```
iteration_method := proc(f, y0, n)
  local y, i;
  y[0] := x→y0;
  for i from 1 to n do
    y[i] := unapply(simplify((y0 + int(f(s, y[i - 1](s)) , s = 0 ..x))), x) :
  end;
  return eval(y[n])
end:

taylor_series_method := proc(f, x0, y0, n)
   local derivs, coef, j, a, y1;
   derivs := proc(f, n)
      option remember;
      if n = 1 then f(x, y(x))
      else
         subs(diff(y(x), x) = f(x, y(x)), diff(derivs(f, n−1), x))
      fi;
   end:
   coef := proc(f, n, x0, y0)
      local d;
      if n = 0 then y0
      else
        d := unapply(derivs(f, n), x) :
        subs(y(x0) = y0, d(x0))
      fi;
   end;
   for j from 0 to n do
     a[j] := 1/j! coef(f, j, x0, y0);
   end;
   y1 := sum('a[j]·(x − x0)^j', 'j' = 0 ..n) ;
   return unapply(y1, x);
end:

euler_method := proc(f, x0, y0, a, b, n)
   local h, x, y, k, j;
   h := evalf((b − a)/n);
   x[0] := x0; y[0] := y0;
   for k from 0 to n − 1 do
      x[k + 1] := evalf(x[k] + h);
      y[k + 1] := evalf(y[k] + h·f(x[k], y[k]))
   end;
   return [x[j], y[j]]$j = 0 ..n;
end:
```

```
runge_kutta := proc(f, x0, y0, a, b, n)
  local h, x, y, k, K1, K2, K3, K4, j;
  h := evalf((b - a)/n);
  x[0] := x0; y[0] := y0;
  for k from 0 to n - 1 do
    x[k + 1] := evalf(x[k] + h);
    K1 := f(x[k], y[k]);
    K2 := f(x[k] + h/2, y[k] + h/2*K1);
    K3 := f(x[k] + h/2, y[k] + h/2*K2);
    K4 := f(x[k] + h, y[k] + h*K3);
    y[k + 1] := evalf(y[k] + h*(1/6*K1 + 2/6*K2 + 2/6*K3 + 1/6*K4));
  end;
  return [x[j], y[j]]$j = 0 ..n;
end:
```

1

```
with(plots) :

evaluate1 := proc(f, x0, y0, len, n_it, n_num)
  local it, ty, eu, rk, p_eu, p_rk, p_it, p_ty, a, b;
  a := x0; b := x0 + len;
  it := iteration_method(f, y0, n_it);
  ty := taylor_series_method(f, x0, y0, n_it) ;
  eu := euler_method(f, x0, y0, a, b, n_num);
  rk := runge_kutta(f, x0, y0, a, b, n_num);
  p_it := plot(it(x), x = a - 1 ..b, color = red);
  p_ty := plot(ty(x), x = a - 1 ..b, color = green);
  p_eu := plot([[eu]], style = point, symbol = circle);
  p_rk := plot([[rk]], style = point, symbol = cross);
  display({p_it, p_ty, p_eu, p_rk});
  # display(p_it);
  #display(p_eu);
  #display(p_rk);
  #display({p_eu, p_rk});
end:
```

a.

