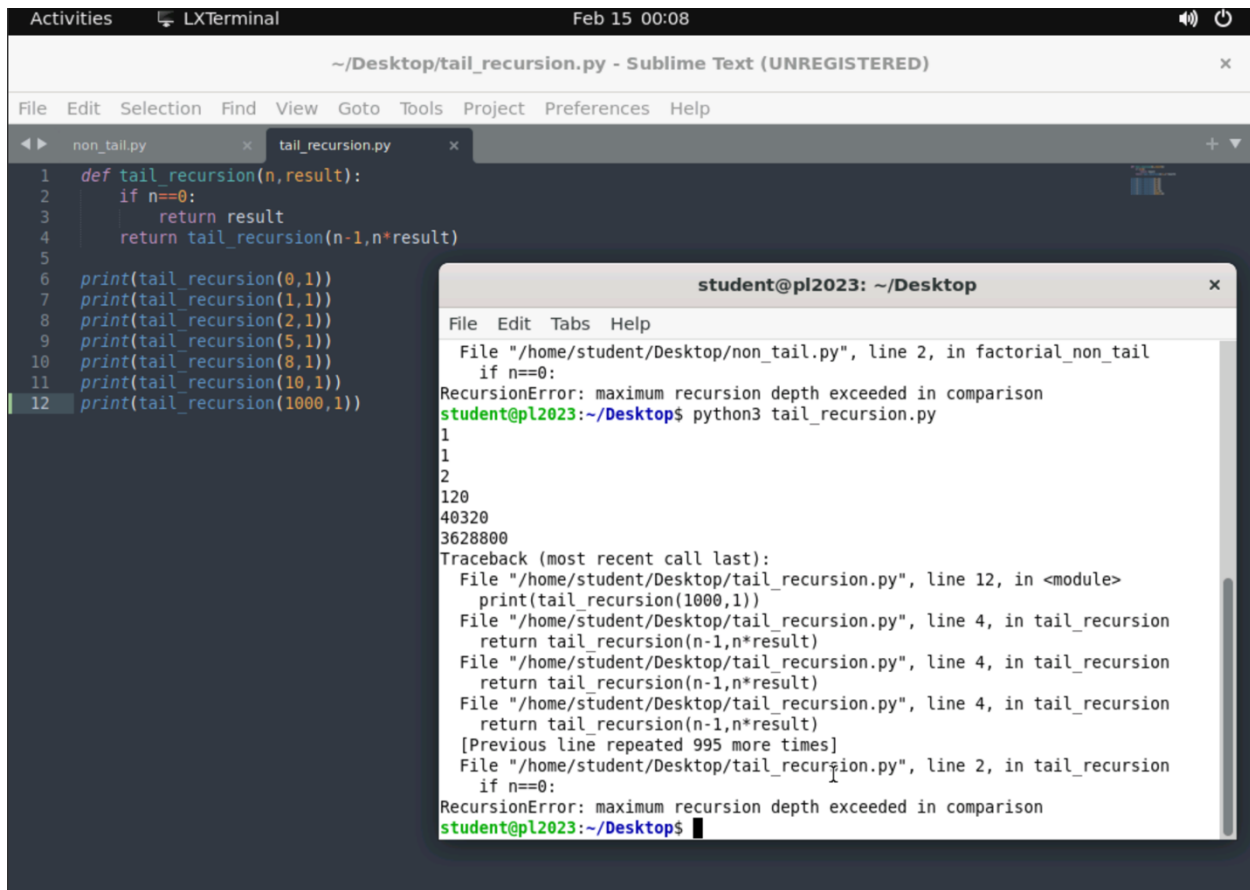


Tail Recursion

In SBCL tail-recursive and non-tail-recursive version were able to handle larger inputs as compared to Python because in python we eventually hit system limits on the stack size which depicts that SBCL supports tail recursive optimization whereas Python doesn't. However, in SBCL when a large number is given as input the non-tail version leads to stack-overflow error first because each recursive call consumes additional stack space to maintain context until its recursive call is resolved. In contrast, a tail recursive function minimizes stack usage by reusing the stack frame for successive calls.

