INF1608 – Análise Numérica
Lab 8: Equações Diferenciais Ordinárias (EDO)

Prof. Waldemar Celes
Departamento de Informática, PUC-Rio

Considere a solução de equações diferenciais ordinárias expressas por:

$$y'(t) = f(t, y(t))$$

1. Pede-se:

   (a) O método de Runge-Kutta de ordem 4, considerando passos h constantes, é dado por:

$$\begin{aligned} k_1 &= hf(t, y(t)) \\ k_2 &= hf(t + h/2, y(t) + k_1/2) \\ k_3 &= hf(t + h/2, y(t) + k_2/2) \\ k_4 &= hf(t + h, y(t) + k_3) \\ y(t + h) &= y(t) + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \end{aligned}$$

   Implemente o método de Runge Kutta com passo constante. Sua função deve receber como parâmetros o tempo inicial t_0, o tempo final t_1, o passo de integração h, o valor inicial $y(t_0)$ e a função derivada $f(t, y(t))$, tendo como retorno o valor no tempo final $y(t_1)$, seguindo o protótipo:

```
double RungeKutta (double t0, double t1, double h, double y0,
                   double (*f) (double t, double y));
```

   Se necessário, o valor do último passo de integração deve ser ajustado para se obter o resultado no tempo t_1 especificado.

(b) Uma estratégia para implementação do método de Runge Kutta com passos adaptativos é a utilização de métodos acoplados. Um exemplo é o método que usa a tabela de parâmetros de Cash-Karp, dado por:

$$
\begin{aligned}
k_1 &= h\, f(t_i, y_i) \\
k_2 &= h\, f(t_i + a_2 h, y_i + b_{21} k_1) \\
&\vdots \\
k_6 &= h\, f(t_i + a_6 h, y_i + b_{61} k_1 + \cdots + b_{65} k_5) \\
y_{i+1} &= y_i + \sum_{i=1}^{6} c_i k_i + O(h^6) \\
y^*_{i+1} &= y_i + \sum_{i=1}^{6} c^*_i k_i + O(h^5) \\
\Delta &= y_{i+1} - y^*_{i+1}
\end{aligned}
$$

onde: a_i, b_{ij}, c_i, c^*_i são parâmetros da tabela *Cash-Karp* mostrada a seguir.

Cash-Karp Parameters for Embedded Runga-Kutta Method								
i	a_i	b_{ij}					c_i	c^*_i
1							$\frac{37}{378}$	$\frac{2825}{27648}$
2	$\frac{1}{5}$	$\frac{1}{5}$					0	0
3	$\frac{3}{10}$	$\frac{3}{40}$	$\frac{9}{40}$				$\frac{250}{621}$	$\frac{18575}{48384}$
4	$\frac{3}{5}$	$\frac{3}{10}$	$-\frac{9}{10}$	$\frac{6}{5}$			$\frac{125}{594}$	$\frac{13525}{55296}$
5	1	$-\frac{11}{54}$	$\frac{5}{2}$	$-\frac{70}{27}$	$\frac{35}{27}$		0	$\frac{277}{14336}$
6	$\frac{7}{8}$	$\frac{1631}{55296}$	$\frac{175}{512}$	$\frac{575}{13824}$	$\frac{44275}{110592}$	$\frac{253}{4096}$	$\frac{512}{1771}$	$\frac{1}{4}$
$j =$		1	2	3	4	5		

O erro do método é avaliado como sendo $\Delta = |y_{i+1} - y^*_{i+1}|$.
Considerando ϵ a tolerância (erro local aceitável), o fator de ampliação/redução do passo é dado por:

$$f = \sqrt[5]{\frac{\epsilon}{\Delta}}$$

Se $f \geq 1.0$, validamos o passo, e adotamos como solução a resposta de ordem superior: y_{i+1}. Neste caso, a próxima iteração pode ser avaliada com um novo passo:

$$h' = \min(1.2, f)\ h$$

Caso contrário, o passo é invalidado e tem que ser reavaliado com h atualizado:

$$h' = 0.9\ f\ h$$

Implemente o método de Runge Kutta acoplado, conforme apresentado acima. Sua função deve receber como parâmetros o tempo inicial t_0, o tempo final t_1, o valor inicial $y(t_0)$, a função derivada $f(t, y(t))$ e a tolerância do erro relativo tol, tendo como retorno o valor no tempo final $y(t_1)$. Como valor de passo inicial, deve-se adotar $h_0 = 10^{-7}$. A função deve ter o seguinte o protótipo:

```
double RungeKuttaAcoplado (double t0, double t1, double y0,
                          double (*f) (double t, double y), double tol);
```

Novamente, o valor do último passo de integração deve ser ajustado para se obter o resultado no tempo t_1 especificado.

2. Para testar suas funções, avalie $y(t)$, para $t = 2.4$ e outros valores, sabendo que $y' = ty + t^3$, com $y(0) = -1$. Para o método com passos constante igual, use, por exemplo, $h = 0.001$; para o método com passo adaptativo, use, por exemplo, $tol = 10^{-12}$.

   Sabe-se que a solução desta EDO para $y(0) = -1$ é:

$$y(t) = e^{\frac{t^2}{2}} - t^2 - 2$$

   Compare os resultados obtidos pelos métodos calculando o *erro relativo* para cada caso. Compare também o número de vezes que a função foi avaliada por cada método.

Agrupe os protótipos das funções pedidas em um módulo “ode.h” e as implementações em um módulo “ode.c”. Escreva o teste em outro módulo “main.c”.

Entrega: O código fonte deste trabalho (isto é, os arquivos “ode.c”, “ode.h” e “main.c”, *e eventuais códigos de laboratórios passados usados na solução*) deve ser enviado via página da disciplina no EAD. O prazo final para envio é **domingo, dia 24 de novembro**.