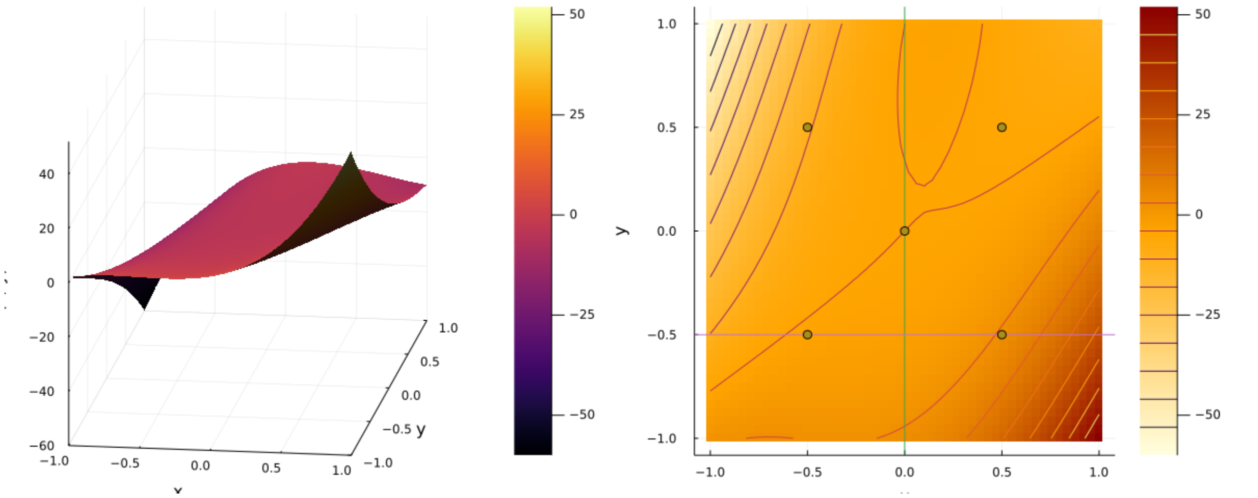
General Information: This assignment contains written and/or programming tasks. Combine all the answers to the written tasks in a single PDF document, named `{lastname}-written.pdf`. You can also scan or take pictures of (readable) handwritten papers. JPEG/PNG image files are accepted in this case and they should be named `{exercisenumber}-{lastname}-written.{jpeg/png}`. Make sure that we can follow the manual calculations. Do not combine too many small steps into one. The programming tasks have to be solved in *Julia* and the source code files have to be submitted using the following naming scheme: `{exercisenumber}-{lastname}.jl`.

(1) (2 points) Given

$$f : \mathbb{R}^2 \to \mathbb{R} \tag{1}$$
$$f(x, y) = -4x^2 + 2(2x - y)^3 - (x + y)^3 + x + 4y^2 - y - 4 \tag{2}$$



a) Compute the partial derivatives of f, i.e.

$$\frac{\partial f(x,y)}{\partial x}, \frac{\partial f(x,y)}{\partial y} \tag{3}$$

$$\boxed{\begin{aligned} \frac{\partial f(x,y)}{\partial x} &= 12(2x - y)^2 - 3(x + y)^2 - 8x + 1 & (4) \\ \frac{\partial f(x,y)}{\partial y} &= -6(2x - y)^2 - 3(x + y)^2 + 8y - 1 & (5) \end{aligned}}$$

b) Find the minimum of the function $f(x, y)$ along the y-axis where $x = 0$ and $y \in [-1, 1]$. Find the minimum of the function $f(x, y)$ along the x-axis where $y = -1/2$ and $x \in [-1, 1]$. See the lines in the Figure above for a visual interpretation.

$$\begin{aligned}
\frac{\partial f(0,y)}{\partial y} &= -9y^2 + 8y - 1 && (6)\\
&= 0 && (7)\\
&\iff \begin{cases} y = y_1 \approx 0.150472 \text{ or} \\ y = y_2 \approx 0.738417 \end{cases} && (8)\\
& && (9)\\
\frac{\partial f(0,y)}{\partial y} < 0 &\iff \begin{cases} y < y_1 \text{ or} \\ y > y_2 \end{cases} && (10)\\
&\Rightarrow \text{candidates for } \min f = \begin{cases} y = y_1 \text{ or} \\ y = 1 \end{cases} && (11)\\
&\text{then check by plugging-in} && (12)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial f(x,-\frac{1}{2})}{\partial x} &= 45x^2 + 19x + \frac{13}{4} && (13)\\
&= 0 \iff x \in \mathbb{C} && (14)\\
&\Rightarrow \text{candidates for } \min f = \begin{cases} x = -1 \text{ or} \\ x = 1 \end{cases} && (15)\\
&\text{then check by plugging-in} && (16)
\end{aligned}$$

