```
> restart : with(LinearAlgebra) : with(VectorCalculus) : with(plots) :
  SetCoordinates(cartesian[x, y]) :
>
> # Define the function
  f := (x, y, z, t) -> 4*x*y*z + t^2 - x^2 - y^2 - z^2;

  # Set the coordinates for partial derivatives
  SetCoordinates(cartesian[x, y, z, t]);

  # Calculate the gradient
  grad_f := Gradient(f(x, y, z, t)) :

  # Solve for critical points by setting each component of the gradient to zero
  critical_points := solve({grad_f[1] = 0, grad_f[2] = 0, grad_f[3] = 0, grad_f[4]
      = 0}, {x, y, z, t}) :

  # Calculate the Hessian matrix of f
  hessian_f := Hessian(f(x, y, z, t)) :
```

$$f := (x, y, z, t) \mapsto 4 \cdot x \cdot y \cdot z + t^2 + (-x^2) + (-y^2) + (-z^2)$$

$$cartesian_{x, y, z, t} \tag{1}$$

```
> # Now for every point we need to calculate the eigenvalues and explore
      these
> eval_hessian := subs({x = 0, y = 0, z = 0, t = 0}, hessian_f);
```

$$eval\_hessian := \begin{bmatrix} -2 & 0 & 0 & 0 \\ 0 & -2 & 0 & 0 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} \tag{2}$$

```
> # For point (0,0,0,0) we get a saddle point cuz lambda {-2, 2}, thus pos and
      negative
> # For point (1:2, 1:2, 1:2, 1:0)
```

$$> \quad eval\_hessian := Eigenvalues\left(subs\left(\left\{x = \frac{1}{2}, y = \frac{1}{2}, z = \frac{1}{2}, t = 0\right\}, hessian\_f\right)\right);$$

$$eval\_hessian := \begin{bmatrix} 2 \\ -4 \\ 2 \\ -4 \end{bmatrix} \tag{3}$$

```
> # saddle point, and so forth...
>
```