: 6  
 Степень полинома для интервала 2: 3  
 Коэффициент  $g_0$  для интервала 2: 1  
 Коэффициент  $g_1$  для интервала 2: 2  
 Коэффициент  $g_2$  для интервала 2: 2  
 Коэффициент  $g_3$  для интервала 2: 1  
 Введите данные для интервала 3  
 Начало временного интервала 3: 6  
 Конец временного интервала 3: 8  
 Степень полинома для интервала 3: 2  
 Коэффициент  $g_0$  для интервала 3: 0  
 Коэффициент  $g_1$  для интервала 3: 0  
 Коэффициент  $g_2$  для интервала 3: 1

График на выходе