Python

Tail recursion



The screenshot shows a Linux desktop environment with a terminal window. The terminal has a title bar that says "Activities", "LXTerminal", and "Feb 15 00:08". The main window is titled "~ /Desktop/tail_recursion.py - Sublime Text (UNREGISTERED)". It contains two tabs: "non_tail.py" and "tail_recursion.py". The "tail_recursion.py" tab is active, showing the following Python code:

```
1 def tail_recursion(n,result):
2     if n==0:
3         return result
4     return tail_recursion(n-1,n*result)
5
6 print(tail_recursion(0,1))
7 print(tail_recursion(1,1))
8 print(tail_recursion(2,1))
9 print(tail_recursion(5,1))
10 print(tail_recursion(8,1))
11 print(tail_recursion(10,1))
12 print(tail_recursion(1000,1))
```

Below the code editor, there is a terminal window titled "student@pl2023: ~/Desktop". It shows the output of running the script:

```
File Edit Tabs Help
File "/home/student/Desktop/non_tail.py", line 2, in factorial_non_tail
if n==0:
RecursionError: maximum recursion depth exceeded in comparison
student@pl2023:~/Desktop$ python3 tail_recursion.py
1
1
2
120
40320
3628800
Traceback (most recent call last):
  File "/home/student/Desktop/tail_recursion.py", line 12, in <module>
    print(tail_recursion(1000,1))
  File "/home/student/Desktop/tail_recursion.py", line 4, in tail_recursion
    return tail_recursion(n-1,n*result)
  File "/home/student/Desktop/tail_recursion.py", line 4, in tail_recursion
    return tail_recursion(n-1,n*result)
  File "/home/student/Desktop/tail_recursion.py", line 4, in tail_recursion
    return tail_recursion(n-1,n*result)
  [Previous line repeated 995 more times]
  File "/home/student/Desktop/tail_recursion.py", line 2, in tail_recursion
    if n==0:
RecursionError: maximum recursion depth exceeded in comparison
student@pl2023:~/Desktop$
```

Non-tail recursion

The screenshot shows a Linux desktop environment with a Sublime Text editor and a terminal window. The Sublime Text editor is open to a file named `non_tail.py` located at `~/Desktop/non_tail.py`. The code in the file is as follows:

```
1 def factorial_non_tail(n):
2     if n==0:
3         return 1
4     return n*factorial_non_tail(n-1)
5
6 print(factorial_non_tail(0))
7 print(factorial_non_tail(1))
8 print(factorial_non_tail(2))
9 print(factorial_non_tail(5))
10 print(factorial_non_tail(8))
11 print(factorial_non_tail(10))
12 print(factorial_non_tail(1000))
```

The terminal window, titled `student@pl2023: ~/Desktop`, shows the following commands and output:

```
student@pl2023:~$ cd Desktop
student@pl2023:~/Desktop$ ls
non_tail.py  tail_recursion.py
student@pl2023:~/Desktop$ python3 non_tail.py
1
1
2
120
40320
3628800
Traceback (most recent call last):
  File "/home/student/Desktop/non_tail.py", line 12, in <module>
    print(factorial_non_tail(1000))
  File "/home/student/Desktop/non_tail.py", line 4, in factorial_non_tail
    return n*factorial_non_tail(n-1)
  File "/home/student/Desktop/non_tail.py", line 4, in factorial_non_tail
    return n*factorial_non_tail(n-1)
  File "/home/student/Desktop/non_tail.py", line 4, in factorial_non_tail
    return n*factorial_non_tail(n-1)
  [Previous line repeated 995 more times]
  File "/home/student/Desktop/non_tail.py", line 2, in factorial_non_tail
    if n==0:
RecursionError: maximum recursion depth exceeded in comparison
student@pl2023:~/Desktop$
```

