Interpolation cubature formula

```
newtMethod[fun_, a_, b_, m_] := Module[{n = 3 m, h, x, x1 = {}, s1, s2}, h = (b - a)/n;
  x = a + h;
  Do[AppendTo[x1, x]; x += h, n];
  s1 = 3 * Sum[fun[x1[[3 i - 2]]] + fun[x1[[3 i - 1]]], {i, 1, m}];
  s2 = 2 * Sum[fun[x1[[3 i]]], {i, 1, m - 1}];
  3/8 h (fun[a] + s1 + s2 + fun[x1[[3 m]]]) // N]
```

Check

```
fun1[x_] := x^2

newtMethod[(# - 2)/2 &, 1, 10, 2]

15.75

Integrate[(x - 2)/2, {x, 1, 10}] // N

15.75

cubeInt[fun_, a_, b_, c_, d_, n_, m_] :=
 Module[{x0, x1 = {}, y0, y1 = {}, ω, n1 = 3 n, m1 = 3 m, v, h1, h2}, h1 = (b - a)/n1;
  h2 = (c - d)/m1;
  x0 = a;
  y0 = c;
  Do[AppendTo[x1, x0]; x0 += h1, n1];
  Do[AppendTo[y1, y0]; y0 += h2, m1];
  ω = Product[x - x1[[j]], {j, 1, n}];
  v = Product[y - y1[[j]], {j, 1, m}];
  Sum[Integrate[ω/((x - x1[[i]]) (D[ω, x] /. x → x1[[i]])), {x, a, b}] *
    Sum[Integrate[v/((y - y1[[j]]) (D[v, y] /. y → y1[[j]])), {y, c, d}] * fun[x1[[i]], y1[[j]]],
     {j, 1, m}], {i, 1, n}]]

fun[x_, y_] := x + y

cubeInt[fun, 2, 10, 3, 15, 10, 10]

1440
```

Check

```
∫_2^10 ∫_3^15 (x + y) dy dx // N

1440.
```