[Self] MiniQuiz Lec13 [ALL CORRECT]

Q1 [CORRECT]

Let $f(x) = 0.2 + 25x - 200x^2 + 675x^3 - 900x^4 + 400x^5$

Evaluate the following integral:

$I = \int_0^{0.8} f(x)dx$

a) analytically

b) single application of the trapezoidal rule

c) composite trapezoidal rule for 4 segments (ns = 4)

d) single application of Simpson's 1/3 rule

e) composite Simpson's 1/3 rule for 4 segments (ns = 4)

f) single application of Simpson's 3/8 rule

g) composite Simpson's 3/8 rule for 4 segments (ns = 4)

For each of the numerical estimates b) through g), determine the absolute of the true percent relative error based on a).

h) at least how many segments in composite trapezoidal rule needed to be computed for $|e_t| < 1\%$?

i) at least how many segments in composite Simpson's 1/3 rule needed to be computed for $|e_t| < 1\%$?

j) at least how many segments in composite Simpson's 3/8 rule needed to be computed for $|e_t| < 1\%$?

Rounding a decimal number to four decimal places.

a) I_{true} =

b) I_{trap_1} = , $|e_t|$ =

c) I_{trap_4} = , $|e_t|$ =

d) $I_{simpson1/3\_1}$ = , $|e_t|$ =

e) $I_{simpson1/3\_4}$ = , $|e_t|$ =

f) $I_{simpson3/8\_1}$ = , $|e_t|$ =

g) $I_{simpson3/8\_4}$ = , $|e_t|$ =

h) ns_{trap} =

i) $ns_{simpson1/3}$ =

j) $ns_{simpson3/8}$ =

a

```
f = @(x) 0.2 + 25.*x - 200.*x.^2 + 675.*x.^3 - 900.*x.^4 + 400.*x.^5;
a = 0; b = 0.8;
Itrue = integral(f, a, b)
```

b

```
Itrap1 = trap(f, a, b, 1)
errtrap1 = round(abs((Itrue - Itrap1) / Itrue) * 100, 4)
```

c

```
Itrap4 = trap(f, a, b, 4)
errtrap4 = round(abs((Itrue - Itrap4) / Itrue) * 100, 4)
```

d

```
% we are actually applying this to the whole thing, while it can be only
% applied to a range of panels (ns)
ns = 1;
Is13 = 0;
for i = 1:ns
    h = (b - a) / (2*ns);
    x0 = (2*(i-1)*h) + a;
    simpson13 = h/3 * (f(x0) + (4*f(x0 + h)) + f(x0 + 2*h));
    Is13 = Is13 + simpson13;
end
round(Is13, 4)
errs13 = round(abs((Itrue - Is13) / Itrue) * 100, 4)
```

e

```
% we are actually applying this to the whole thing, while it can be only
% applied to a range of panels (ns)
ns = 4;
Is13 = 0;
for i = 1:ns
    h = (b - a) / (2*ns);
    x0 = (2*(i-1)*h) + a;
    simpson13 = h/3 * (f(x0) + (4*f(x0 + h)) + f(x0 + 2*h));
    Is13 = Is13 + simpson13;
end
round(Is13, 4)
errs13 = round(abs((Itrue - Is13) / Itrue) * 100, 4)
```

f

```
% we are actually applying this to the whole thing, while it can be only
% applied to a range of panels (ns)
ns = 1;
Is38 = 0;
for i = 1:ns
```

```
    h = (b - a) / (3*ns);
    x0 = (3*(i-1)*h) + a;
    simpson38 = ((3*h)/8) * (f(x0) + (3*f(x0 + h)) + (3*f(x0 + (2*h))) + f(x0 + 3*h));
    Is38 = Is38 + simpson38;
end
round(Is38, 4)
errs38 = round(abs((Itrue - Is38) / Itrue) * 100, 4)
```

g

```
% we are actually applying this to the whole thing, while it can be only
% applied to a range of panels (ns)
ns = 4;
Is38 = 0;
for i = 1:ns
    h = (b - a) / (3*ns);
    x0 = (3*(i-1)*h) + a;
    simpson38 = ((3*h)/8) * (f(x0) + (3*f(x0 + h)) + (3*f(x0 + (2*h))) + f(x0 + 3*h));
    Is38 = Is38 + simpson38;
end
round(Is38, 4)
errs38 = round(abs((Itrue - Is38) / Itrue) * 100, 4)
```

h

```
Itrapn = trap(f, a, b, 13)
errtrapn = round(abs((Itrue - Itrapn) / Itrue) * 100, 4)
% n = 13
```

i

```
ns = 3;
Is13 = 0;
for i = 1:ns
    h = (b - a) / (2*ns);
    x0 = (2*(i-1)*h) + a;
    simpson13 = h/3 * (f(x0) + (4*f(x0 + h)) + f(x0 + 2*h));
    Is13 = Is13 + simpson13;
end
round(Is13, 4)
errs13 = round(abs((Itrue - Is13) / Itrue) * 100, 4)
% n = 3
```

j

```
ns = 2;
Is38 = 0;
for i = 1:ns
    h = (b - a) / (3*ns);
    x0 = (3*(i-1)*h) + a;
    simpson38 = ((3*h)/8) * (f(x0) + (3*f(x0 + h)) + (3*f(x0 + (2*h))) + f(x0 + 3*h));
    Is38 = Is38 + simpson38;
end
round(Is38, 4)
errs38 = round(abs((Itrue - Is38) / Itrue) * 100, 4)
% n = 2
```

