場所も書いていただきありがとうございます

	最小値	場所(複数の内の一つ)	備考	最大値	場所(複数の内の一つ)	備考
歩	0	1一		☆ 6	5五	と金
香	0	1一		8	9九	
桂	0	1一		☆ 6	5五	成桂
銀	1	1一		☆ 6	5五	成銀
金	2	1一		6	5五	
角	8	1一		☆ 20	5五	馬
飛	16	1一		☆ 20	5五	龍
王	3	1一		8	5五	
計	30			80		

解答:最小値は30、最大値は80

それぞれの駒の動けるマスの数の最大最小なので、合計はなくて大丈夫でした。

☆成り駒で考えていただいたのですね。普通に将棋を指すときは成り駒をいきなり盤上に置くことはないので、その発想はありませんでした。頭やわらかい!

最小：0マス

						銀	王	金
					桂	香	飛	角
							歩	

最大：62マス

	歩		王					
			桂	角	銀			
					金			
						飛		
							香	

問題文で
"盤上に1つだけ駒を置いた時"と
なっているので、いっぱい駒を置くなんて紅おこです
といっても、この解答の盤面はパズルみたいで面白いですし
こんなこと考えたこともなかったので、
座布団5枚くらい差し上げます(￣▽￣)

公益財団法人 日本数学検定協会

パーフェクト！
すばらしいです☺

駒が動ける箇所

	最大値	最小値
王将	8	3
飛車	16	16
角行	16	8
金将	6	2
銀将	5	1
桂馬	2	0
香車	8	0
歩兵	1	0

色もつけていただいてありがとうございます

公益財団法人
日本数学検定協会

Q2

(1) 歩の場合

全部で $9\times 2=18$(枚)存在する。

よって、0, 1, 2, …, 17, 18(枚)のときが考えられ、19通り。

(2) 香、桂、銀、金の場合

全部で $2\times 2=4$(枚)ずつ存在。

よって、それぞれに対して、5通り。

(3) 飛、角の場合

全部で $2\times 1=2$(枚)ずつ存在。

よって、それぞれに対して、3通り。

以上より、駒の数から考えた持ち駒のパターン数は

$19\times 5\times 5\times 5\times 5\times 3\times 3=19\times 5^4\times 3^2=106875$

より、106875通りである。

思考の過程も
きちんと書かれていて
すばらいいです☆☆

19×5×5×5×5×3×3 = $19 \times 5^4 \times 3^2$ = 106875

より 106875 通りである。

そうなんです！これ超大事！

次に実際上で言ったパターンを作り出すことができるかについて考える。そのために、まず、互いの玉以外の駒すべてを持ち駒とすることができることを示す。まず、

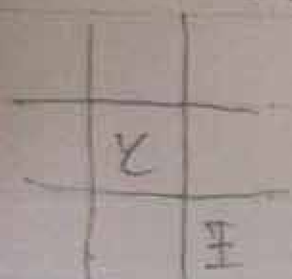
▲5六歩 △5四歩 ▲5八玉 △5二玉 ▲5七玉 △5三玉 ▲4六玉

△6四玉 ▲3五玉 △7五玉 ▲2五玉 △8五玉 ▲1五玉 △9五玉

▲4六歩 △6四歩 ▲4五歩 △6五歩 ▲4四歩 △6六歩

▲4三歩成 △6七歩成　数学の解答に棋譜が！

とし、この過程でできたと金だけを動かし互いの駒を取る。その過程において、王手をかけないように動かすことができることに注意する。そして盤面にと金2枚と互いの玉将だけになったら、まず向いてそれぞれと金をとる。このとき、と金を取る直前は以下のようになるように注意