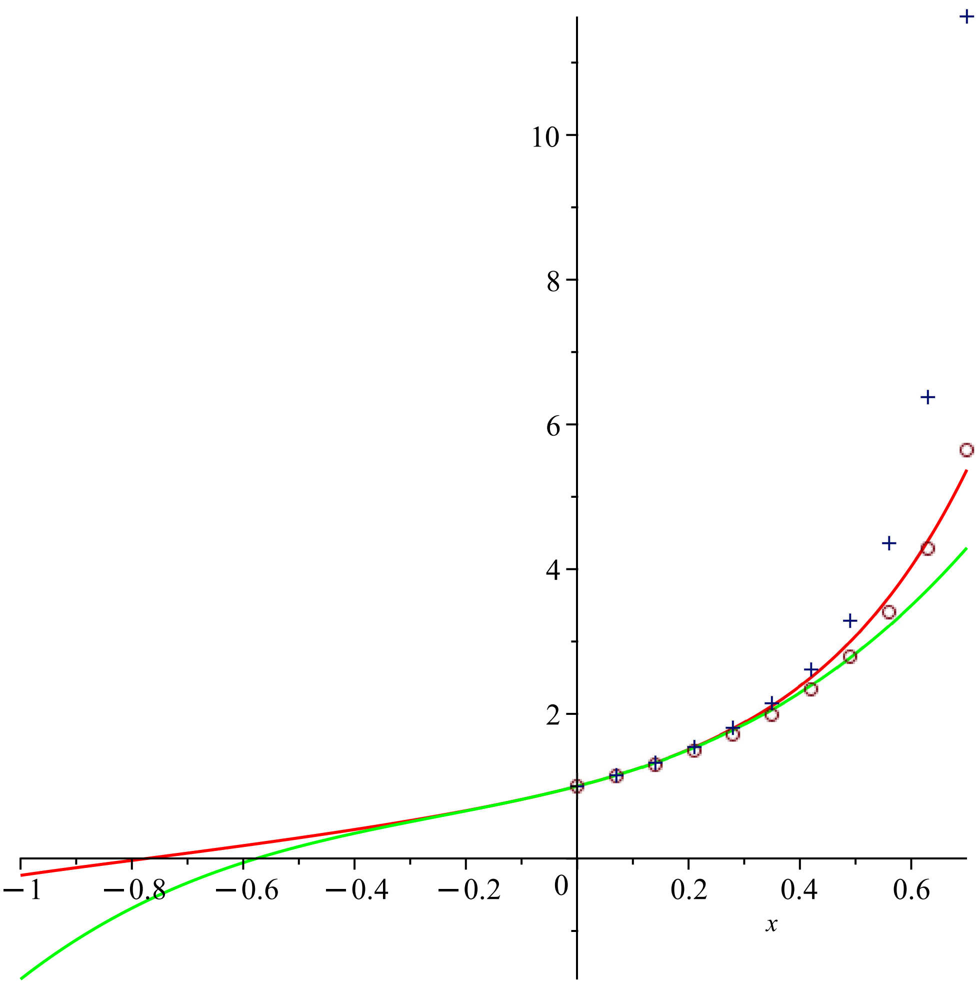
$ys := dsolve(\{D(y)(x) = 1 + y(x)^2, y(0) = 1\}, y(x))$

$$ys := y(x) = \tan\left(x + \frac{\pi}{4}\right) \qquad (1)$$

$runge\_kutta\left((x, y) \rightarrow 1 + y^2, 0, 1, 0, \frac{1}{2}, 5\right);$

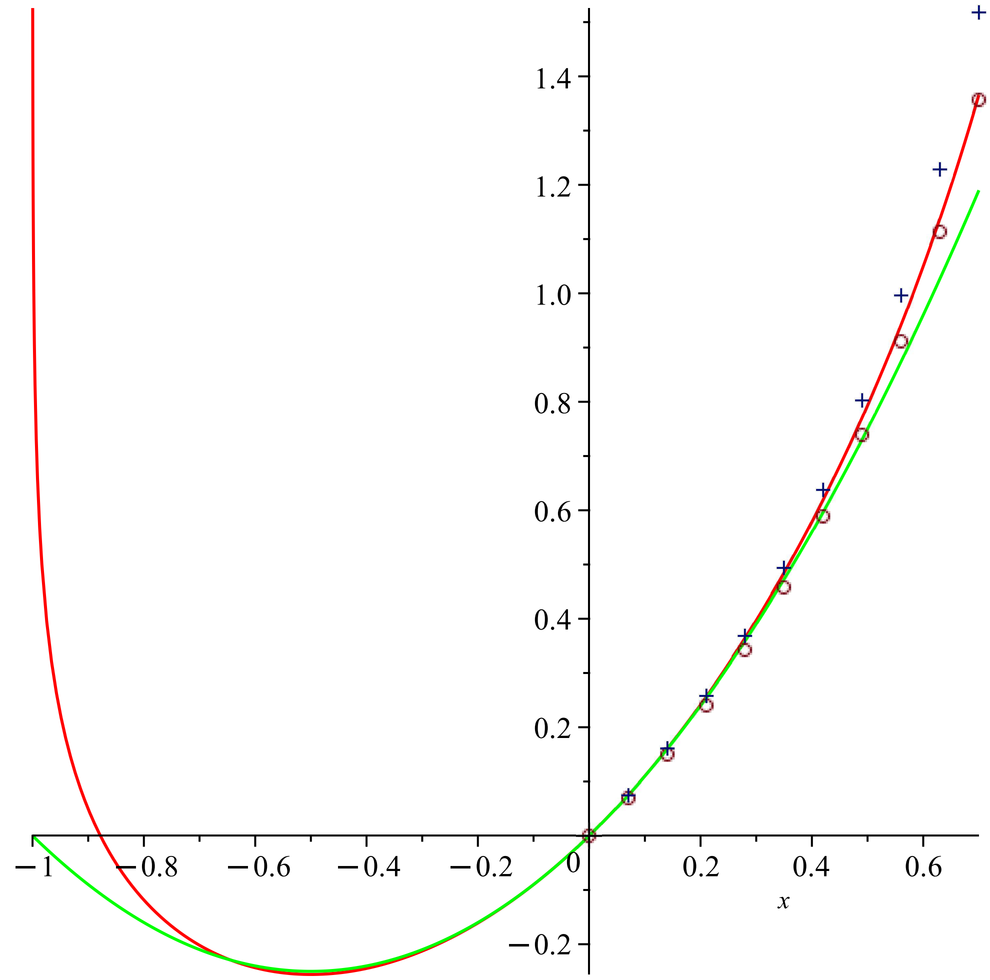
$[0, 1], [0.1000000000, 1.223048914], [0.2000000000, 1.508496167], [0.3000000000, 1.895754160], [0.4000000000, 2.464899687], [0.5000000000, 3.407820425]$ **(2)**

