REPORT : Afifah Maya Iknaningrum (1715011053)

Problem :

Make a program code of Euler Method and classical Runge-Kutta for

$$\begin{cases} y' = 5(2-y)y \quad (0 \le t \le 1) \\ y(0) = 0.04 \end{cases}$$

with C language and draw a graph to compare the numerical solution and exact solution for $N = 4, 8, 16, 32, 64, 128$ and $h = 1/N$. The exact solution is

$$y(t) = \frac{2}{1 + 49e^{-10t}}$$

C Code :

```c
#include<stdio.h>
#include<stdlib.h>
#include<math.h>

int main(int argc, char *argv[])
{
    int i;
    double y, yy, h, t, er, k1, k2, k3, k4, log_1, log_2;

    FILE *euler;
    FILE *rk;
    FILE *er_log;
    char filename1[30];
    char filename2[30];

    er_log = fopen("log_error.txt","w");

    for(int dev=128;dev>=4;dev=dev/2)
    {
        h = 1./dev;

        sprintf(filename1, "euler_%.3d.txt",dev);
        sprintf(filename2, "rk_%.3d.txt",dev);

        euler = fopen(filename1,"w");
        rk = fopen(filename2, "w");

        y = 0.04; //initial condition
        log_1 = 0.;
        log_2 = 0.;

        fprintf(euler,"%.3f %.3f %.3f %.3f \n", 0., y, y, 0.);
        fprintf(rk,"%.3f %.3f %.3f %.3f \n", 0., y, y, 0.);

        //Using Euler
        for(i=1;i<=dev;i++)
        {
            t=i*h;
            y = 5*h*(2-y)*y + y;
            yy = 2./(1+49*exp(-10*t));
            er = fabs(yy-y);
            fprintf(euler,"%.3f %.3f %.3f %.3f \n", t, y, yy, er);
            if(er>=log_1) log_1=er;
        }

        //Using RK
        y = 0.04; //initial condition
        for(i=1;i<=dev;i++)
        {
```

```
50              t=i*h;
51              k1 = 5*(2-y)*y;
52              k2 = 5*(2-(y+(k1*h/2.)))*(y+(k1*h/2.));
53              k3 = 5*(2-(y+(k2*h/2.)))*(y+(k2*h/2.));
54              k4 = 5*(2-(y+(k3*h)))*(y+(k3*h));
55              y = h*(k1 + 2*k2 + 2*k3 + k4)/6. + y;
56              yy = 2./(1+49*exp(-10*t));
57              er = fabs(yy-y);
58              fprintf(rk,"%.3f %.3f %.3f %.3f \n", t, y, yy, er);
59              if(er>=log_2) log_2=er;
60          }
61
62          fprintf(er_log, "%.3f %.3f %.3f \n", h, log_1, log_2);
63
64          //plot
65          FILE *pipe = popen("gnuplot", "w");
66          fprintf(pipe, "reset \n");
67          fprintf(pipe, "set terminal png \n");
68          fprintf(pipe, "set output 'plot%.3d.png'\n", dev);
69          fprintf(pipe, "set title 'Euler and Runge-Kutta for devider %.3d' \n", dev);
70          fprintf(pipe, "set xrange [0:1] \n");
71          fprintf(pipe, "set xlabel 't' \n");
72          fprintf(pipe, "set yrange [0:4] \n");
73          fprintf(pipe, "set ylabel 'y' \n");
74          fprintf(pipe, "plot '%s' using 1:2 w linespoint title 'Euler y numeric',
                '%s' using 1:2 w linespoint title 'RK y numeric', '%s' using 1:3 w
                linespoint title 'Exact y' , '%s' using 1:4 w linespoint title 'Euler
                differences' , '%s' using 1:4 w linespoint title 'RK differences'\n",
                filename1, filename2, filename1, filename1, filename2);
75          fclose(pipe);
76      }
77
78      FILE *pipe1 = popen("gnuplot","w");
79      fprintf(pipe1, "reset \n");
80      fprintf(pipe1, "set term png \n");
81      fprintf(pipe1, "set logscale \n");
82      fprintf(pipe1, "set output 'log.png'\n");
83      fprintf(pipe1, "set xlabel '1/N' \n");
84      fprintf(pipe1, "set ylabel 'error' \n");
85      fprintf(pipe1, "plot '%s' using 1:2 w l title 'Euler', '%s' using 1:3 w l title
            'Classical RK'\n","log_error.txt","log_error.txt");
86      fclose(pipe1);
87
88      fclose(euler);
89      fclose(rk);
90      fclose(er_log);
91  }
```

