MATH 156 LAB 3

BYUNG DO PARK

Topic 1: Comparing sums with increasing number of subintervals.

We come back to the example of velocity-distance travelled from Laboratory 2. The velocity function was

```
> f := t->-22108.111351860603123*t^12+267716.97873114933992*t^7
  -28550.588718383953421*t^4-.69089471139707261503e-1*t^18
  -593.25933891402415755*t^2+88214.018798090967877*t^5
  -182944.24328666353298*t^6+5693.6391073166057753*t^3
  -285572.24388626189800*t^8+227046.45357947547549*t^9
  -136597.49693854806554*t^10+62764.645490312705181*
  t^11+1.5172197744989322043*t^17+5957.3207937003999701*t^13
  -1217.1081971634308471*t^14+185.14666942246783552*t^15
  -20.307901075143421783*t^16+.14466922546287623311e-2*
  t^19+33.706872407075593812*t+20;Digits:=20;
```

$$f := t \to -22108.111351860603123\, t^{12} + 2.6771697873114933992\; 10^5\, t^7 - 28550.588718383953421\, t^4 - 0.069089471139707261503\, t^{18} - 593.25933891402415755\, t^2 + 88214.018798090967877\, t^5 - 1.8294424328666353298\; 10^5\, t^6 + 5693.6391073166057753\, t^3 - 2.8557224388626189800\; 10^5\, t^8 + 2.2704645357947547549\; 10^5\, t^9 - 1.3659749693854806554\; 10^5\, t^{10} + 62764.645490312705181\, t^{11} + 1.5172197744989322043\, t^{17} + 5957.3207937003999701\, t^{13} - 1217.1081971634308471\, t^{14} + 185.14666942246783552\, t^{15} - 20.307901075143421783\, t^{16} + 0.0014466922546287623311\, t^{19} + 33.706872407075593812\, t + 20$$

$$Digits := 20$$

We are working on the interval [0, 5].

Write commands that show the left-hands sums with 5 and 10 subintervals simultaneously. Explain why LHS(5) is less than LHS(10). Graph and explain why LHS(10) is less than LHS(20). Graph and explain why RHS(5)>RHS(10)>RHS(20). Do not forget the plots and student packages.

```
> with(student);
```

[D, *Diff*, *Doubleint*, *Int*, *Limit*, *Lineint*, *Product*, *Sum*, *Tripleint*, *changevar*, *completesquare*, *distance*, *equate*, *integrand*, *intercept*, *intparts*, *leftbox*, *leftsum*, *makeproc*, *middlebox*, *middlesum*, *midpoint*, *powsubs*, *rightbox*, *rightsum*, *showtangent*, *simpson*, *slope*, *summand*, *trapezoid*]

```
> lhs5:=leftbox(f(t),t=0..5,5);
```

lhs5 := *PLOT*(...)

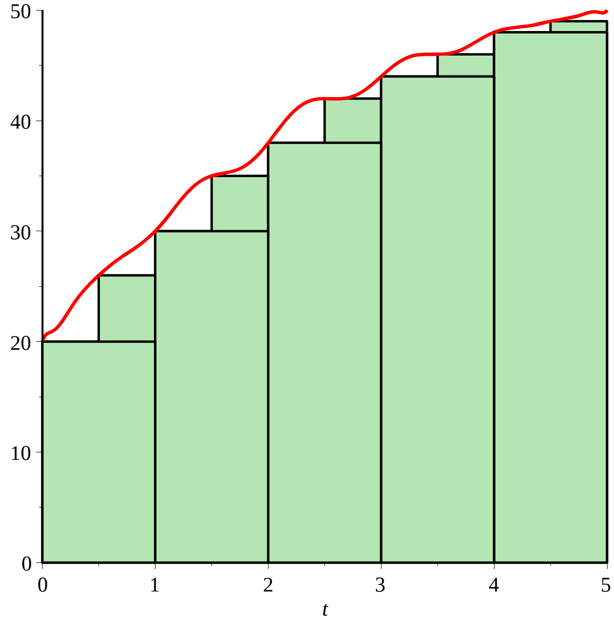
```
> lhs10:=leftbox(f(t),t=0..5,10);
```

lhs10 := *PLOT*(...)

