

```

> restart;
> f:=proc(n :: integer) :: integer;
  description "cite how many Goldbach partitions there are for an integer n"
  local a, b, c, d, counter;
  with(numtheory) :
  a := Vector[row](20);
  for b from 2 to 20 do
    a[b] := ithprime(b);
  end do;
  counter := 0;
  for d from 2 to 20 do
    for c from 1 to  $\frac{n}{2}$  do
      if ithprime(d) + a[c + 1] = n then counter := counter + 1; end if;
    end do;
  end do;
  return counter;
end proc;
f:=proc(n::integer)::integer;
  local a, b, c, d, counter;
  description "cite how many Goldbach partitions there are for an integer n";
  with(numtheory);
  a := Vector[row](20);
  for b from 2 to 20 do a[b] := ithprime(b) end do;
  counter := 0;
  for d from 2 to 20 do
    for c to 1/2 * n do
      if ithprime(d) + a[c + 1] = n then counter := counter + 1 end if
    end do
  end do;
  return counter
end proc

```

(1)

```

> f(6)
1
> f(8)
2
> f(10)
3
> for g from 6 to 20 by 2 do
  f(g)
end do;
1
2
3
2
3

```

(2)

(3)

(4)

└─>

4
4
4

(5)