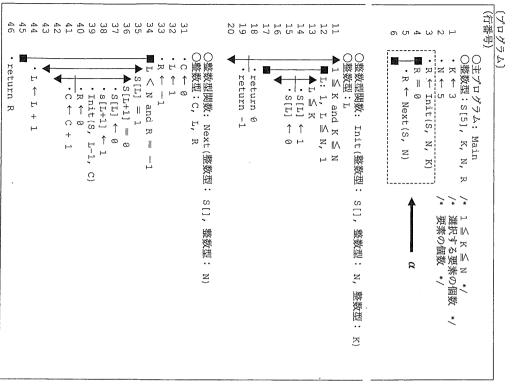
四 四 四 組合す વા -夕構造及びアルゴリズム)

(H23 春-FE 午後間 8)

[解]

b−±, C d - γ , e-√, f-γ, g-

N個の要素中からK個の要素を選ぶ組合せをすべて求めるアルゴリズムの問題である。プログラムがどのように処理を行っているかを問う内容であり、プログラムをトレースする力、関数の引数と返却値に関する知識などが必要である。〔プログラムの説明〕だけでは配列sの扱いや各関数のアルゴリズムを把握しにくく、利用部門に所属する受験者にとっては難しく感じられたと思われる。プログラムの動きを理解するために、与えられたデータを使ってトレースしたり、設問中の記述や解答群を参考に考察したりする必要があり、難易度は、やや高いといえる。



関数 Init 及び Next の処理の概要を確認する。 最初に (プロ グラ ムの説明) [プロ グラム]の内容を基に, 主プログラム Main,

主プログラム Main は、行番号 3 で関数 Init を呼び出す。このとき、関数 Init には引数 (関数に渡す値) として配列 S と変数 N に 5、変数 K に 3 が渡される。関数 Init の処理が終わると、変数 R に関数 Init の返却値 (関数 Init が Return する値) をセットする。行番号 5 の関数 Next を呼び出す処理は、変数 R の値が 0 である間繰り返される。 関数 Next の引数は配列 S と変数 N の値であり、 関数 Next の処理が終わると変数 R に関数 Next の返却値をセットする。 関数 Init は、1 \le K \le K \le M \ge M

0を返して関数を終了する。

Ω	
1	ш
1	2
1	မ
0	4
0	σı

めて配列 S に設定し、返却値として 0 を返す。具体的には、番号 1 から順に調べ、S[L]=1 (行番号 35)、S[L+1]=6 その値を入れ替える(行番号 37、38)。例えば、1 回目に1の要素番号 1~5 の内容は次の①のようになっているので、=1、S[L+1]=8 (連続する 2 要素の値の組合せが 1 と 0 の入替えが行われ, 関数 Next は, 渡された配列Sに対して所定の操作を行い,次の組合せの状態を記し、返却値として0を返す。具体的には,受け取った配列Sを要調べ,S[I]=1 (行番号 35)。 [L+1]=6 (行番号 36) の場合に, (連続する2要素の値の組合せが1と 配列Sは②のようになる。 1回目に呼び出された際は,配列 s いるので,変数 L=3のときに S[L] 0 になり, 次の組合せの状態を求 3 要素

	(0)		Θ	
	ß		ß	
	יי	_	1	ᆫ
-	ы	12	1	2
	0	ယ	1	ယ
	н	4	0	4
	0	Oī	0	οı

でいるめる。 となが1と でなる。ま で、10場 のため。関 で、2)。関数 で、2)。 続いて、関数 Init を呼び出す (行番号 39)。 である。なお、先の説明からも分かるように、 この場合, 関数 Init 実行後の配列 とになる。 配列sの先頭から 北 0となる位置 (要素番号) であり、 変数 C は S [I] = 1, S [I + 1] ≠ 0 のときに+1 される (行番号 41) の変数 C は 2 となる。呼び出された関数 I nit では,N=2, K=2 とな3の先頭から 2 個の要素に 1 を設定する (N-K=0 なので 0 は設定しな 大実行後の配列 S は次のようになる。これで,次の組合せの状態ができ この場合は L=3 であるから このときの引数は配列 S, 変数 L-1, 変数 L は連続する 2 要素の値の組合 5 L-1 は