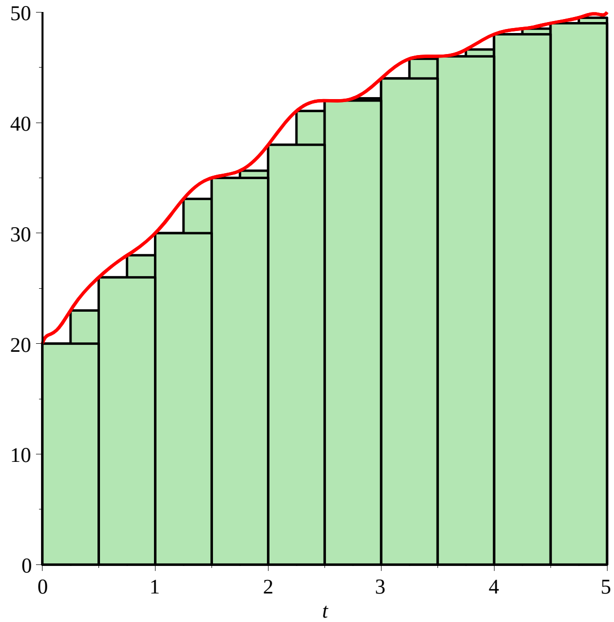
```
> with(plots);
```

[*animate*, *animate3d*, *animatecurve*, *arrow*, *changecoords*, *complexplot*, *complexplot3d*, *conformal*, *conformal3d*, *contourplot*, *contourplot3d*, *coordplot*, *coordplot3d*, *densityplot*, *display*, *dualaxisplot*, *fieldplot*, *fieldplot3d*, *gradplot*, *gradplot3d*, *implicitplot*, *implicitplot3d*, *inequal*, *interactive*, *interactiveparams*, *intersectplot*, *listcontplot*, *listcontplot3d*, *listdensityplot*, *listplot*, *listplot3d*, *loglogplot*, *logplot*, *matrixplot*, *multiple*, *odeplot*, *pareto*, *plotcompare*, *pointplot*, *pointplot3d*, *polarplot*, *polygonplot*, *polygonplot3d*, *polyhedra_supported*, *polyhedraplot*, *rootlocus*, *semilogplot*, *setcolors*, *setoptions*, *setoptions3d*, *spacecurve*, *sparsematrixplot*, *surfdata*, *textplot*, *textplot3d*, *tubeplot*]

```
> display(lhs5, lhs10);
```

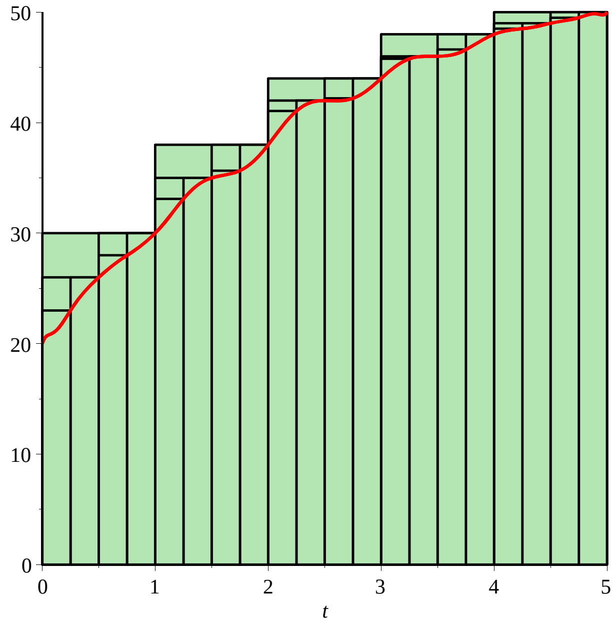



```
> lhs10:=leftbox(f(t),t=0..5,10): lhs20:=leftbox(f(t),t=0..5,20):
> display(lhs10, lhs20);
```



> **rhs5 := rightbox(f(t), t = 0 ..5, 5) :** *rhs10* **:= rightbox(f(t), t = 0 ..5, 10) : rhs20 := rightbox(f(t), t = 0 ..5, 20) :**

> **display(rhs20, rhs10, rhs5);**



>

Topic 2: Evaluate the sums numerically.

