11-10_Line_Integral

November 11, 2021

[2]:
```
from sympy import *
from sympy import vector
```

[3]:
```
R = vector.CoordSys3D('R')
```

[4]:
```
F = (R.x * R.y)**2 * R.i + 2 * R.x**3 * R.y / 3 * R.j + R.z * R.k
```

[6]:
```
var('t')
r1 = t * R.i + t / 2 * R.j + t**2 * R.k
r2 = t**2 * R.i + t / 2 * R.j + t * R.k
```

[7]:
```
r1_p = diff(r1, t)
r2_p = diff(r2, t)
```

[8]:
```
r1_p
```

[8]: $\mathbf{\hat{i}_R} + (\frac{1}{2})\mathbf{\hat{j}_R} + (2t)\mathbf{\hat{k}_R}$

[9]:
```
r1.coeff(R.i)
```

[9]: t

[10]:
```
F.subs(R.x, r1.coeff(R.i))
```

[10]: $(\mathbf{y_R}^2 t^2)\mathbf{\hat{i}_R} + (\frac{2\mathbf{y_R} t^3}{3})\mathbf{\hat{j}_R} + (\mathbf{z_R})\mathbf{\hat{k}_R}$

[11]:
```
def vec_subs(vec1, vec2, coord):
  return vec1.subs({
    coord.x: vec2.coeff(coord.i),
    coord.y: vec2.coeff(coord.j),
    coord.z: vec2.coeff(coord.k),
  })
```

[12]:
```
vec_subs(F, r1, R)
```

[12]: $(\frac{t^4}{4})\mathbf{\hat{i}_R} + (\frac{t^4}{3})\mathbf{\hat{j}_R} + (t^2)\mathbf{\hat{k}_R}$

[14]:
```
vec_subs(F, r1, R).dot(r1_p)
```

[14]:

$$\frac{5t^4}{12} + 2t^3$$

[15]:
```
integrate(vec_subs(F, r1, R).dot(r1_p), t)
```

[15]:
$$\frac{t^5}{12} + \frac{t^4}{2}$$

[16]:
```
integrate(vec_subs(F, r1, R).dot(r1_p), (t, 0, 1))
```

[16]:
$$\frac{7}{12}$$

[17]:
```
integrate(vec_subs(F, r2, R).dot(r2_p), (t, 0, 1))
```

[17]:
$$\frac{7}{12}$$

[ ]:
```

```