c) The gradient of a real-valued function at some point is a 2D-vector pointing into the direction of highest increase. Formally, it is given by the partial derivatives evaluated at that point, written as:

$$\nabla f(a,b) = \left(\frac{\partial}{\partial x} f(a,b), \frac{\partial}{\partial y} f(a,b)\right) \tag{17}$$

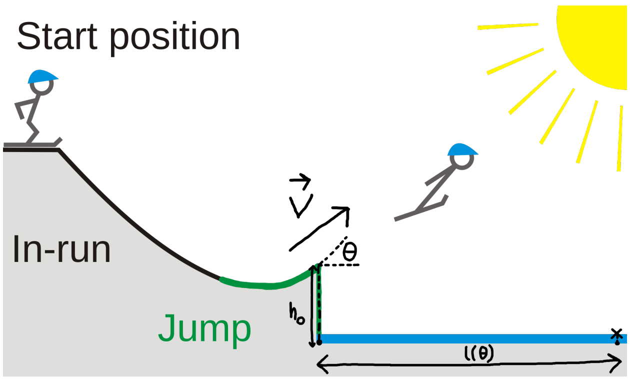
Compute the following gradients and their magnitude, and sketch their direction on paper:

$$\nabla f\left(-\frac{1}{2},-\frac{1}{2}\right), \nabla f\left(\frac{1}{2},\frac{1}{2}\right), \nabla f\left(-\frac{1}{2},\frac{1}{2}\right), \nabla f\left(\frac{1}{2},-\frac{1}{2}\right), \nabla f\left(0,-\frac{1}{2}\right) \tag{18}$$

The positions are indicated by the dots in the Figure above.

$$\begin{aligned}
\nabla f &= \begin{pmatrix} 12(2x-y)^2 - 3(x+y)^2 - 8x + 1 \\ -6(2x-y)^2 - 3(x+y)^2 + 8y - 1 \end{pmatrix} \Rightarrow && (19)\\
\left\|\nabla f\left(-\frac{1}{2},-\frac{1}{2}\right)\right\| &= \left\|\left\{5,-\frac{19}{2}\right\}\right\| = \frac{\sqrt{461}}{2} \approx 10.7355 && (20)\\
\left\|\nabla f\left(\frac{1}{2},\frac{1}{2}\right)\right\| &= \left\|\left\{-3,-\frac{3}{2}\right\}\right\| = \frac{3\sqrt{5}}{2} \approx 3.3541 && (21)\\
\left\|\nabla f\left(-\frac{1}{2},\frac{1}{2}\right)\right\| &= \left\|\left\{32,-\frac{21}{2}\right\}\right\| = \frac{\sqrt{4537}}{2} \approx 33.6786 && (22)\\
\left\|\nabla f\left(\frac{1}{2},-\frac{1}{2}\right)\right\| &= \left\|\left\{24,-\frac{37}{2}\right\}\right\| = \frac{\sqrt{3673}}{2} \approx 30.3026 && (23)\\
\left\|\nabla f\left(0,-\frac{1}{2}\right)\right\| &= \left\|\left\{\frac{13}{4},-\frac{29}{4}\right\}\right\| = \frac{\sqrt{\frac{505}{2}}}{2} \approx 7.94512 && (24)
\end{aligned}$$

(2) (2 points) A skier wants to improve the WR on ski jumping and can decide to take off the platform at any angle θ. Neglecting the effect of any external forces except gravity, which angle θ should the skier choose in order to maximize the length of the jump $l(\theta)$ (starting from the ground location of taking off, i.e. when there is just air under the skis)?