We would like now to calculate the sums numerically. Maple has the commands: leftsum(function, variable= lower limit .. upperlimit, number of subintervals) for the left-hand sum and rightsum(function, variable=lower limit ..upper limit, number of subintervals). We will use first 5 subintervals.

```
> leftsum(f(t), t=0..5, 5); rightsum(f(t), t=0..5, 5);
```

$$\sum_{i=0}^{4} \Big(-22108.111351860603123\, i^{12} + 2.6771697873114933992\, 10^{5}\, i^{7} - 28550.588718383953421\, i^{4} - 0.069089471139707261503\, i^{18} - 593.25933891402415755\, i^{2} + 88214.018798090967877\, i^{5} - 1.8294424328666353298\, 10^{5}\, i^{6} + 5693.6391073166057753\, i^{3}$$

$- 2.8557224388626189800\ 10^5\ i^8 + 2.2704645357947547549\ 10^5\ i^9$

$- 1.3659749693854806554\ 10^5\ i^{10} + 62764.645490312705181\ i^{11}$

$+ 1.5172197744989322043\ i^{17} + 5957.3207937003999701\ i^{13}$

$- 1217.1081971634308471\ i^{14} + 185.14666942246783552\ i^{15}$

$- 20.307901075143421783\ i^{16} + 0.0014466922546287623311\ i^{19}$

$+ 33.706872407075593812\ i + 20)$

$$\sum_{i=1}^{5} (-22108.111351860603123\ i^{12} + 2.6771697873114933992\ 10^5\ i^7$$

$- 28550.588718383953421\ i^4 - 0.069089471139707261503\ i^{18}$

$- 593.25933891402415755\ i^2 + 88214.018798090967877\ i^5$

$- 1.8294424328666353298\ 10^5\ i^6 + 5693.6391073166057753\ i^3$

$- 2.8557224388626189800\ 10^5\ i^8 + 2.2704645357947547549\ 10^5\ i^9$

$- 1.3659749693854806554\ 10^5\ i^{10} + 62764.645490312705181\ i^{11}$

$+ 1.5172197744989322043\ i^{17} + 5957.3207937003999701\ i^{13}$

$- 1217.1081971634308471\ i^{14} + 185.14666942246783552\ i^{15}$

$- 20.307901075143421783\ i^{16} + 0.0014466922546287623311\ i^{19}$

$+ 33.706872407075593812\ i + 20)$

As you see Maple does not give us the numerical value, so we use evalf:

```
> evalf(leftsum(f(t), t=0..5, 5)); evalf(rightsum(f(t), t=0..5, 5))
  ;
```

180.00000005459568487

210.00000109957365393

Write commands to compute the left-hand sums and the right-hand sums with 10, 20, 40, 80, 160 subintervals.

```
> <<subintervals|leftsum|rightsum>, <10|evalf(leftsum(f(t), t=0..5,
  10))|evalf(rightsum(f(t), t=0..5, 10))>,<20|evalf(leftsum(f(t),
  t=0..5, 20))|evalf(rightsum(f(t), t=0..5, 20))>,<40|evalf(leftsum
  (f(t), t=0..5, 40))|evalf(rightsum(f(t), t=0..5, 40))>,<80|evalf
  (leftsum(f(t), t=0..5, 80))|evalf(rightsum(f(t), t=0..5, 80))>,
  <160|evalf(leftsum(f(t), t=0..5, 160))|evalf(rightsum(f(t), t=0.
  .5, 160))>>;
```

subintervals	leftsum	rightsum
10	189.00000016386502650	204.00000069155401103
20	192.85223682312342253	200.35223710462791479
40	194.75495599170545161	198.50495609839019775
80	195.70568235029533812	197.58068242212771118
160	196.17832082556839381	197.11582084216714284

```
>
```

EXERCISE: Do this over using loops command of MAPLE.

We see that indeed the left-hand sums are increasing and the right-hand sums are decreasing. At the same time the right-hand sums are always larger than all the left-hand sums.

As you see we have not gotten great accuracy. We do not even have accuracy to the nearest integer. To compute more accurately it will be nice to increase the number of subintervals. As it is tiring to write the same commands over and over, it is important to introduce loops in Maple.

Topic 3: Loops and convergence of Riemann sums.

The following commands produce a table of the left-hand sums with 5, 10, 20, 40, 80, 160, 320, 640, 1280, 2560, 5120 subintervals. Notice that every time we multiply the number of subintervals by 2, so at the first step (j=1) we have 5*2^0 subintervals, at the second step (j=2) we have 5*2^1=10 subintervals, at the third step (j=3) we have 5*2^2=20 subintervals etc. This gives the formula 5*2^{j-1} in the commands.

```
> for j from 1 to 11 do n:= 5*2^(j-1): evalf(leftsum(f(t), t=0..5,
  n)):od;
```

$n := 5$

180.00000005459568487

$n := 10$

189.00000016386502650

$n := 20$

192.85223682312342253

$n := 40$

194.75495599170545161

$n := 80$

195.70568235029533812

$n := 160$

196.17832082556839381

$n := 320$

196.41370793522434534

$n := 640$

196.53115103341478632

$n := 1280$

196.58980890462175191

$n := 2560$

196.61912174504013779

$n := 5120$

196.63377391351571576

We see that the first decimal digit is stabilized. Morever, as you have noticed before, the left-hand sums are increasing. To be certain that the first decimal digit is indeed 6 we need to give overestimates, which are the right-hand sums.

