```
R0 = 6 378 137;
μ = 3.9860047 * 10^14;

h0 = 1 000 000;
R = R0 + h0;
h[x_, y_] := Sqrt[x^2 + y^2] - R0
T[R_] := 2 * π * Sqrt[R^3 / μ];
Vx0 = Sqrt[μ / R];
Rcube[x_, y_] := (Sqrt[x^2 + y^2])^3;
crkamat = {{1/2}, {0, 1/2}, {0, 0, 1}};
crkbvec = {1/6, 1/3, 1/3, 1/6};
crkcvec = {1/2, 1/2, 1};
ClassicalRungeKuttaCoefficients[4, p_] := N[{crkamat, crkbvec, crkcvec}, p];
benchmarkTest[tStart_, tEnd_, tStep_, printOutput_, rungeKutta4_] :=
  Module[{},
   system := {
     x'[t] == Vx[t],
     y'[t] == Vy[t],
     Vx'[t] == -μ / Rcube[x[t], y[t]] * x[t],
     Vy'[t] == -μ / Rcube[x[t], y[t]] * y[t],
     x[0] == 0,
     y[0] == R,
     Vx[0] == Vx0,
     Vy[0] == 0
    };
   solution := If[rungeKutta4,
     First@NDSolve[system, {x, y, Vx, Vy}, {t, 0, 10 000},
       Method → {"ExplicitRungeKutta", "DifferenceOrder" → 4, "Coefficients" →
          ClassicalRungeKuttaCoefficients}, StartingStepSize → tStep]
     ,
     First@NDSolve[
       system, {x, y, Vx, Vy}, {t, 0, 10 000}, PrecisionGoal → 10
      ]
    ];
   {Vx1, Vy1, x1, y1} = {Vx, Vy, x, y} /. solution;
   Print[Vx1[10 000 - tStep] - -6314.591];
   Print[Vy1[10 000 - tStep]  - 3761.713];
   Print[x1[10 000 - tStep] + 3 776 042];
   Print[y1[10 000 - tStep] - -6 338 645];

   Print[Vx1[10 000 - 1]];
   Print[Vy1[10 000 - 1]];
   Print[x1[10 000 - 1]];
   Print[y1[10 000 - 1]];
```

```
  ];

steps = {1, 0.1, 0.01};

times = ConstantArray[0, 12];

For[i = 3, i <= Length[steps], i++,
 times[[i]] =
  First@AbsoluteTiming[benchmarkTest[0, 10 000, steps[[i]], False, False]];
 times[[i + 6]] = First@AbsoluteTiming[
    benchmarkTest[0, 10 000, steps[[i]], False, True]];
]

times
```

-0.0000300461

-0.000377808

0.279127

0.467148

-6318.3

3755.48

-3.76979×10^6

-6.34237×10^6

-0.0000369835

-0.000393715

0.297848

0.456064

-6318.3

3755.48

-3.76979×10^6

-6.34237×10^6

{0, 0, 0.011001, 0, 0, 0, 0, 0, 4.773273, 0, 0, 0}