```cpp
/*
A program to solve initial value problem using Runge-Kutta method.
This program is created by Mohammad Sazzad Hossain.
*/


    # include <iostream>
    # include <math.h>

        using namespace std;

        double q = 0.5, b = 0.9, w = 2/3, tau = 0.1;

        double g1_funct (double y1, double y2, double t){return (y2);}
        double g2_funct (double y1, double y2, double t){return (-q * y2 - sin(y1) + b * cos (w * t));}
        double k1_funct (double y1, double y2, double t);
        double k2_funct (double y1, double y2, double t);

    int main (){
        double pi, t = 0, y1 = 0, y2 = 2, y = 0;
        FILE *rungekutta;

        rungekutta = fopen ("\\rungekutta.txt", "w");

        pi = 4 * atan (1);

        fprintf (rungekutta,"y1 \t y2 \n");
    do {
        y1 += (k1_funct(y1, y2, t) / 6);
        y2 += (k2_funct(y1, y2, t) / 6);
        t += tau;

        if (abs (y1) > pi)
        y1 -= 2 * pi * abs (y1) / y1;

        cout << t << "\t" << y1 << "\t" << y2 << endl;
        fprintf (rungekutta,"%g \t %g \n", y1, y2);
    }while (t <= 20 * pi);

        fclose (rungekutta);
    return 0;
    }

        double k1_funct (double y1, double y2, double t){
```

```
    double k[4];

    k[0] = tau * g1_funct (y1, y2, t);
    k[1] = tau * g1_funct (y1 + k[0] / 2, y2 + k[0] / 2, t + tau / 2);
    k[2] = tau * g1_funct (y1 + k[1] / 2, y2 + k[1] / 2, t + tau / 2);
    k[3] = tau * g1_funct (y1 + k[2], y2 + k[2], t + tau);

return (k[0] + 2 * (k[1] + k[2]) + k[3]);
}

    double k2_funct (double y1, double y2, double t){
    double k[4];

    k[0] = tau * g2_funct (y1, y2, t);
    k[1] = tau * g2_funct (y1 + k[0] / 2, y2 + k[0] / 2, t + tau / 2);
    k[2] = tau * g2_funct (y1 + k[1] / 2, y2 + k[1] / 2, t + tau / 2);
    k[3] = tau * g2_funct (y1 + k[2], y2 + k[2], t + tau);

return (k[0] + 2 * (k[1] + k[2]) + k[3]);
}
```