```
(* Project Euler Problem 116 *)
(* A Riemann1337 Production. *)
(* Goal: Partition the integer corresponding to total number of blocks.  Then
  keep partitions that only have 1's and 2's for red squares and 1's and 3's
  for green squares and 1's and 4's for the blue squares.  Store these lists
  in cds2 cds3 and cds4.  Then compute the number of unique permutations of
  each element of these lists.  We do this by tallying the number of elements
  in each candidate list and dividing the order of the full permutation
  group of the list by the product of the factorials of the tally counts. *)

Prod[lst_] := Product[lst[[i]], {i, 1, Length[lst]}]

nmblks = 50;
cds2 = Select[IntegerPartitions[nmblks], Max[#] == 2 && (Min[#] == 1 || Min[#] == 2) &];
cds3 =
  Select[IntegerPartitions[nmblks], Max[#] == 3 && (Length[Position[#, 2]] == 0) &];
cds4 = Select[IntegerPartitions[nmblks],
   Max[#] == 4 && (Length[Position[#, 2]] == 0) && (Length[Position[#, 3]] == 0) &];
ls1 = Sum[Length[cds2[[i]]]! / Prod[Map[Factorial,
      Transpose[Tally[cds2[[i]]]][[2]]]], {i, 1, Length[cds2]}];
ls2 = Sum[Length[cds3[[i]]]! / Prod[Map[Factorial,
      Transpose[Tally[cds3[[i]]]][[2]]]], {i, 1, Length[cds3]}];
ls3 = Sum[Length[cds4[[i]]]! / Prod[Map[Factorial,
      Transpose[Tally[cds4[[i]]]][[2]]]], {i, 1, Length[cds4]}];
ls1 +
 ls2 +
 ls3
```

```
20 492 570 929
```