Lisp

Tail recursion

```
student@pl2023: ~/Desktop
File Edit Tabs Help
GNU nano 6.2 factorial-tail.lisp
(defun factorial-tail (n &optional (acc 1))
  (if (zerop n)
      acc
      (factorial-tail(1- n)(* acc n))))
(print(factorial-tail 0 1))
(print(factorial-tail 1 1))
(print(factorial-tail 2 1))
(print(factorial-tail 5 1))
(print(factorial-tail 8 1))
(print(factorial-tail 10 1))
(print(factorial-tail 1000 1))

[ Read 12 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

```
student@pl2023: ~/Desktop
File Edit Tabs Help
student@pl2023:~/Desktop$ nano factorial-tail.lisp
student@pl2023:~/Desktop$ sbcl --script factorial-tail.lisp

1
1
2
120
40320
3628800
4023872600770937735437024339230039857193748642107146325437999104299385123986290205920442084
8696940480047998861019719605863166687299480855890132382966994459099742450408707375991882362
7727188732519779505950995276120874975462497043601418278094646496291056393887437886487337119
1810458257836478499770124766328898359557354325131853239584630755574091142624174743493475534
2864657661166779739666882029120737914385371958824980812686783837455973174613608537953452422
1586593201928090878297308431392844403281231558611036976801357304216168747609675871348312025
4785893207671691324484262361314125087802080002616831510273418279777047846358681701643650241
5369139828126481021309276124489635992870511496497541990934222156683257208082133318611681155
3615836546984046708975602900950537616475847728421889679646244945160765353408198901385442487
9849599533191017233555566021394503997362807501378376153071277619268490343526252000158885351
4733161170210396817592151090778801939317811419454525722386554146106289218796022383897147608
8506276862967146674697562911234082439208160153780889893964518263243671616762179168909779911
9037540312746222899880051954444142820121873617459926429565817466283029555702990243241531816
1721046583203678690611726015878352075151628422554026517048330422614397428693306169089796848
2590125458327168226458066526769958652682272807075781391858178889652208164348344825993266043
3676601769996128318607883861502794659551311565520360939881806121385586003014356945272242063
4463179746059468257310379008402443243846565724501440282188525247093519062092902313649327349
7565513958720559654228749774011413346962715422845862377387538230483865688976461927383814900
1407673104466402598994902222217659043399018860185665264850617997023561938970178600408118897
2991831102117122984590164192106888438712185564612496079872290851929681937238864261483965738
```

```
student@pl2023: ~/Desktop
File Edit Tabs Help
GNU nano 6.2 factorial-tail.lisp
(defun factorial-tail (n &optional (acc 1))
  (if (zerop n)
      acc
      (factorial-tail(1- n)(* acc n))))
(print (factorial-tail 100000))

[ Read 6 lines ]
G Help      O Write Out  W Where Is  K Cut       T Execute   C Location  M-U Undo
X Exit      R Read File  \ Replace  U Paste     J Justify  / Go To Line M-E Redo
```