(a) Scheme of the problem, by Arsenikk (talk) - Schema_einer_Skisprungschanze.svg, CC BY-SA 3.0

(b) Landing, by Clément Bucco-Lechat - Own work, CC BY-SA 3.0

Hint: height of skier at time t, i.e. $h(t)$ is given by

$$h(t) = h_0 + tv_y - \frac{gt^2}{2} = h_0 + tv\sin\theta - \frac{gt^2}{2} \tag{25}$$

where h_0 is the height of taking off, v_y is the vertical component of the velocity when taking off, g the gravity acceleration.

$$
\begin{aligned}
h(\text{end of flight}) &= h(T(\theta)) \tag{26}\\
&= 0 \tag{27}\\
&= h_0 + T(\theta)v_y - \frac{gT(\theta)^2}{2} \tag{28}\\
\Rightarrow T(\theta) &= \frac{-v_y \pm \sqrt{v_y^2 + 2gh_0}}{g} \tag{29}\\
&= \frac{-v\sin\theta + \sqrt{v^2\sin\theta^2 + 2gh_0}}{g} \tag{30}\\
\Rightarrow \text{distance at end of flight} = d(\theta) &= v_x T(\theta) \tag{31}\\
& \tag{32}\\
\text{candidates for } \max d(\theta) &\iff \frac{\mathrm{d}d(\theta)}{\mathrm{d}\theta} = 0 \tag{33}\\
&\iff \frac{\mathrm{d}}{\mathrm{d}\theta}\frac{v(\cos\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)}{g} = 0 \tag{34}\\
&\iff \frac{\frac{\mathrm{d}}{\mathrm{d}\theta}v(\cos\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)}{g} = 0 \tag{35}\\
&\iff \frac{v\left(\frac{\mathrm{d}}{\mathrm{d}\theta}(\cos\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)\right)}{g} = 0 \tag{36}\\
&\iff \frac{1}{g}v\left(\cos\theta\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right) + \left(\frac{\mathrm{d}}{\mathrm{d}\theta}\cos\theta\right)\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right) = 0 \tag{37}\\
&\iff \frac{1}{g}v\left(\cos\theta\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right) - (\sin\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)\right) = 0 \tag{38}\\
&\iff \frac{1}{g}v\left(\cos\theta\left(\frac{\mathrm{d}}{\mathrm{d}\theta}v(\sin\theta) + \frac{\mathrm{d}}{\mathrm{d}\theta}\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)}\right) - (\sin\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)\right) = 0 \tag{39}\\
&\iff \frac{1}{g}v\left(\cos\theta\left(v\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\sin\theta\right) + \frac{\mathrm{d}}{\mathrm{d}\theta}\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)}\right) - (\sin\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)\right) = 0 \tag{40}\\
&\iff \frac{1}{g}v\left(\cos\theta\left(v(\cos\theta) + \frac{\mathrm{d}}{\mathrm{d}\theta}\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)}\right) - (\sin\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)\right) = 0 \tag{41}\\
&\iff \frac{1}{g}v\left(\cos\theta\left(v(\cos\theta) + \frac{\frac{\mathrm{d}}{\mathrm{d}\theta}2gh_0 + v^2\left(\sin\theta^2\right)}{2\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)}}\right) - (\sin\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)\right) = 0 \tag{42}\\
&\iff \frac{1}{g}v\left(\cos\theta\left(v(\cos\theta) + \frac{\frac{\mathrm{d}}{\mathrm{d}\theta}2gh_0 + \frac{\mathrm{d}}{\mathrm{d}\theta}v^2\left(\sin\theta^2\right)}{2\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)}}\right) - (\sin\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)\right) = 0 \tag{43}\\
&\iff \frac{1}{g}v\left(\cos\theta\left(v(\cos\theta) + \frac{\frac{\mathrm{d}}{\mathrm{d}\theta}v^2\left(\sin\theta^2\right)}{2\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)}}\right) - (\sin\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)\right) = 0 \tag{44}\\
&\iff \frac{1}{g}v\left(\cos\theta\left(v(\cos\theta) + \frac{v^2\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\sin\theta^2\right)}{2\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)}}\right) - (\sin\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)\right) = 0 \tag{45}\\
&\iff \frac{1}{g}v\left(\cos\theta\left(v(\cos\theta) + \frac{v^2\left(\frac{\mathrm{d}}{\mathrm{d}\theta}\sin\theta\right)\sin\theta}{\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)}}\right) - (\sin\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)\right) = 0 \tag{46}\\
&\iff \frac{v\left((\cos\theta)\left(\frac{v^2(\sin\theta)(\cos\theta)}{\sqrt{2gh_0+v^2(\sin\theta^2)}} + v(\cos\theta)\right) - (\sin\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right)\right)}{g} = 0 \tag{47}\\
&\iff (\cos\theta)\left(\frac{v^2(\sin\theta)(\cos\theta)}{\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)}} + v(\cos\theta)\right) - (\sin\theta)\left(\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)} + v(\sin\theta)\right) = 0 \tag{48}\\
&\iff \frac{v^2\sin\theta\cos^2\theta}{\sqrt{2gh_0 + v^2\left(\sin\theta^2\right)}} + v\cos^2\theta - \sin\theta\sqrt{2gh_0 + v^2\sin^2\theta} - v\sin^2\theta = 0 \tag{49}\\
&\iff \frac{v^2\sin\theta(1-\sin^2\theta)}{\sqrt{2gh_0 + v^2\sin^2\theta}} + v - v\sin^2\theta - \sin\theta\sqrt{2gh_0 + v^2\sin^2\theta} - v\sin^2\theta = 0 \tag{50}\\
&\iff \frac{v^2\sin\theta(1-\sin^2\theta)}{\sqrt{2gh_0 + v^2\sin^2\theta}} + v - 2v\sin^2\theta - \sin\theta\sqrt{2gh_0 + v^2\sin^2\theta} = 0 \tag{51}\\
&\iff \ldots \tag{52}\\
&\iff \theta = \begin{cases} -\arcsin\left(\frac{v}{\sqrt{2gh_0+2v^2}}\right) & \text{or}\\ \arcsin^{-1}\left(\frac{v}{\sqrt{2gh_0+2v^2}}\right) \end{cases} \tag{53}
\end{aligned}
$$

