```
1   utilize acs::machinery
2   ... Look at what I've done
```

```
  2   fold::ruby { Look at what I've done
  3      require "rknbody.rb"
  4      ... process options
108      include Math
109      nb = Nbody.new
110      nb.simple_read
111      nb.evolve(method, eps,
112                dt, dt_dia, dt_out, dt_end,
113                init_out, x_flag)
114   }
```

```
  4  fold::ruby { process options
  5     ... print_help
 20     ... get parser
 40     ... read_options
 94     eps, dt, dt_dia, dt_out, dt_end,
 95     init_out, x_flag, method =
 96        read_options(parser)
 97     ... print options
107  }
```

```ruby
fold::ruby { print_help
  def print_help
    print "usage: ",
    $0,
    " [-h (for help)]",
    " [-s softening_length]",
    " [-d step_size]\n",
    " [-e diagnostics_interval]",
    " [-o output_interval]\n",
    " [-t total_duration]",
    " [-i (start output at t = 0)]\n",
    " [-x (extra diagnostics)]\n",
    " [-m integration_method]\n"
  end
}
```

```
20  fold::ruby { get parser
21     require "getoptlong"
22     parser = GetoptLong.new
23     ... get types A, N
27     parser.set_options(
28       ["-d", "--step_size", A],
29       ["-e", "--diagnostics_interval",A],
30       ["-h", "--help", N],
31       ["-i", "--initial_output", N],
32
33       ["-m", "--integration_method", A],
34       ["-o", "--output_interval", A],
35       ["-s", "--softening_length", A],
36       ["-t", "--total_duration", A],
37
38       ["-x", "--extra_diagnostics", N])
39  }
```

```
23   fold::ruby { get types A, N
24      A  =  GetoptLong::REQUIRED_ARGUMENT
25      N  =  GetoptLong::NO_ARGUMENT
26   }
```

```
40   fold::ruby { read_options
41      def read_options(parser)
42        ... set defaults
52        loop do
53           ... scan options
89        end
90        return eps, dt, dt_dia, dt_out, dt_end,
91               init_out, x_flag, method
92      end
93   }
```

```
42  fold::ruby { set defaults
43     dt = 0.001
44     dt_dia = 1
45     dt_out = 1
46     dt_end = 10
47     eps = 0
48     init_out = false
49     x_flag = false
50     method = "rk4"
51  }
```

```
53  fold::ruby { scan options
54     begin
55        opt, arg = parser.get
56        break if not opt
57        ... known option?
84     rescue => err
85        print_help
86        exit  # exit if option unknown
87     end
88  }
```

```
57   fold::ruby { known option?
58      case opt
59        ... -d => -m
72        ... -o => -x
82      end
83   }
```

```
59  fold::ruby { -d => -m
60     when "-d"
61       dt = arg.to_f
62     when "-e"
63       dt_dia = arg.to_f
64     when "-h"
65       print_help
66       exit    # exit after providing help
67     when "-i"
68       init_out = true
69     when "-m"
70       method = arg
71  }
```

```
72  fold::ruby { -o => -x
73     when "-o"
74       dt_out = arg.to_f
75     when "-s"
76       eps = arg.to_f
77     when "-t"
78       dt_end = arg.to_f
79     when "-x"
80       x_flag = true
81  }
```

```
 97   fold::ruby { print options
 98      STDERR.print "eps = ", eps, "\n",
 99            "dt = ", dt, "\n",
100            "dt_dia = ", dt_dia, "\n",
101            "dt_out = ", dt_out, "\n",
102            "dt_end = ", dt_end, "\n",
103            "init_out = ", init_out, "\n",
104            "x_flag = ", x_flag, "\n",
105            "method = ", method, "\n"
106   }
```