$evaluate1\left((x, y) \rightarrow 1 + y^2, 0, 1, 0.7, 3, 10\right)$



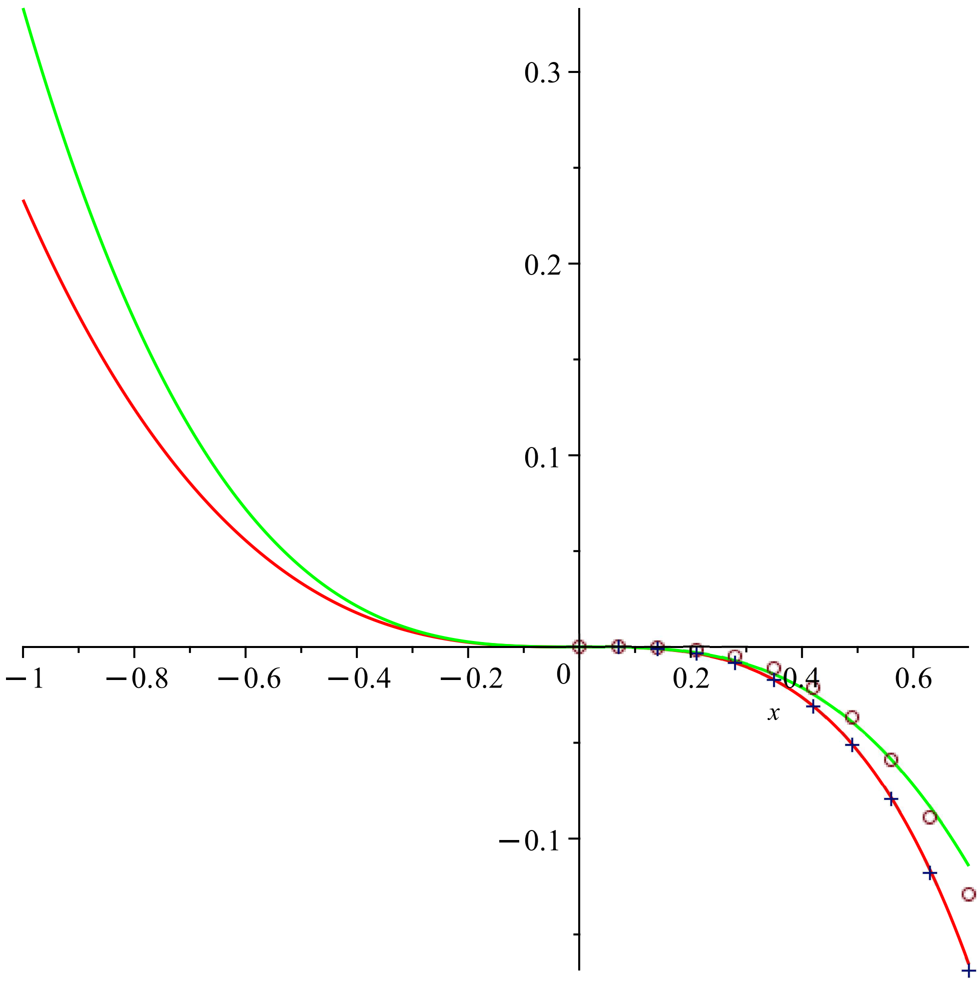
b.