(3) (1.5 points) Newton's method is an iterative root-finding algorithm. Starting from an initial value of x, each iteration contains the following steps:

a) compute tangent at $f(x)$

b) compute intersection between the tangent and the axis $x = 0$

c) use this intersection point as the new value of x

The value of x should converge towards the root of f. An easy way to implement Newton's method is to compute, at each iteration, the new value of x using Equation 54

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, f'(x_k) \neq 0 \tag{54}$$

where x_{k+1} is the new value of x and x_k the value of x at the previous iteration.

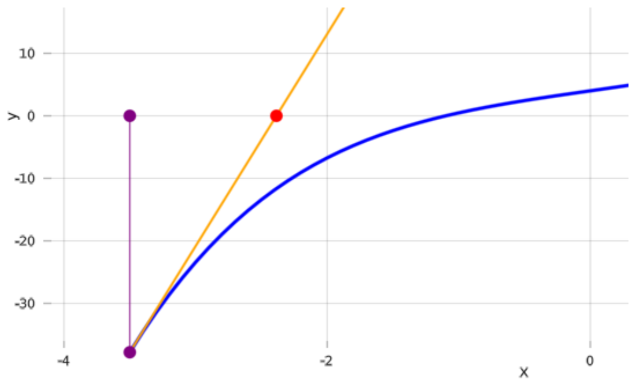


Figure 2: One iteration of Newton's method. The values of x_k and $f(x_k)$ are shown by purple dots. The tangent at $f(x_k)$ is displayed in orange. The value of x_{k+1} is figured by a red dot.

a) implement Newton's method to find the root of

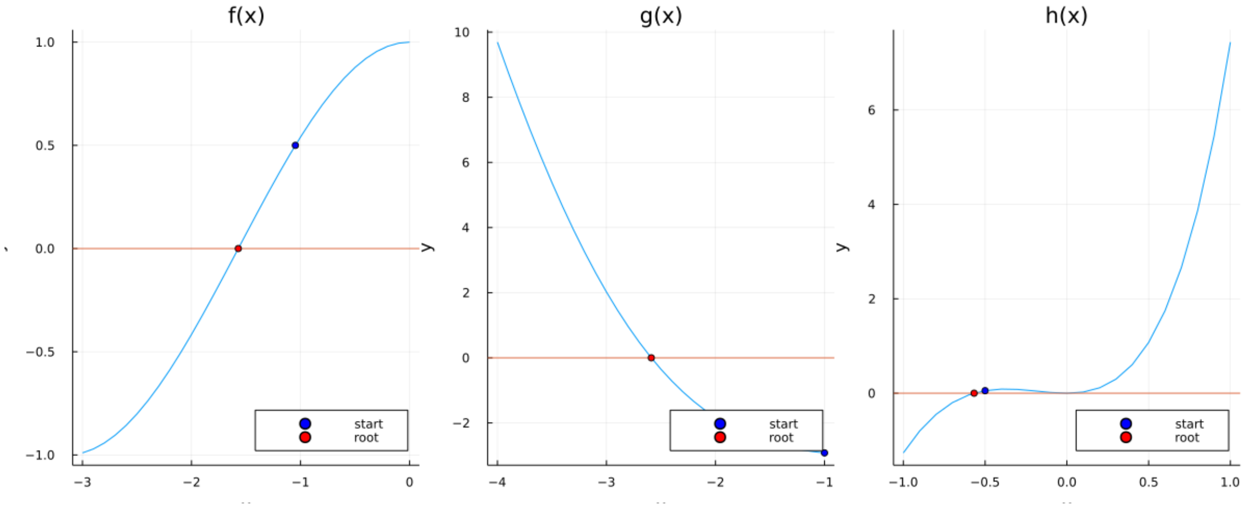
$$f(x) = \cos(x) \tag{55}$$
$$g(x) = x^2 + 2\cos(x) - 5 \tag{56}$$
$$h(x) = 2x^2 e^x + 2x^3 \tag{57}$$