2

ß	
1]-
,	2
0	ω
ш	4
0	οτ
	_

遍合 合)は、配列sには値を設定せずに、 了する。 となるため, 要素の入替えが行われると, :深の人含なが行われると,変数Rの値が0になるので,繰返し条件の一つであー1 を満たさなくなり,繰返し処理を抜けて,返却値として 0 を返して関数をこう。 ここ る。なお、配列Sの要素中にS[I]=1,S[I+1]=0の組合せが見つからなった(渡された組合せの状態が、この関数のアルゴリズムで得られる最終形である場は、配列Sには値を設定せずに、変数Rは-1のままとなり、関数 Next の返却-1となる。このとき、主プログラム Main では、行番号5の変数Rの値が-1るため、繰返し処理を抜けて、プログラムの実行が終了する。 変数 R の値が 0 になるので、繰返し条件の

- 空欄 の内容は変化しない)。このため、最後に関数 Next の返却値が 0 以外 (-1) になったときには配列 S の内容を印字する必要はない。これらのことから、関数 Next 実行直後に処理を繰り返すかどうかを判定し、処理を継続する場合だけ配列 S の内容を印字する (P) が正解となる。 答群を見ると、(イ)以外は、最初に関数Nextを呼び出す前に、副プログラムDumpを呼び出すので、(イ)以外が正解の候補となる。そして、(ウ)は繰返し処理の中で、関数Nextの実行後にDumpを実行しており、(エ)は繰返し処理の終了後にもDumpを実行している。このため、(ウ)と(エ)では関数Nextの返却値が一1の場合にも配列Sの内容を印字することになる。[プログラムの説明]によると、関数Nextはな返却値として一を返すのは、この関数のアル 』a:配列Sの先頭からΝ個の要素に格納されている値を印字できるようにαの部分を変更する。配列Sには、まず行番号3の関数 Init の呼出しで値が格納される。続いて行番号 4~6 の繰返し処理中で関数 Next を呼び出す都度、そ ゴリズムで得られる最終形の場合であり、配列 S には値を設定しない(配列 S の内容は変化しない)。このため、最後に関数 Next の返却値が 0 以外 (-1)の内容は変化する。したがって、最初に関数 Next を呼び出す前に印字しな と行番号 3 の関数 Init 実行後の配列 s の内容を印字す とができない。
- S[L+1] (その次 ,次に繰返 $35\sim38$) し処そ
- を表すことになる。よって, 空欄 d:与えられた"1, 0, 1, レースする。 (エ) が正解である。 1"を配列 S
- 1, 0, の初期状態として関数 Next 4

初期状態	<u> </u>	0		0	1	ΟΩ
	Οī	4	ω	2	ш	

変数Lの初期値は1であるため,

S[1](S[L])=1, S[2](S[L+1]) となり, この2要素の内容が入れ替わる (行番号 37,

Ø	
0	,_
-	22
ш	ω
0	4
щ	57

配列 S の内容は変化しない。 に 0 をセットする。そして, し条件を判定するが,R=-1~ を終了する。よって、 (=0)次に関数 Init を満たさない。 の内容は変化しない。その後、関数 Next の行番号 40 に戻り である。関数 Init 側では、変数 Kが Oであるため、 を呼び出す。 配列 s の内容は, "0, 1, 1, 0, このとき引数は配列 S, 変数 1-数 L-1 (=0) と C , 1≤K≦N (行番号

空欄 e, f:関数 Next から関数 Init が呼び出されるのは

S[L+1] = 0

合せが最初に見つかった位置であり、変数 c は配列 S[L] の要素が 1 でかつ、S[L+1] の要素が 0 でない場合に+1 されるが、これは 1,0 の組合せを見つい てはし の場合である。そして -1 とcに対応する。処理の概要でも確認したが, のとき,関数 Init が受け取る引数 N と これは 1,0の組合せを見つけ 发 N と K は, 関数 Next 変数 L は, 1,0の組