$evaluate1\left((x, y) \rightarrow \frac{y}{1 - x^2} + 1 + x, 0, 0, 0.7, 2, 10\right)$



c.

$evaluate1\left((x, y) \rightarrow 2y - x^2, 0, 0, 0.7, 3, 10\right)$



2.

a.

```
with(DEtools) :
shooting_method := proc(d_eq, y, x00, y00, x01, y01)
  local a, b, ic1, y_alpha, eq, alpha1;
  a := x00; b := x01;
   ic1 := y(a) = y00, D(y)(a) = alpha;
   y_alpha := unapply(simplify(rhs(dsolve({d_eq, ic1}, y(x)))) , x, alpha) ;
  eq := y_alpha(b, alpha) = y01;
  alpha1 := solve(eq, alpha);
  return eval(y_alpha), alpha1, eval(simplify(y_alpha(x, alpha1)))
 end:
```

$y\_alpha, alpha1, y\_sol := shooting\_method\left(D(D(y))(x) + y(x) = x^3, y, 0, 1, \frac{Pi}{2}, 0\right)$

$$y\_alpha, \alpha 1, y\_sol := (x, \alpha) \mapsto \sin(x)\cdot(\alpha + 6) + \cos(x) + x^3 - 6\cdot x, -6 - \frac{1}{8}\pi^3 + 3\pi, \tag{3}$$

$$\frac{\sin(x)\left(-\pi^3 + 24\,\pi\right)}{8} + \cos(x) + x^3 - 6\,x$$

eval(*y_alpha*)

$$(x, \alpha) \mapsto \sin(x)\cdot(\alpha + 6) + \cos(x) + x^3 - 6\cdot x \tag{4}$$

alpha1

$$-6 - \frac{1}{8}\,\pi^3 + 3\,\pi \tag{5}$$

eval(*y_sol*)

$$\frac{\sin(x)\left(-\pi^3 + 24\,\pi\right)}{8} + \cos(x) + x^3 - 6\,x \tag{6}$$

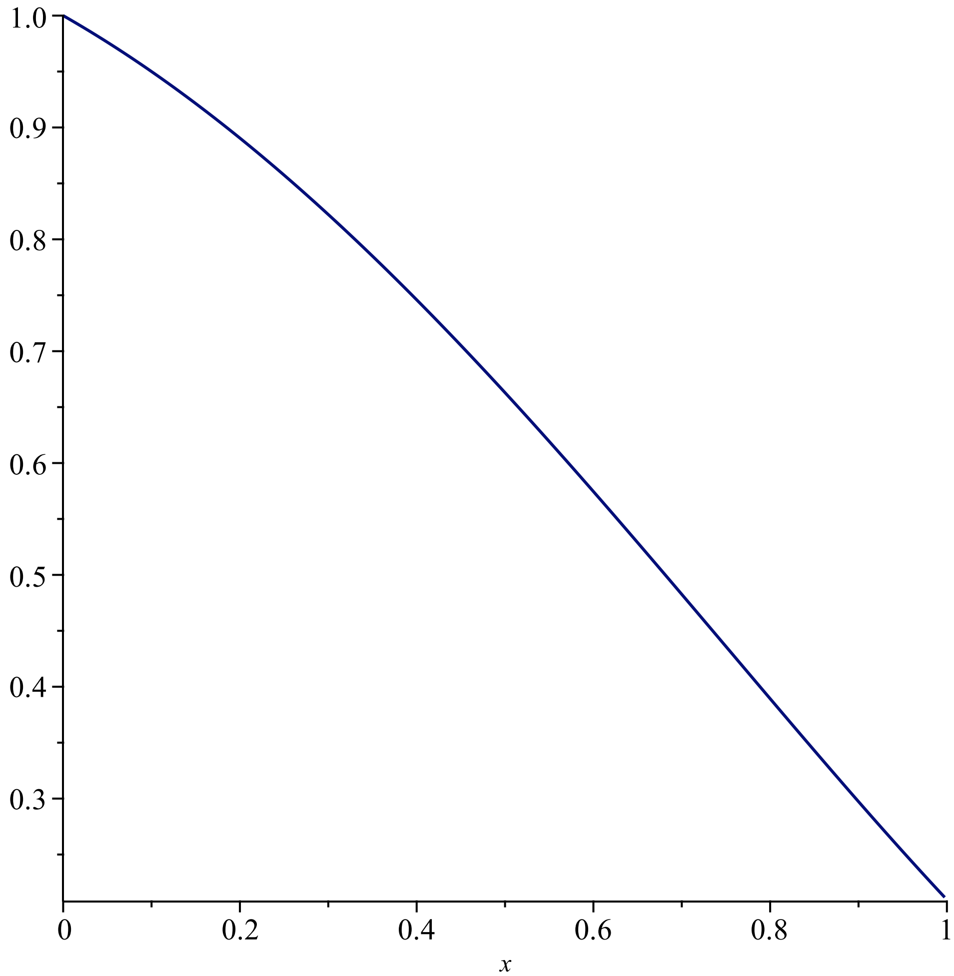
$y\_alpha\left(\frac{\mathrm{Pi}}{2}, alpha1\right) = 0$