Code is already provided to you in `newton.jl`. Implement the `f, g, h` and `f_prime, g_prime, h_prime` and the update step to find x_{k+1}.

b) what happens if we choose to start the search for the root of $f(x)$ with $x = 0$?

c) explain how you can use Newton's method to find a local minimum or maximum of a function

$$\boxed{\begin{aligned} \text{local minimum/maximum} \iff\ & f'(x) = 0 && (58)\\ \Rightarrow\ & x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}, f''(x_k) \neq 0 && (59)\end{aligned}}$$




```
newton_method(f, f_prime, 0, 10)
```

```
MethodError: no method matching newton_method(::var"#9#10", ::var"#11#12", ::Int64, ::Int64)
Closest candidates are:
  newton_method(::Any, ::Any, ::Float64, ::Integer) at In[21]:1

Stacktrace:
 [1] top-level scope
   @ In[42]:1
 [2] eval
   @ ./boot.jl:373 [inlined]
 [3] include_string(mapexpr::typeof(REPL.softscope), mod::Module, code::String, filename::St
ring)
   @ Base ./loading.jl:1196
```

(4) (1.5 points) *Gradient descent* is an iterative algorithm for finding a local minimum of a differentiable function. Starting from some point (a_n, b_n) the idea is to move against the gradient $\nabla f(a_n, b_n)$, into the direction of a local minimum. Formally, we add the negative of the gradient to the current estimate (a_n, b_n), multiplied by a scalar γ in order to get a new better estimate (a_{n+1}, b_{n+1}):

$$\begin{pmatrix} a_{n+1} \\ b_{n+1} \end{pmatrix} = \begin{pmatrix} a_n \\ b_n \end{pmatrix} - \gamma \nabla f(a_n, b_n) \tag{60}$$

By repeated application of this rule we move closer and closer to a local minimum until a certain number of iterations is reached.

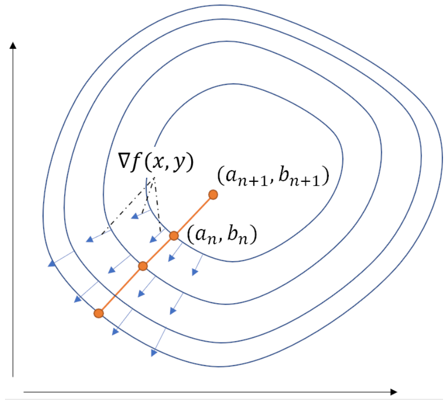
Implement in Julia *gradient descent* for the three functions below, by completing the code provided to you in `gradient_descent.jl`:

$$f(x,y) = 3x^2 + (x+2y)^2 + 5(2x-y)^2 + (x-y-3)^2 + 2y^2 \tag{61}$$

$$g(x,y) = \frac{\cos(x) + \sin(y)}{2} \tag{62}$$

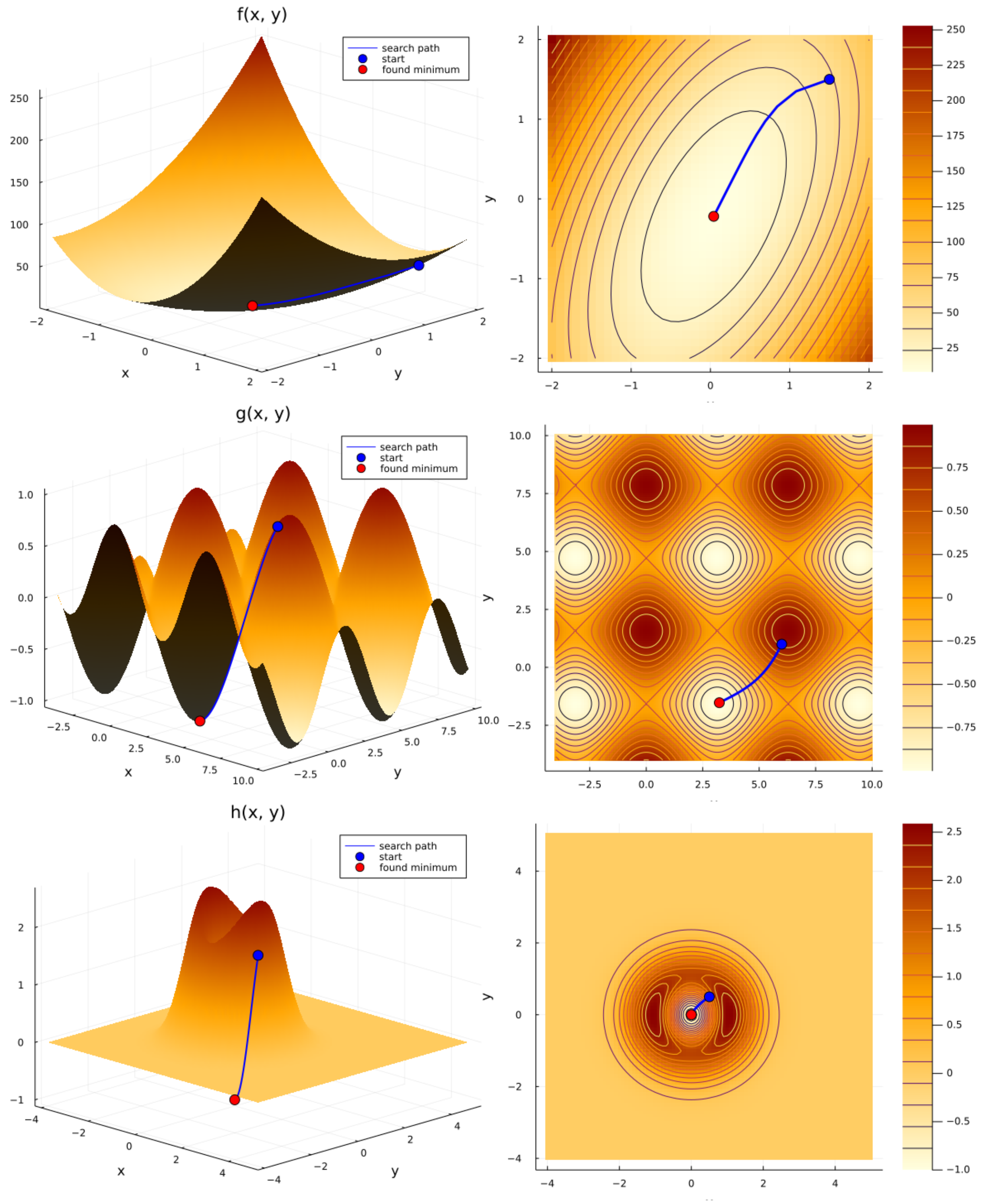
$$h(x,y) = (8x^2 + 6 * y^2 - 1)\mathrm{e}^{-x^2-y^2} \tag{63}$$

a) compute the partial derivatives of f, g and h. Implement the functions `f`, `g`, `h` and `f_gradient`, `g_gradient`, `h_gradient`



b) fill in the update step (Equation 60) which moves the current estimate into the opposite direction of the gradient.

c) is the method guaranteed to find a global minimum? Is the method guaranteed to find a local minimum? What happens if we choose $[\cos^{-1}(1), \pi/2]$ as the starting point for finding a minimum of $g(x, y)$?


f(x, y)
search path
start
found minimum
x
y
g(x, y)
search path
start
found minimum
x
y
h(x, y)
search path
start
found minimum
x
y

```
gradient_descent([acos(1), π / 2], g_gradient, 0.01, 1000)
```

```
1000×2 Matrix{Float64}:
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 ⋮
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
 0.0  1.5708
```