Write commands that produce the right-hand sums with the same number of subintervals, using a loop. All you have to do is copy the command for the left-hand sums and change leftsum into rightsum.

```
> for j from 1 to 11 do n:= 5*2^(j-1): evalf(rightsum(f(t), t=0..5,
  n)):od;
```

$n := 5$

210.00000109957365393

$n := 10$

204.00000069155401103

$n := 20$

200.35223710462791479

$n := 40$

198.50495609839019775

$n := 80$

197.58068242212771118

$n := 160$

197.11582084216714284

$n := 320$

196.88245794942821986

$n := 640$

196.76552604162299702

$n := 1280$

196.70699640951867406

$n := 2560$

196.67771549720066391

$n := 5120$

196.66307078876143340

Now let us introduce the command that evaluates the integral exactly. To introduce $\int_a^b f(t)\,dt$ we use the command Int(f(t), t= a..b)

Topic 4: The Int command.

```
> integraloff:=Int(f(t), t=0..5);
```

$$integraloff := \int_0^5 \left(-22108.111351860603123\, t^{12} + 2.6771697873114933992\; 10^5\, t^7 - 28550.588718383953421\, t^4 - 0.069089471139707261503\, t^{18} - 593.25933891402415755\, t^2 + 88214.018798090967877\, t^5 - 1.8294424328666353298\; 10^5\, t^6 + 5693.6391073166057753\, t^3 - 2.8557224388626189800\; 10^5\, t^8 + 2.2704645357947547549\; 10^5\, t^9 - 1.3659749693854806554\; 10^5\, t^{10} + 62764.645490312705181\, t^{11} + 1.5172197744989322043\, t^{17} + 5957.3207937003999701\, t^{13} - 1217.1081971634308471\, t^{14} + 185.14666942246783552\, t^{15} - 20.307901075143421783\, t^{16} + 0.0014466922546287623311\, t^{19} + 33.706872407075593812\, t + 20 \right) dt$$

To compute the numerical value we use the value command.

```
> value(integraloff);
```

196.64842394482687587

Now we will work with the function g(x)=1/x and the integral

$$\int_1^2 \frac{1}{x}\,dx\,.$$

Compute the left-hand sums and right-hand sums with 2, 4, 8, 16, 32, 64, 128, 256, 516, 1024 subintervals. Use a loop. How many decimal digits are you certain of?
Which are overestimates and which are underestimates and why? Do the left-hand sums form an increasing or decreasing sequence of numbers, as the number of subintervals increase? Show this in appropriate graphs.

```
> g:=x->1/x;
```

$$g := x \rightarrow \frac{1}{x}$$

```
> for j from 1 to 10 do n:= 2^j: evalf(rightsum(f(t), t=1..2, n))
  :od;
```

$n := 2$

36.500000000006854996

$n := 4$

35.439371273841679732

$n := 8$

34.928516774858730958

$n := 16$

34.676026047532648644

$n := 32$

34.550416343070793447

$n := 64$

34.487764713503230358

$n := 128$

34.456476855670501323

$n := 256$

34.440842394306248022

$n := 512$

34.433027529158495156

$n := 1024$

34.429120687883635395

```
> for j from 1 to 10 do n:= 2^j: evalf(leftsum(f(t), t=1..2, n))
  :od;
```

$n := 2$

32.500000000000266416

$n := 4$

33.439371273838385440

$n := 8$

33.928516774857083812

$n := 16$

34.176026047531865854

$n := 32$

34.300416343070402050

$n := 64$

34.362764713503034659

$n := 128$

34.393976855670403585

$n := 256$

34.409592394306199154

$n := 512$

34.417402529158470721

$n := 1024$

34.421308187883623177

```
> for j from 1 to 10 do n:= 2^j: display(leftbox(f(t), t=1..2, n),
  rightbox(f(t), t=1..2, n)):od;
```

$n := 2$