

第2問

解説

問1 アイ ②⑤ ウ ②

エ ① オ ⑤ (エとオは順不同)

ア、イ 配列 **Hikareru** と配列 **Hiku** の要素はどちらも5個であり、図1のプログラムを実行すると、変数 **i** を0から4まで1ずつ増やしながら、変数 **i** の値のそれぞれについて、変数 **j** を0から4まで1ずつ増やしながら式を表示する処理を繰り返すので、式は全部で $5 \times 5 = 25$ (個)

表示される。よって、**アイ**は②⑤となる。

ウ 図1のプログラムを実行して表示される式は、順に

1-1, 1-3, 1-5, 1-7, 1-8,
2-1, 2-3, 2-5, 2-7, 2-8,
5-1, 5-3, 5-5, 5-7, 5-8,
6-1, 6-3, 6-5, 6-7, 6-8,
9-1, 9-3, 9-5, 9-7, 9-8

である。「5-3」の式は「5-1」の式の次に表示される。よって、**ウ**は②となる。

エ、オ 図2のプログラムで表示するのは計算結果が負の数にならない引き算の式である。つまり、「引かれる数」が「引く数」より大きい式か、等しい式のどちらかである。図2のプログラムでは、「引かれる数」は **Hikareru[i]**、「引く数」は **Hiku[j]** であるから、(6) 行目の条件の一つは、**Hikareru[i] >= Hiku[j]** である。また、変数 **kaisu** は式を表示した回数を表す変数であり、図2のプログラムにおいて、(7) 行目で式を1個表示すると、(8) 行目で **kaisu** は1だけ増える。つまり、式を10個表示したあと(8) 行目の処理により **kaisu** の値は10になる。**kaisu** が0から9まで増えるとき、式は10個表示されるから、式を10個まで表示させる場合、**kaisu** の値が10のときに(7) 行目が実行されないようにすればよいので、(6) 行目の条件の一つは、**kaisu < 10** である。よって、**エ**、**オ**は①と⑤である。

問2 カ ① キ ②

カ 図3のプログラムにおいて、変数 **p**、**q** は配列 **Hikareru** の添字(要素の番号)である。また、関数「**乱数**(最初の数, 最後の数)」は、引数として「最初の数」と「最後の数」が与えられ、最初のを最小値、最後の数を最大値とする範囲における整数が戻り値となる関数である。つまり、関数「**乱数**(最初の数, 最後の数)」の引数は、配列 **Hikareru** の添字である0以上4以下の整数に対応する。よって、**カ**は①となる。

キ **[9,6,5,2,1]** と **[1,2,5,6,9]** の要素を見比べると、**1, 2, 6, 9** の4個が入れ替わっているので、関数「**入れ替え**(一つ目の要素,

◀ **i=0** に対し、**j=0, 1, ..., 4** の五つが対応し、**i=1, 2, 3, 4** に対しても、**j=0, 1, ..., 4** の五つがそれぞれ対応する。

◀ **i=0, j=0, 1, ..., 4**

◀ **i=1, j=0, 1, ..., 4**

◀ **i=2, j=0, 1, ..., 4**

◀ **i=3, j=0, 1, ..., 4**

◀ **i=4, j=0, 1, ..., 4**

◀ 引き算の式は
「引かれる数」-「引く数」
で表されている。

◀ 「**kaisu = kaisu + 1**」により、**kaisu** の値は1だけ増える。