```
student@pl2023: ~/Desktop
File Edit Tabs Help
47801433814209909846757607521501169265051181456421855025018184013769076250956601814993199403086528884261
82272323130451995919490543044644920205573769871961590155894027993494118223028371875967119907783155279025
69525909257695064911783363680339300502381586831578957934017553782534867134530012877786670306510335933746
62126761854914513310316296523149785156857928507593134104809606000445103802611546961578233024253367287117
73607212355035708819795804341280203100792174021456249062309906350680593395981377034317675382664169044153
24550870785653971278952497614663085657270646860934927784743712067092776015749656841421797827796325369128
88258940955665438849806826882456602487318259163685807497963084483554364061908178163844636864906921851352
0426266939735676694187626493794209128682196253075361384553489675088407406205397080563709343337297055959
19934795970053377201242314741060473990555220975762522778204304139270785175919909116525414954764808867073
22821875811461482814800745686164591309192807060198605151395351414298409678984311478832597326462030988185
59811008099548739490541826287806276069779990247360569313736105442595209970008483126219592455102578135012
90022057696761268936414721051969205225371011842006053392108786351354853186118612238354224284507041394764
33190410600709817867752758063258903118680540406757795248437332980285155284336500717170426800865162755182
41043390100842336559049726669451765822910434497748416539367770051599757734187546648801632479825742217034
23593871626682626197414370608125712937380456395170300458450396135477532377410082060174431055692077530437
03371021631027386218311786974777439520194482163355440204640763699348557972483831571199272312914487230594
56232658510548396630867224254027308580508719620185959953380168700632900109801774120748515586342773813048
90787620200242134244790257462437855677144357411384529793248934182287769689628790271903512856491498222373
84939039656224462799765459220433192736746319316703649098818441530563964482049225345991088444361325565217
78073860489039917326320142262595027565899327717924599108416859586700982058386548872743719388391741207461
74499623079500774920789216325908101586246144920142212452436433002216235968256360474315795706440652490853
05228163798932460877031889669367852423559612986340477839021641026418126663436226474573182938940904775193
10441299124952077392690851662476821095904911719124549684524204074555566761330545858680633005626362771760
74831803613049274037357397258743274761504894418355451349139700057907343607335324512020447600523219162815
62675385106918983855657105310235750539779331601469255581212676587509842066421705955730784422534735722510
73584002338891695032608869018118450092434353813660124636530621825414964713352184522292531214059474299371
8853183279796437263509039792929869177302367624004538039520529110130002678502645655285525953014348115353
70115202286618765620166458891013269346136163256367280830508512990469222885152083258772975814249748198749
63262749205633309127026011797096728439843127934664596136238900227177517713218215086433114260479230653785
77620311716850367768216790851333339219682300983794257493074762030873843282615287417286327701281233757149
09084983106340687769359606961701576146495814028243880501323008402365564123597221495310969902123461703843
38130057578946736621734918137930716509698157327325273831689460048459609422810580147808902992842927437011
05133223642026695161315141718961035913493398999522468815440951641371109981157528789119034782927290435793
```

Non-tail recursion

```
student@pl2023: ~/Desktop
File Edit Tabs Help
GNU nano 6.2 factorial-non-tail.lisp
(defun factorial (n)
  (if (zerop n)
      1
      (* n (factorial (1- n)))))
(print (factorial 0))
(print (factorial 1))
(print (factorial 2))
(print (factorial 5))
(print (factorial 8))
(print (factorial 10))
(print (factorial 1000))