$$0 = 0 \tag{7}$$

$y\_exact := rhs\left(dsolve\left(\left\{\mathrm{D}(\mathrm{D}(y))(x) + y(x) = x^3, y(0) = 1, y\left(\frac{\mathrm{Pi}}{2}\right) = 0\right\}, y(x)\right)\right)$

$$y\_exact := \sin(x)\left(-\frac{1}{8}\,\pi^3 + 3\,\pi\right) + \cos(x) + x^3 - 6\,x \tag{8}$$

$display(plot(y\_sol, x = 0\,..1), plot(y\_exact, x = 0\,..1))$



b.

$d\_eq := \mathrm{D}(\mathrm{D}(y))(x) + \mathrm{D}(y)(x) = 1;$

$$d\_eq := \mathrm{D}^{(2)}(y)(x) + \mathrm{D}(y)(x) = 1 \tag{9}$$

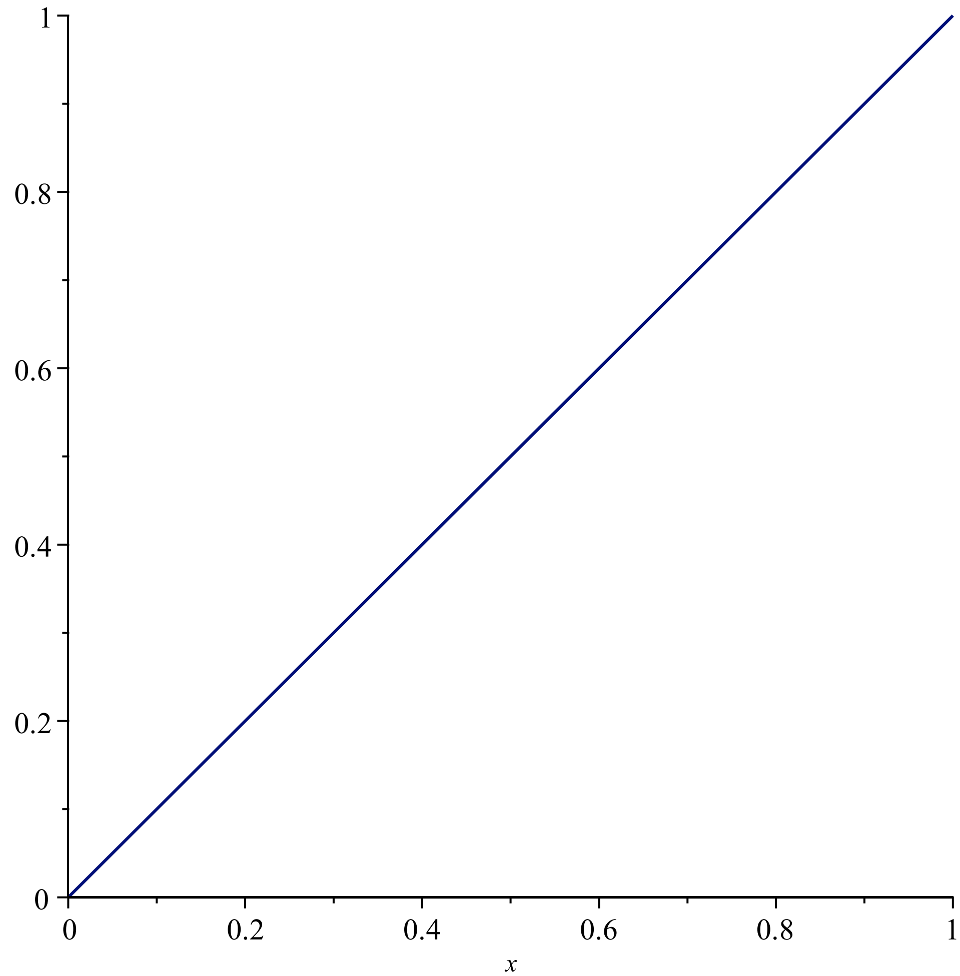
$y\_alpha, alpha1, y\_sol := shooting\_method(\mathrm{d\_eq}, y, 0, 0, 1, 1)$