Q2 [CORRECT]

Evaluate the following integral: $I = \int_{-1}^{1} e^x dx$

a) analytically

b) single application of the trapezoidal rule

c) composite trapezoidal rule for 4 segments (ns = 4)

d) single application of Simpson's 1/3 rule

e) composite Simpson's 1/3 rule for 4 segments (ns = 4)

f) single application of Simpson's 3/8 rule

g) composite Simpson's 3/8 rule for 4 segments (ns = 4)

For each of the numerical estimates b) through g), determine the absolute of the true percent relative error based on a).

Rounding a decimal number to four decimal places.

a) I_{true} =

b) I_{approx} = , $|e_t|$ =

c) I_{approx} = , $|e_t|$ =

d) I_{approx} = , $|e_t|$ =

e) I_{approx} = , $|e_t|$ =

f) I_{approx} = , $|e_t|$ =

g) I_{approx} = , $|e_t|$ =

a

```
f = @(x) exp(x);
a = -1; b = 1;
Itrue = integral(f, a, b)
```

```
Itrue =
2.3504
```

b

```
Itrap1 = trap(f, a, b, 1)
```

```
Itrap1 =
3.0862
```

```
errtrap1 = round(abs((Itrue - Itrap1) / Itrue) * 100, 4)
```

```
errtrap1 =
31.3035
```

c

```
Itrap4 = trap(f, a, b, 4)
```

```
Itrap4 =
2.3992
```

```
errtrap4 = round(abs((Itrue - Itrap4) / Itrue) * 100, 4)
```

```
errtrap4 =
2.0747
```

d

```
% we are actually applying this to the whole thing, while it can be only
% applied to a range of panels (ns)
ns = 1;
Is13 = 0;
for i = 1:ns
    h = (b - a) / (2*ns);
    x0 = (2*(i-1)*h) + a;
    simpson13 = h/3 * (f(x0) + (4*f(x0 + h)) + f(x0 + 2*h));
    Is13 = Is13 + simpson13;
end
round(Is13, 4)
```

```
ans =
2.3621
```

```
errs13 = round(abs((Itrue - Is13) / Itrue) * 100, 4)
```

```
errs13 =
0.4957
```

e

```
% we are actually applying this to the whole thing, while it can be only
% applied to a range of panels (ns)
ns = 4;
Is13 = 0;
for i = 1:ns
    h = (b - a) / (2*ns);
    x0 = (2*(i-1)*h) + a;
    simpson13 = h/3 * (f(x0) + (4*f(x0 + h)) + f(x0 + 2*h));
    Is13 = Is13 + simpson13;
end
round(Is13, 4)
```

```
ans =
2.3505
```

```
errs13 = round(abs((Itrue - Is13) / Itrue) * 100, 4)
```

```
errs13 =
0.0022
```

f

```
% we are actually applying this to the whole thing, while it can be only
% applied to a range of panels (ns)
ns = 1;
Is38 = 0;
for i = 1:ns
    h = (b - a) / (3*ns);
    x0 = (3*(i-1)*h) + a;
    simpson38 = ((3*h)/8) * (f(x0) + (3*f(x0 + h)) + (3*f(x0 + (2*h))) + f(x0 + 3*h));
    Is38 = Is38 + simpson38;
end
round(Is38, 4)
```

```
ans =
2.3556
```

```
errs38 = round(abs((Itrue - Is38) / Itrue) * 100, 4)
```

```
errs38 =
0.2232
```

g

```
% we are actually applying this to the whole thing, while it can be only
% applied to a range of panels (ns)
ns = 4;
Is38 = 0;
for i = 1:ns
    h = (b - a) / (3*ns);
    x0 = (3*(i-1)*h) + a;
    simpson38 = ((3*h)/8) * (f(x0) + (3*f(x0 + h)) + (3*f(x0 + (2*h))) + f(x0 + 3*h));
    Is38 = Is38 + simpson38;
end
round(Is38, 4)
```

```
ans =
2.3504
```

```
errs38 = round(abs((Itrue - Is38) / Itrue) * 100, 4)
```

```
errs38 =
1.0000e-03
```