そして、先手の玉を9九、後手の玉を1一へと移動させる。そして先手の持ち駒の中から2つ任意に選び、それをX、Yとすると、

▲1二X打 △同玉 ▲1一Y打 △同玉とできる。これを14回くり返すと、盤面には互いの玉、そして後手の持ち駒は互いの玉以外のすべての駒となる。

読んでいてテンションの上がる解答でした！

最後に前述した状態から任意の持ち駒のパターンxとすることができることを示す。
まずxを任意の持ち駒のパターンであり、駒の集合とする。
このとき$\bar{x}$はxに含まれない駒の集合とする。
例えば、
x={歩:4枚, 香:1枚, 桂:0枚, 銀:2枚, 金:0枚, 飛:1枚, 角:0枚}
ならば
$\bar{x}$={歩:14枚, 香:1枚, 桂:2枚, 銀:2枚, 金:2枚, 飛:1枚, 角:2枚}
となる。
以下で空であるマスとは駒が存在しないマスのことを表す。
$\bar{x}$の全ての元、$\bar{x}_1$, $\bar{x}_2$, … $\bar{x}_n$に対してそれぞれをk一 [illegible]
(kは1〜8の自然数)の空である任意のマスに打つこと。(※1)　[illegible]
▲8九王 △2一$\bar{x}$, ▲9九王 △3一$\bar{x}$,
のようにすると、互いの王が詰むことなく、後手の持ち駒がxとなる。(※2)
よって以上より、最初に考えたパターンがすべて再現可能であることが示された。
よって全部で106875通り。

証明せよと言われなくてもする姿勢、すばらしいです

(kは1〜8の自然数)の段である任意のマスにすることが可能

▲8k王 △2一玉, ▲9k王 △3一玉

のようにすると、互いの王が詰むことなく後手の指す番で

終了となる。(※2)

よって以上より、最初に考えたパターンすべてで再現

可能であることが示された。

よって全部で 1068775通り。

(※1) このとき、角や飛を打ちこむ際には、王手になったり

この先王手になったりするような所には打ちこまないように

する。つまり、角であれば9一, 9二など飛であれば7一, 7二

に打ちこむようにする。

(※2) 二歩とならないようにするために、

例えばk四には歩だけを打ちこむようにする。そして

玉の中に10枚以上の歩がある場合には

任意の歩(場所はk四)をk一まで進め成れば

再度k四へ持ち駒から打ちこめる。

持ち駒のパターン数を求めよ
→ 自分の持ち駒のパターン数を求めよ
→ 自分および相手の持ち駒のパターン数を求めよ
どちらと取るかで大きく回答が異なる

そうですね…わかりにくくてすみません(汗)

思考の過程がわかりやすい!

前者の場合

① 持ち駒となりうる駒の数を列挙する

歩	18
香車	4
桂馬	4
銀将	4
金将	4
角行	2
飛車	2

② 持ち駒なし、がありうるので、各駒の枚数のパターン数は上記＋1

GJ!

歩	19
香車	5
桂馬	5
銀将	5
金将	5
角行	3
飛車	3

③ 全部かけたら答え

後者の場合 →②までは前者の場合と同一

③ それぞれの駒に対し、自分が取りうる枚数と、そのときの相手が取りうる枚数を考える
歩を自分が10枚とった場合 → 相手が取りうるのは0枚～8枚の9通り
歩を自分が17枚とった場合 → 相手が取りうるのは0枚か1枚、どちらか
ある駒の取りうるパターン数がNパターンあるとき、自分がK個取ったら相手はN-Kパターンの取り方がある
つまり、N個ある1種の駒に対し、相手と自分で採りうるパターン数は1からNまでの総和となる
1からNまでの総和は N*(N+1)/2 で求められる

歩	19	→	19*20/2	190
香車	5	→	5*6/2	15
桂馬	5	→	5*6/2	15
銀将	5	→	5*6/2	15
金将	5	→	5*6/2	15
角行	3	→	3*4/2	6
飛車	3	→	3*4/2	6

考えを一般化できていて、すばらしいです

④ これらを全部かけたら答え

自分のみの持ち駒のパターン数を数えた場合	自分と相手の持ち駒のパターン数の組み合わせすべて数えた場合
答え：106875通り	答え：346275000

自分で考えた問題なのになんですけど、持ち駒のパターン数ってめっちゃ多いですよね(しみじみ)