$$y\_alpha, \alpha 1, y\_sol := (x, \alpha) \mapsto -\mathrm{e}^{-x} \cdot \alpha + \mathrm{e}^{-x} + \alpha + x - 1, 1, x \tag{10}$$

$y\_exact := rhs(dsolve(\{d\_eq, y(0) = 0, y(1) = 1\}, y(x)))$

$$y\_exact := x \tag{11}$$

$display(plot(y\_sol, x = 0..1), plot(y\_exact, x = 0..1))$



c

$$d\_eq := \mathrm{D}(\mathrm{D}(y))(x) + 3\,\mathrm{D}(y)(x) + 2\,y(x) = \frac{1}{\exp(x) + 1};$$

$$d\_eq := \mathrm{D}^{(2)}(y)(x) + 3\,\mathrm{D}(y)(x) + 2\,y(x) = \frac{1}{\mathrm{e}^{x} + 1} \tag{12}$$

$$y\_alpha, alpha1, y\_sol := shooting\_method\left(\mathrm{d\_eq}, \mathrm{y}, 0, 2\ln(2) + 2, 1, \frac{(\exp(1) + 1)}{\exp(1)^{2}}(\ln(\exp(1) + 1) + 1)\right)$$

$$y\_alpha, \alpha 1, y\_sol := (x, \alpha) \mapsto \mathrm{e}^{-2\cdot x}\cdot\left(\ln\left(\mathrm{e}^{x} + 1\right)\cdot\left(\mathrm{e}^{x} + 1\right) - \ln\left(\mathrm{e}^{x}\right)\cdot\mathrm{e}^{x} + \left(x + \alpha + 3\cdot\ln(2) + 3\right)\cdot\mathrm{e}^{x} - \alpha - 3\cdot\ln(2) - 1\right), -\frac{1}{(\mathrm{e})^{2}\,\mathrm{e}^{-2}\,(\mathrm{e} - 1)}\left(3\ln(2)\,(\mathrm{e})^{3}\,\mathrm{e}^{-2} + (\mathrm{e})^{3}\,\mathrm{e}^{-2}\ln(\mathrm{e} + 1) - 3\,(\mathrm{e})^{2}\ln(2)\,\mathrm{e}^{-2} + 3\,(\mathrm{e})^{3}\,\mathrm{e}^{-2} + (\mathrm{e})^{2}\ln(\mathrm{e} + 1)\,\mathrm{e}^{-2} - (\mathrm{e})^{2}\,\mathrm{e}^{-2} - \ln(\mathrm{e} + 1)\,\mathrm{e} - \mathrm{e} - \ln(\mathrm{e} + 1) - 1\right), \left(\ln\left(\mathrm{e}^{x} + 1\right)\mathrm{e}^{x} - \ln\left(\mathrm{e}^{x}\right)\mathrm{e}^{x} + x\,\mathrm{e}^{x} + \ln\left(\mathrm{e}^{x} + 1\right) + \mathrm{e}^{x} + 1\right)\mathrm{e}^{-2x} \tag{13}$$