zad. 11.5

January 14, 2022

[3]:
```
using LinearAlgebra
```

[6]:
```
function AdaptiveSimpson(f, a, b; abstol=1.0e-6)
    nf = 3;
    ff = [f(a), f((a+b)/2), f(b)];
    nf = 3; # Initial Simpson approximation
    I1 = (b-a)*dot([1, 4, 1], ff)/6;
    function adaptrec(f, a, b, ff, I1, tol, nf)
        h = (b-a)/2;
        fm = [f(a+h/2), f(b-h/2)];
        nf = nf + 2;
        # Simpson approximations for left and right subinterval
        fR = [ff[2], fm[2], ff[3]];
        fL = [ff[1], fm[1], ff[2]];
        IL = h*dot([1, 4, 1], fL)/6;
        IR = h*dot([1, 4, 1], fR)/6;
        I2 = IL + IR;
        I = I2 + (I2 - I1)/15;
        # Extrapolated approximation
        if (abs(I-I2) > tol)
            IL, nf = adaptrec(f, a, a+h, fL, IL, tol/2, nf);
            IR, nf = adaptrec(f, b-h, b, fR, IR, tol/2, nf);
            I = IL + IR;
        end
        #print(nf, "\n")
        return I, nf;
    end
    return adaptrec(f, a, b, ff, I1, abstol, nf);
end;
```

[7]:
```
f(x) = sqrt(x + 1.0) # 133
f(x) = sqrt(x^3 + 1.0)/(x+2.0) # 161
f(x) = sqrt(1.0 - x^2) # 265
f(x) = sqrt(sqrt(1.0 - x^2)) # 529
f(x) = sqrt(sqrt(1.0 - x^4)) #  60969, a następnie StackOverflowError:
#f(x) = sqrt(sqrt(sqrt(1-x))) #  61161, a następnie StackOverflowError:

 print(AdaptiveSimpson(f, -1.0, 1.0))
```

```
StackOverflowError:

Stacktrace:
  [1] ^
    @ .\math.jl:920 [inlined]
  [2] literal_pow
    @ .\intfuncs.jl:317 [inlined]
  [3] f
    @ .\In[7]:5 [inlined]
  [4] (::var"#adaptrec#6")(f::typeof(f), a::Float64, b::Float64, ff::
 ↪Vector{Float64}, I1::Float64, tol::Float64, nf::Int64)
    @ Main .\In[6]:8
  [5] (::var"#adaptrec#6")(f::typeof(f), a::Float64, b::Float64, ff::
 ↪Vector{Float64}, I1::Float64, tol::Float64, nf::Int64) (repeats 2 times)
    @ Main .\In[6]:19
  [6] (::var"#adaptrec#6")(f::typeof(f), a::Float64, b::Float64, ff::
 ↪Vector{Float64}, I1::Float64, tol::Float64, nf::Int64) (repeats 7618 times)
    @ Main .\In[6]:20
  [7] (::var"#adaptrec#6")(f::typeof(f), a::Float64, b::Float64, ff::
 ↪Vector{Float64}, I1::Float64, tol::Float64, nf::Int64) (repeats 54 times)
    @ Main .\In[6]:19
  [8] AdaptiveSimpson(f::typeof(f), a::Float64, b::Float64; abstol::Float64)
    @ Main .\In[6]:26
  [9] AdaptiveSimpson(f::Function, a::Float64, b::Float64)
    @ Main .\In[6]:2
 [10] top-level scope
    @ In[7]:8
 [11] eval
    @ .\boot.jl:360 [inlined]
 [12] include_string(mapexpr::typeof(REPL.softscope), mod::Module, code::String
 ↪filename::String)
    @ Base .\loading.jl:1116
```