Euler.c

GNUPLOT :

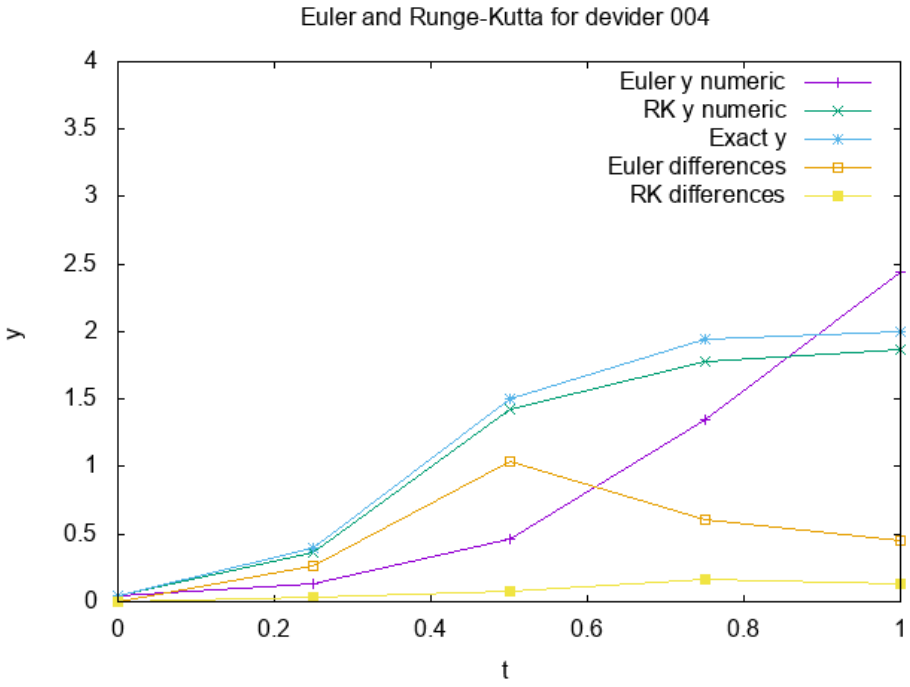

Figure 1:

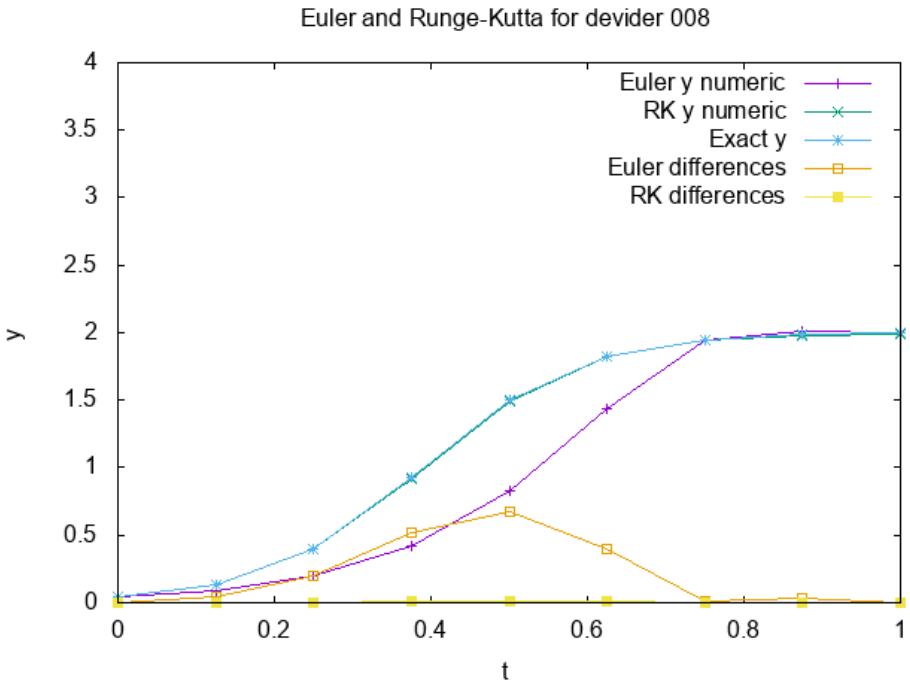

Figure 2:

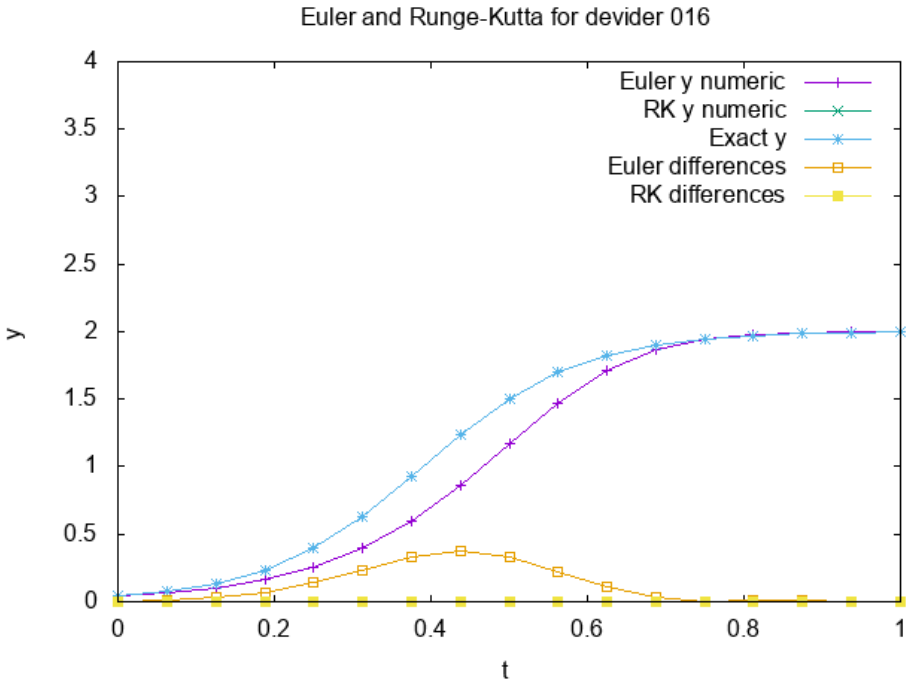

Figure 3:

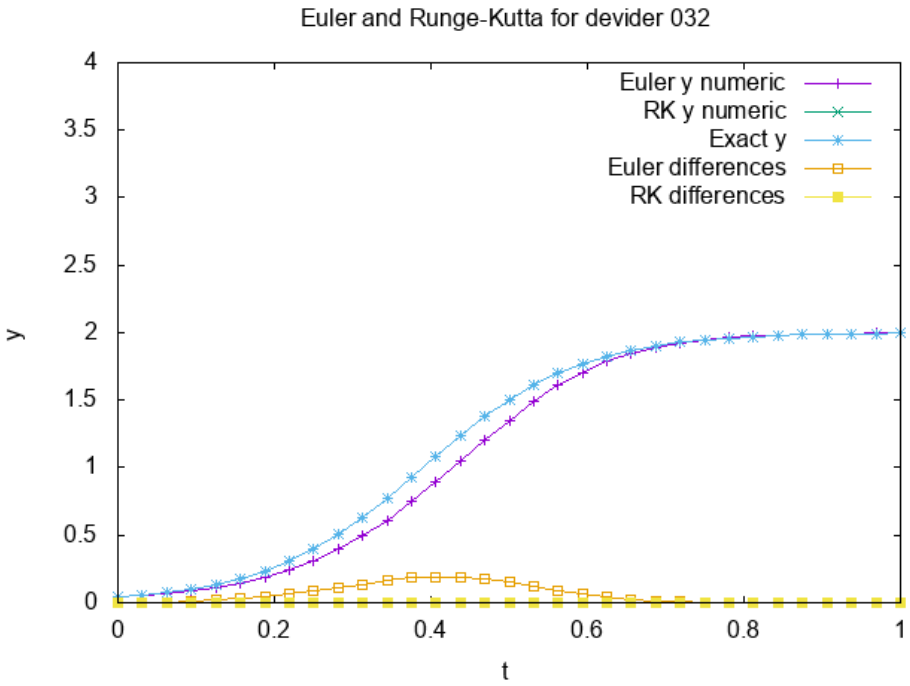

Figure 4:

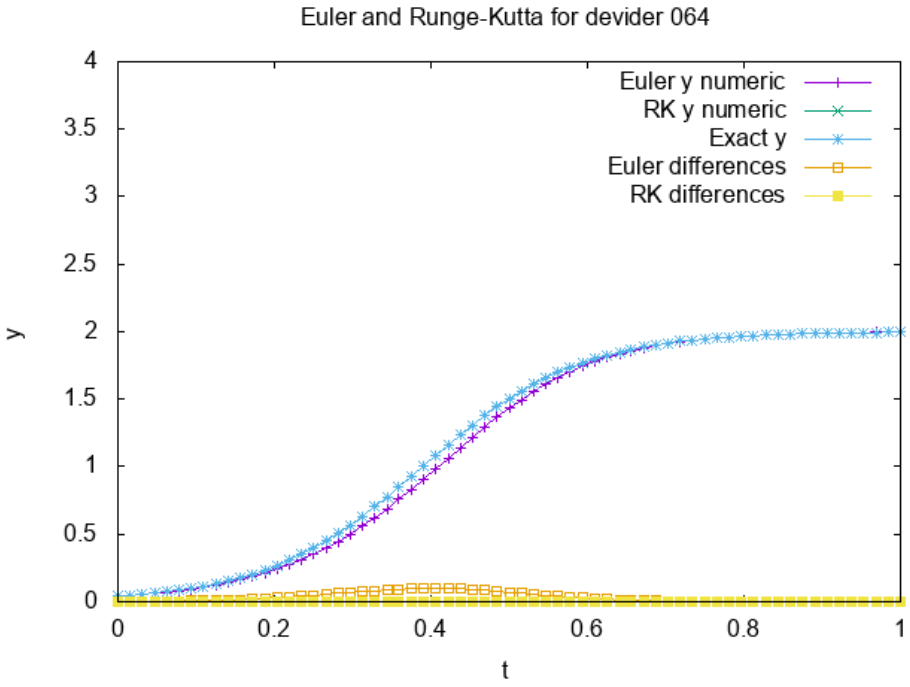

Figure 5:

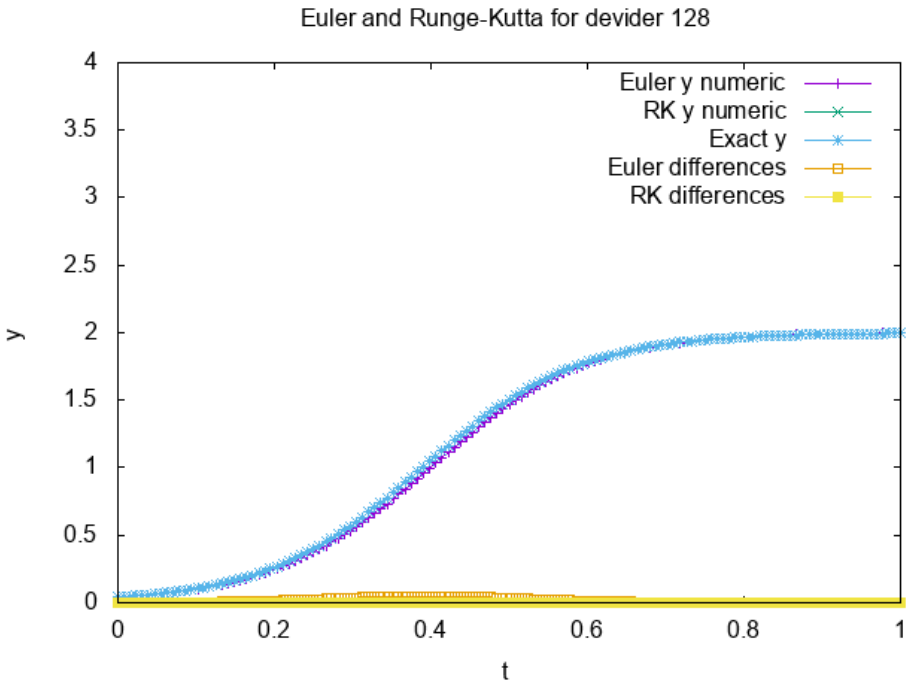

Figure 6: