Autor: Karolina Tatarczyk

Metody numeryczne w technice

(kierunek Matematyka)

Projekt 3

Metoda Adamsa-Moultona

Napisać procedurę realizującą algorytm czterokrokowej metody Adamsa-Moultona (argumenty : f, x_0, y_0, b, n, m).
Wykorzystać metodę iteracji prostej (m powtórzeń), a jako metodę startową zastosować metodę Rungego-Kutty rzędy czwartego. Zminimalizować liczbę obliczeń funkcji f.

Korzystając z napisanej procedury wyznaczyć rozwiązanie przybliżone zagadnienia początkowego:

$$\begin{cases} y'(x) = \sin y(x), \quad x \in [0, 25], \\ y(0) = 1. \end{cases}$$

Obliczenia wykonać dla 10 i 20 kroków.

Na wspólnym rysunku wykreślić rozwiązanie dokładne oraz uzyskane rozwiązania przybliżone.Wykreślić także, na jednym rysunku, błędy uzyskanych rozwiązań przybliżonych.Policzyć ponadto błędy maksymalne oraz średnie dla obu siatek.

Rozwiązanie

In[5]:=
```
RungeKutty4[function_, X0_, Y0_, H_, number_] :=
 Module[{f = function, x0 = X0, y0 = Y0, h = H, n = number, x, y},
x = {x0};
y = {y0};
For[i = 1, i ≤ n, i++,
AppendTo[x, x[[i]] + h];
k1 = f[x[[i]], y[[i]]];
k2 = f[x[[i]] + h / 2, y[[i]] + h * k1 / 2];
k3 = f[x[[i]] + h / 2, y[[i]] + h * k2 / 2];
k4 = f[x[[i + 1]], y[[i]] + h * k3];
AppendTo[y, y[[i]] + h * (k1 + 2 * k2 + 2 * k3 + k4) / 6];
];
Return[Transpose[{x, y}]]
]
```

In[63]:=

In[6]:=
```
AdamsMoulton[function_, X0_, Y0_, B_, number_, M_] :=
 Module[{f = function, x0 = X0, y0 = Y0, b = B, m = M, n = number, Points},
k = 4;
vectorB = {251 / 720, 646 / 720, -264 / 720, 106 / 720, -19 / 720};
h = (b - x0) / n;
Points = RungeKutty4[f, x0, y0, h, k - 1];
ListF = Table[f[Points[[i, 1]], Points[[i, 2]]], {i, 1, k, 1}];
For[i = k, i ≤ n, i++,
xn = Points[[i, 1]] + h;
phi[z_] := Points[[i, 2]] +
     h * Sum[vectorB[[j + 1]] * ListF[[i + 1 - j]], {j, 1, k, 1}] + h * vectorB[[1]] * f[xn, z];
yn = Points[[i, 2]];
For[l = 1, l < m, l++,
yn = N[phi[yn]];
];
AppendTo[ListF, f[xn, yn]];
AppendTo[Points, {xn, yn}];
];
Return[Points]
]
```

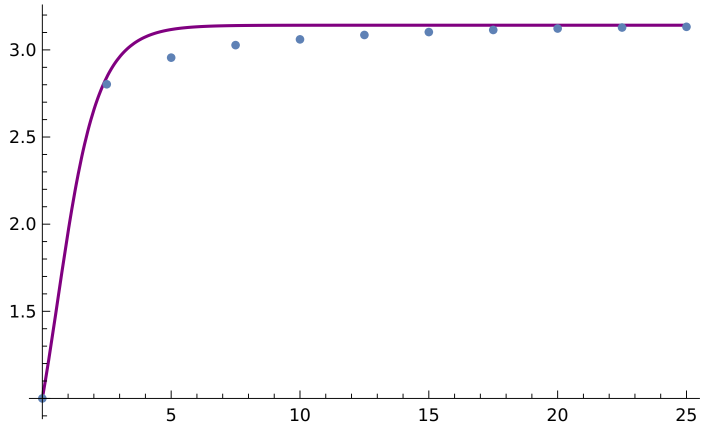
Obliczenia na przykładzie

In[12]:=
```
f[x_, y_] := Sin[y];
y0 = 1;
x0 = 0;
result = DSolve[{y'[x] == Sin[y[x]], y[0] == 1}, y[x], x][[1, 1, 2]];
plot = Plot[result, {x, 0, 25}, PlotStyle → Purple, PlotRange → All];
```

Testy n=10

In[88]:=
```
result10 = N[AdamsMoulton[f, x0, y0, 25, 10, 3]];
plot10 = ListPlot[result10];
Show[plot, plot10]
```

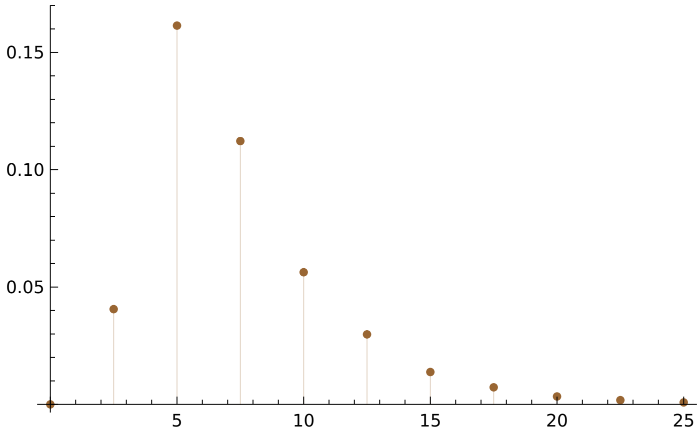
Out[90]=



Błędy otrzymanych wyników dla n=10

In[20]:=
```
ListX10 = Transpose[result10][[1]];
ListY10 = Transpose[result10][[2]];
resultPoints10 = Table[result /. {x → ListX10[[i]]}, {i, 1, Length[ListX10]}];
bladbezwzgledny10 = Abs[ListY10 - resultPoints10];
bar10 = ListPlot[Transpose[{ListX10, bladbezwzgledny10}],
  PlotStyle → Brown, Filling → Axis]
Print["Błąd maksymalny: ", Max[bladbezwzgledny10]]
Print["Średnia wartość błędu:", Mean[bladbezwzgledny10]]
```

Out[24]=




```
Błąd maksymalny: 0.161417
Średnia wartość błędu:0.0388542
```

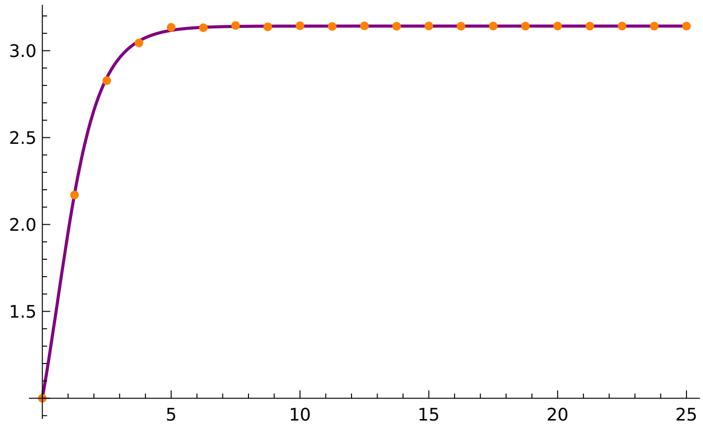
In[91]:=

Testy n=20

In[36]:=

In[37]:=
```
result20=N[AdamsMoulton[f,x0,y0,25,20,5]];
plot20=ListPlot[result20, PlotStyle→Orange];
Show[plot,plot20]
```

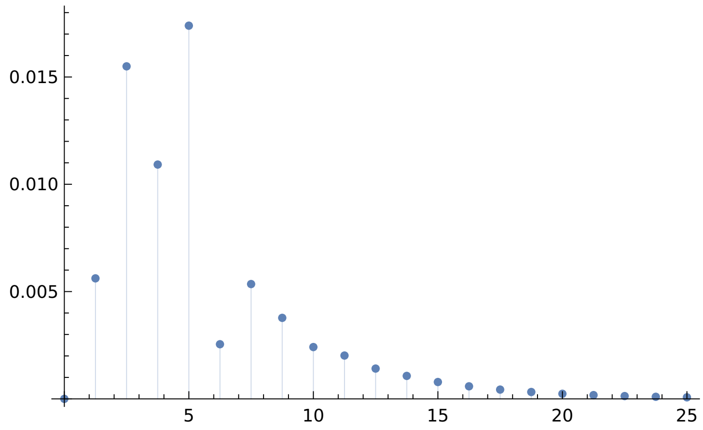
Out[39]=



Błędy otrzymanych rozwiązań dla n=20

In[40]:=
```
ListX20 = Transpose[result20][[1]];
ListY20 = Transpose[result20][[2]];
resultPoints20 = Table[result /. {x → ListX20[[i]]}, {i, 1, Length[ListX20]}];
bladbezwzgledny20 = Abs[ListY20 - resultPoints20];
bar20 = ListPlot[Transpose[{ListX20, bladbezwzgledny20}], Filling → Axis]
Print["Błąd maksymalny: ", Max[bladbezwzgledny20]]
Print["Średnia wartość błędu:", Mean[bladbezwzgledny20]]
```

Out[44]=

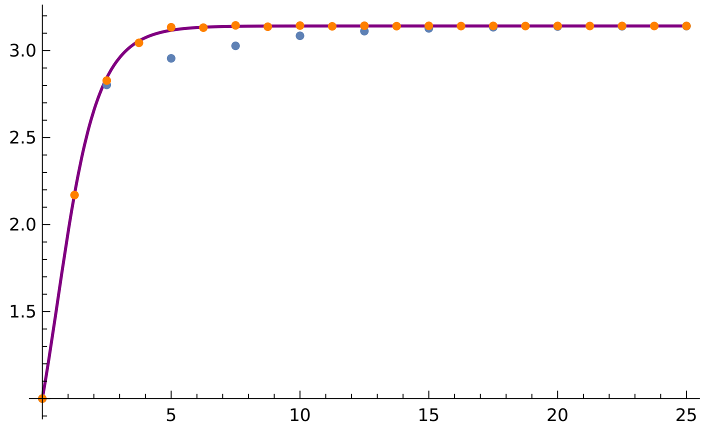



```
Błąd maksymalny: 0.0173915
Średnia wartość błędu:0.00337323
```

Otrzymane wyniki

In[47]:=
```
Show[plot, plot10, plot20]
```

Out[47]=



Błędy otrzymanych wyników

In[48]:=
```
Show[bar10, bar20]
```

Out[48]=