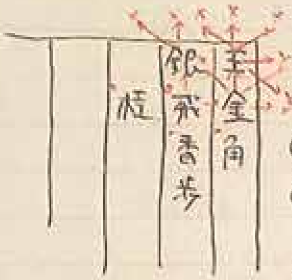
竹俣紅からの挑戦

Q1

最小

各駒の進行方向の数が多い順に角から埋めていく

王8 → 金6 → 銀5 → 飛4 → 角4 →
桂2 → 香1 → 歩1



①「王」は1-5の角に
②「金」「銀」は「王」の左下に
③「飛」は王の左下
④「角」は端っこ
でも「金」>「銀」より右の方
⑤「桂」は1or2行目 香「歩」は下に

→ 最小値は 0

最大

各駒の動ける最大マス数順に埋める

角16 → 飛16(横) → 王8 → 金6 → 銀5 →
香8
桂2 → 歩1

①「角」は中心
②「角」で分断されたところに
「王」>「金」>「銀」を
利用して配置
③「銀」のサポート終わり
空いたところに「飛」
④「桂」の右列に香
⑤ ラスト一口「歩」

ダブりなし 4 8 8 8 4 8 5 4 5 = 54
ダブりあり 4 10 11 9 4 9 5 5 5 = 62 ← 最大値

問題文で"盤上に1つだけ駒を置いた時"となっているのでいっぱい駒を置くなんて紅おこです💢 ですが、思考の過程がわかりやすくて良い解答ですね😊

Q2 コマの枚数は

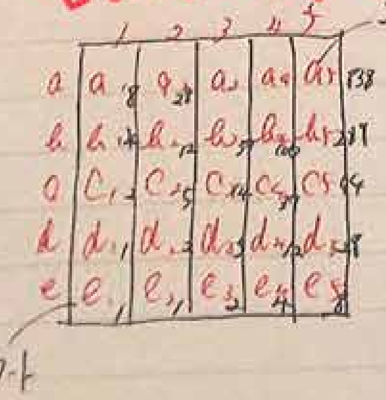
歩18枚、香桂銀金 各4枚、角飛 各2枚
↓ ↓ ↓

コマをもつ可能性は 0〜18枚の19通り　0〜4枚の5通り　0〜2枚の3通り

$\Rightarrow 19^1 \times 5^4 \times 3^2 = 106{,}875$ 通り

持ち駒なしの可能性もきちんと含めていますね。GJ!

Q3



飛の駒は一回につき上または右方向に1〜4マスを自由にすすめる
ここでスタートに駒をおくことを1通り($e_1 = 1$)とすると、
各のマスへ行く通りは そのマスの左の総和と下のマスの総和の合計となるため

838通り

そう!
これがわかればカンタン!

Q3

丁寧な場合分けですね♡ 地道な作業ですがきれいに整理されていて良いですね☆

(i) 全て1マスずつ動く場合

$_8C_4 = \frac{8\cdot7\cdot6\cdot5}{4\cdot3\cdot2\cdot1} = 70$

(ii) 1回だけ2マス動く場合

$2\times\frac{7!}{4!2!1!} = \frac{7\cdot6\cdot5\cdot4\cdot3\cdot2\cdot1}{4\cdot3\cdot2\cdot1\cdot2\cdot1\cdot1}\times2$

$= 210$

(iii) 1回だけ3マス動く場合

$2\times\frac{6!}{4!1!1!} = \frac{6\cdot5\cdot4!\times2}{4!\cdot1!\cdot1!} = 30\times2$

$= 60$

(iv) 1回だけ4マス動く場合

$2\times\frac{5!}{4!} = 5\times2$

$= 10$

(v) 同方向に2マスを2回動く場合

$2\times\frac{6!}{4!2!} = \frac{6\cdot5\cdot2}{2} = 15\cdot2$

$= 30$

(vi) 別方向に2マスを2回動く場合

$\frac{6!}{2!2!} = \frac{6\cdot5\cdot4\cdot3}{2} = 180$

(vii) 3回2マス動く場合

$2\times\frac{5!}{2!2!} = \frac{5\cdot4\cdot3}{2\cdot1}\times2$

$= 60$

(viii) 全て2マスずつ動く場合

$\frac{4!}{2!2!} = \frac{4\cdot3}{2\cdot1} = 6$

(viv) 3マス1回2マス1回の場合

$2\times\frac{5!}{2!} = 120$

(x) 3マス1回 2マス2回の場合

$2\times\frac{4!}{2!} = 24$

(xi) 3マス2回の場合

$4! = 24$

(xii) 4マス1回2マス1回の場合

$2\times\frac{4!}{2!} = 24$

(xiii) 4マス1回2マス2回の場合

$2\times\frac{3!}{2!} = 6$

(xiv) 4マス1回3マス1回の場合

$2\times3! = 12$

(xv) 4マス2回の場合

$2! = 2$

(i) ～ (xv) より

838 (通り)

