Calculați $\int_1^2 \sqrt{3t-t^2-2}\sin(t)dt$, cu o precizie de $\varepsilon = 1e-10$, folosind o cuadratură de tip Gauss-Cebîșev #2.

Soluție;

Facem schimbarea de variabilă $x = 2t-3 \to t = \frac{x+3}{2}, dt = \frac{1}{2}dx$:

$$\int_1^2 \sqrt{3t-t^2-2}\sin(t)dt = \frac{1}{2}\int_{-1}^1 \sqrt{3\frac{x+3}{2} - \frac{(x+3)^2}{4} - 2}\cdot \sin\left(\frac{x+3}{2}\right)dx$$
$$= \frac{1}{2}\int_{-1}^1 \sqrt{\frac{6x+18-x^2-6x-9-8}{4}}\cdot \sin\left(\frac{x+3}{2}\right)dx = \frac{1}{4}\int_{-1}^1 \sqrt{1-x^2}\cdot \sin\left(\frac{x+3}{2}\right)dx$$
$$= \frac{1}{4}\int_{-1}^1 w(x)f(x)dx,$$

unde am notat $w(x) = \sqrt{1-x^2}$ ponderea corespunzătoare nodurilor Cebîșev de speța a doua, și $f(x) = \sin\left(\frac{x+3}{2}\right)$.

```
format("long");
f = @(x) sin((x+3)/2);
gauss_quad_int = GaussQuadIntPrecision(f, @Gauss_Cebisev2, 10, 1e-10) / 4
```

```
Precision reached after 6 epochs
gauss_quad_int =
   0.379601109535519
```

```
real =integral(@(t) sqrt(3*t-t.*t-2).*sin(t),1,2)
```

```
real =
   0.379601109535519
```

```
function I=GaussQuadIntPrecision(f, nodes_generator, max_iters, tol)
    I = GaussQuadInt(f, nodes_generator, 1);
    I0 = I;
    for i=2:max_iters
        I = GaussQuadInt(f, nodes_generator, i);
        if(abs(I-I0)<tol)
            fprintf("Precision reached after %i epochs", i);
            return
        end
        I0=I;
    end
    fprintf("Required precision was not reached. Error is %f", abs(I-I0));
end
```

```
function I=GaussQuadInt(f, nodes_generator, n)
    [g_nodes, g_coeff] = nodes_generator(n);
    I = vquad(g_nodes, g_coeff, f);
end
```

```
function I = vquad(g_nodes, g_coeff, f)
    I = g_coeff * f(g_nodes);
end
```

```
% Gauss-Cebîșev de speța a II-a.
function [g_nodes, g_coeff] = Gauss_Cebisev2(n)
    beta = [pi / 2, 1 / 4 * ones(1, n - 1)];
    alpha = zeros(n, 1);
    [g_nodes, g_coeff] = GaussQuad(alpha, beta);
end
```

```
function [q_nodes, q_coeff] = GaussQuad(alpha, beta)

    n = length(alpha);
    rb = sqrt(beta(2 : n));
    J = diag(alpha) + diag(rb, -1) + diag(rb, 1);
    [v, d] = eig(J);
    q_nodes = diag(d);
    q_coeff = beta(1) * v(1, :).^2;

end
```