```c
// C program for Picard's iterative method

#include <math.h>
#include <stdio.h>
#define Y1(x) (1 + (x) + pow(x, 2) / 2)
#define Y2(x) (1 + (x) + pow(x, 2) / 2 + pow(x, 3) / 3 + pow(x, 4) / 8)
#define Y3(x) (1 + (x) + pow(x, 2) / 2 + pow(x, 3) / 3 + pow(x, 4) / 8 + pow(x, 5) / 15 + pow(x, 6) / 48)

int main()
{
    double start_value = 0, end_value = 3,
           allowed_error = 0.4, temp;
    double y1[30], y2[30], y3[30];
    int count;

    for (temp = start_value, count = 0;
         temp <= end_value;
         temp = temp + allowed_error, count++) {

        y1[count] = Y1(temp);
        y2[count] = Y2(temp);
        y3[count] = Y3(temp);
    }

    printf("\nX\n");
    for (temp = start_value;
         temp <= end_value;
         temp = temp + allowed_error) {

        printf("%.4lf ", temp);
    }

    printf("\n\nY(1)\n");
    for (temp = start_value, count = 0;
         temp <= end_value;
         temp = temp + allowed_error, count++) {

        printf("%.4lf ", y1[count]);
    }

    printf("\n\nY(2)\n");
    for (temp = start_value, count = 0;
         temp <= end_value;
         temp = temp + allowed_error, count++) {

        printf("%.4lf ", y2[count]);
    }

    printf("\n\nY(3)\n");
    for (temp = start_value, count = 0;
         temp <= end_value;
         temp = temp + allowed_error, count++) {

        printf("%.4lf ", y3[count]);
    }
    return 0;
}
```