n \ m	1	2	3	4	5
1	a_{11}	a_{21}	a_{31}	a_{41}	a_{51}
2	a_{12}				
3	a_{13}				
4	a_{14}				
5	a_{15}				a_{55}

とすると

a_{mn}に行く方法は

$$\begin{cases} a_{11}=1,\ a_{21}=1,\ a_{12}=1 \\ a_{mn}=\sum_{k=1}^{n} a_{mk}+\sum_{\ell=1}^{m} a_{\ell n} \quad (1\leq m,\ell\leq 5) \end{cases}$$

通りある

そうなんです！
それがわかれば
単純明快✧

これを表にすると

1	1	2	4	8
1	2	5	12	28
2	5	14	37	94
4	12	37	106	289
8	28	94	289	838

838通り

わっ！なんじゃこれ笑

```
%% egglang
%%
%% 賢者からの挑戦 http://www.math4life.jp/
%% 竹俣紅 Q3.

-module(m).
-compile(export_all).

main() ->
    beni3(create_matrix(5, 5)).

put_and_print({X,Y}, Val) ->
    put({X,Y}, Val),
    io:format("X: ~p, Y: ~p Value: ~p ~n",[X, Y, Val]).

-spec create_matrix(integer(), integer()) -> list().
create_matrix(XNum, YNum) ->
    Mat = [{X,Y} ||  X <- lists:seq(1, XNum), Y <- lists:seq(1, YNum)],
    [put(T, 0) || T <- Mat], % put initial values into a process dictionary.
    Mat.

-spec beni3(list()) -> integer().
beni3(L) -> beni3(L, 0).

-spec beni3(list(), integer()) -> integer().
beni3([], Val) -> Val;
beni3([{1,1}|T], Val) -> put_and_print({1,1}, 1), beni3(T, Val);
beni3([{2,1}|T], Val) -> put_and_print({2,1}, 1), beni3(T, Val);
beni3([{1,2}|T], Val) -> put_and_print({1,2}, 1), beni3(T, Val);
beni3([{X,Y}|T], _) ->
    MyVal = lists:sum([get({IX,Y}) || IX <- lists:seq(1,X)]) + lists:sum([get({X,IY}) || IY <- lists:seq(1,Y)]),
    put_and_print({X,Y}, MyVal),
    beni3(T, MyVal).
```

```
math — beam.smp —...
1> c(m).
{ok,m}
2> m:main().
X: 1, Y: 1 Value: 1
X: 1, Y: 2 Value: 1
X: 1, Y: 3 Value: 2
X: 1, Y: 4 Value: 4
X: 1, Y: 5 Value: 8
X: 2, Y: 1 Value: 1
X: 2, Y: 2 Value: 2
X: 2, Y: 3 Value: 5
X: 2, Y: 4 Value: 12
X: 2, Y: 5 Value: 28
X: 3, Y: 1 Value: 2
X: 3, Y: 2 Value: 5
X: 3, Y: 3 Value: 14
X: 3, Y: 4 Value: 37
X: 3, Y: 5 Value: 94
X: 4, Y: 1 Value: 4
X: 4, Y: 2 Value: 12
X: 4, Y: 3 Value: 37
X: 4, Y: 4 Value: 106
X: 4, Y: 5 Value: 289
X: 5, Y: 1 Value: 8
X: 5, Y: 2 Value: 28
X: 5, Y: 3 Value: 94
X: 5, Y: 4 Value: 289
X: 5, Y: 5 Value: 838
838
3>
```

何が起きているのかよくわかりませんが、ありがとうございます。
将来自分が母になって、子どもに「ママ〜宿題教えて」とこれを持って来られたらどうしようと心配になる紅です。

ここが答えですよね…
正解です！