[ Read 12 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

```
student@pl2023: ~/Desktop
File Edit Tabs Help
student@pl2023:~/Desktop$ ls
factorial-non-tail.lisp factorial-tail.lisp non_tail.py tail_recursion.py
student@pl2023:~/Desktop$ nano factorial-non-tail.lisp
student@pl2023:~/Desktop$ sbcl --script factorial-non-tail.lisp

1
1
2
120
40320
3628800
4023872600770937735437024339230039857193748642107146325437999104299385123986290205920442084
8696940480047998861019719605863166687299480855890132382966994459099742450408707375991882362
7727188732519779505950995276120874975462497043601418278094646496291056393887437886487337119
1810458257836478499770124766328898359557354325131853239584630755574091142624174743493475534
2864657661166779739666882029120737914385371958824980812686783837455973174613608537953452422
1586593201928090878297308431392844403281231558611036976801357304216168747609675871348312025
4785893207671691324484262361314125087802080002616831510273418279777047846358681701643650241
5369139828126481021309276124489635992870511496497541990934222156683257208082133318611681155
3615836546984046708975602900950537616475847728421889679646244945160765353408198901385442487
9849599533191017233555566021394503997362807501378376153071277619268490343526252000158885351
4733161170210396817592151090778801939317811419454525722386554146106289218796022383897147608
8506276862967146674697562911234082439208160153780889893964518263243671616762179168909779911
9037540312746222899880051954444142820121873617459926429565817466283029555702990243241531816
1721046583203678690611726015878352075151628422554026517048330422614397428693306169089796848
2590125458327168226458066526769958652682272807075781391858178889652208164348344825993266043
3676601769996128318607883861502794659551311565520360939881806121385586003014356945272242063
4463179746059468257310379008402443243846565724501440282188525247093519062092902313649327349
7565513958720559654228749774011413346962715422845862377387538230483865688976461927383814900
```



```
student@pl2023: ~/Desktop
File Edit Tabs Help
GNU nano 6.2 factorial-non-tail.lisp
(defun factorial (n)
  (if (zerop n)
      1
      (* n (factorial (1- n)))))
(print (factorial 100000))

I

[ Read 5 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line M-E Redo
```

```
student@pl2023: ~/Desktop
File Edit Tabs Help
student@pl2023:~/Desktop$ nano factorial-non-tail.lisp
student@pl2023:~/Desktop$ sbcl --script factorial-non-tail.lisp
fatal error encountered in SBCL pid 1918 tid 1918:
Control stack exhausted, fault: 0x7fb6876e7ff8, PC: 0x534370ac

0: fp=0x7fb6876e8000 pc=0x534370ac CL-USER::FACTORIAL
1: fp=0x7fb6876e8020 pc=0x53437097 CL-USER::FACTORIAL
2: fp=0x7fb6876e8040 pc=0x534370ae CL-USER::FACTORIAL
3: fp=0x7fb6876e8060 pc=0x534370ae CL-USER::FACTORIAL
4: fp=0x7fb6876e8080 pc=0x534370ae CL-USER::FACTORIAL
5: fp=0x7fb6876e80a0 pc=0x534370ae CL-USER::FACTORIAL
6: fp=0x7fb6876e80c0 pc=0x534370ae CL-USER::FACTORIAL
7: fp=0x7fb6876e80e0 pc=0x534370ae CL-USER::FACTORIAL
8: fp=0x7fb6876e8100 pc=0x534370ae CL-USER::FACTORIAL
9: fp=0x7fb6876e8120 pc=0x534370ae CL-USER::FACTORIAL
10: fp=0x7fb6876e8140 pc=0x534370ae CL-USER::FACTORIAL
11: fp=0x7fb6876e8160 pc=0x534370ae CL-USER::FACTORIAL
12: fp=0x7fb6876e8180 pc=0x534370ae CL-USER::FACTORIAL
13: fp=0x7fb6876e81a0 pc=0x534370ae CL-USER::FACTORIAL
14: fp=0x7fb6876e81c0 pc=0x534370ae CL-USER::FACTORIAL
15: fp=0x7fb6876e81e0 pc=0x534370ae CL-USER::FACTORIAL
16: fp=0x7fb6876e8200 pc=0x534370ae CL-USER::FACTORIAL
17: fp=0x7fb6876e8220 pc=0x534370ae CL-USER::FACTORIAL
18: fp=0x7fb6876e8240 pc=0x534370ae CL-USER::FACTORIAL
19: fp=0x7fb6876e8260 pc=0x534370ae CL-USER::FACTORIAL
20: fp=0x7fb6876e8280 pc=0x534370ae CL-USER::FACTORIAL
21: fp=0x7fb6876e82a0 pc=0x534370ae CL-USER::FACTORIAL
22: fp=0x7fb6876e82c0 pc=0x534370ae CL-USER::FACTORIAL
23: fp=0x7fb6876e82e0 pc=0x534370ae CL-USER::FACTORIAL
24: fp=0x7fb6876e8300 pc=0x534370ae CL-USER::FACTORIAL
25: fp=0x7fb6876e8320 pc=0x534370ae CL-USER::FACTORIAL
26: fp=0x7fb6876e8340 pc=0x534370ae CL-USER::FACTORIAL
27: fp=0x7fb6876e8360 pc=0x534370ae CL-USER::FACTORIAL

I
```