

Re: make 100 by using 1, 7, 7, 7, 7

Re: make 100 by using 1, 7, 7, 7, 7

Source: <http://sci.tech-archive.net/Archive/sci.math/2007-11/msg02715.html>

- *From:* Robert Israel <israel@xx>
 - *Date:* Tue, 13 Nov 2007 14:05:02 -0600
-

Kira Yamato <kirakun@xxxxxxxxxxxxxx> writes:

On 2007-11-13 13:22:13 -0500, Robert Israel
<israel@xx> said:

"Benjamin A. Bartsch" <benjamin.a.bartsch@xxxxxxxxxx> writes:

dangerousgam...@xxxxxxxxxx schrieb:

So, using only +, -, x, /, and parentheses,
and ONLY these numbers:

1, 7, 7, 7, 7

how can you make 100? Is there more than
one solution?

--> $(1/7+7)*(7+7)$

Benjamin

Yes, and that's the only solution (up to commutativity).

I'm very curious how you concluded this. Was it by a program that
tests through all possible combinations?

I tried to write a program too, but I stop when I couldn't quickly find
a way to list out all possible trees with 7 leaves.

Here's my search program, written in Maple.

Re: make 100 by using 1, 7, 7, 7, 7

```
trees:= proc(S)
option remember;
local oper,SL,L,res,ret,V,W,v,w,R,r;
if nops(S) = 1 then {[op(S),op(S)]}
else
res:= {}; ret:= {};
for oper in [^+`,`-`,`^`,`*`] do
for SL in combinat[powerset]({$1..nops(S)}) do
if nops(SL) > 0 and nops(SL) < nops(S) then
L:= sort([seq(S[j],j=SL)]);
R:= sort([seq(S[j],j=({$1..nops(S)} minus SL)]));
V:= trees(L);
W:= trees(R);
for v in V do
for w in W do
if ((oper = `+`) or (oper = `*`)) and (v[1] > w[1]) then next
end if;
try
r:= oper(v[1],w[1]);
catch: next;
end try;
res:= res union {r};
ret:= ret union {[r, [v[2],oper,w[2]]]};
end do
end do
end if
end do
end do;
ret
end if
end proc;
```

```
select(t -> (t[1]=100), T);
```

```
{[100, [[[1, /, 7], +, 7], *, [7, +, 7]]]}
```

—

Robert Israel israel@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Department of Mathematics <http://www.math.ubc.ca/~israel>
University of British Columbia Vancouver, BC, Canada

.