るまでの、値が1の要素の個数である。連続する2要素が最も早く1,0になるパターンは、配別5の要素が1,0,…の場合であり、このとき、1=1,C=0となるため、関数 Init 側ではN=0,K=0となる。また、最も遅く1と0になるパターンは、"0,1,1,0"の場合であり、このとき、1=4,C=2となるため、関数 Init 側ではN=8,K=2とる。よって、Nの範囲は、0~3,Kの範囲は0~2となり、空欄eは(イ)、空欄fは(ア)が正解である。で組合せの一つを表現する。問題ではK=3なので、1が3個設定されなければならない。解答群を見ると、1が3個あるのは、(イ)と(ウ)である。そして、配列5は行番号3の関数 Init の呼出しによって"1,1,0,0"から始まり、連続する2要素の値の組合せが1,0の場合にその内容を"1,1,0,10"のように入れ替えるため、主プログラム Main の終了時点における配列5、要素は"1,1,1,0,0"から変化しているはずなので、(ウ)は誤りである。 したがって、(イ)が正解となる。なお、配列 S の変化をトレースすると次のようになる。○付き数字の行は、プログラムのαの部分を入れ替えた場合に、副プログラム Dumb で印字される配列 S の内容である。最後の⑩で印字される凡容は(イ)と一致している。

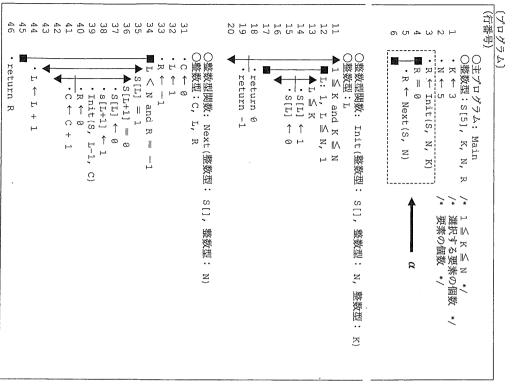
四 四 四 組合す વા -夕構造及びアルゴリズム)

(H23 春-FE 午後間 8)

[解]

b−±, C d - γ , e-√, f-γ, g-

N個の要素中からK個の要素を選ぶ組合せをすべて求めるアルゴリズムの問題である。プログラムがどのように処理を行っているかを問う内容であり、プログラムをトレースする力、関数の引数と返却値に関する知識などが必要である。〔プログラムの説明〕だけでは配列sの扱いや各関数のアルゴリズムを把握しにくく、利用部門に所属する受験者にとっては難しく感じられたと思われる。プログラムの動きを理解するために、与えられたデータを使ってトレースしたり、設問中の記述や解答群を参考に考察したりする必要があり、難易度は、やや高いといえる。



関数 Init 及び Next の処理の概要を確認する。 最初に (プロ グラ ムの説明) [プロ グラム]の内容を基に, 主プログラム Main,

主プログラム Main は、行番号 3 で関数 Init を呼び出す。このとき、関数 Init には引数 (関数に渡す値) として配列 S と変数 N に 5、変数 K に 3 が渡される。関数 Init の処理が終わると、変数 R に関数 Init の返却値 (関数 Init が Return する値) をセットする。行番号 5 の関数 Next を呼び出す処理は、変数 R の値が 0 である間繰り返される。 関数 Next の引数は配列 S と変数 N の値であり、 関数 Next の処理が終わると変数 R に関数 Next の返却値をセットする。 関数 Init は、1 \le K \le K \le M \ge M

0を返して関数を終了する。

Ω	
1	ш
1	2
1	မ
0	4
0	σı

めて配列 S に設定し、返却値として 0 を返す。具体的には、番号 1 から順に調べ、S[L]=1 (行番号 35)、S[L+1]=6 その値を入れ替える(行番号 37、38)。例えば、1 回目に1の要素番号 1~5 の内容は次の①のようになっているので、=1、S[L+1]=8 (連続する 2 要素の値の組合せが 1 と 0 の入替えが行われ, 関数 Next は, 渡された配列Sに対して所定の操作を行い,次の組合せの状態を記し、返却値として0を返す。具体的には,受け取った配列Sを要調べ,S[I]=1 (行番号 35)。 [L+1]=6 (行番号 36) の場合に, (連続する2要素の値の組合せが1と 配列Sは②のようになる。 1回目に呼び出された際は,配列 s いるので,変数 L=3のときに S[L] 0 になり, 次の組合せの状態を求 3 要素

	(0)		Θ	
	ß		ß	
	יי	_	1	ᆫ
-	ы	12	1	2
	0	ယ	1	ယ
	н	4	0	4
	0	Oī	0	οı

でいるめる。 となが1と でなる。ま で、10場 のため。関 で、2)。関数 で、2)。 続いて、関数 Init を呼び出す (行番号 39)。 である。なお、先の説明からも分かるように、 この場合, 関数 Init 実行後の配列 とになる。 配列sの先頭から 北 0となる位置 (要素番号) であり、 変数 C は S [I] = 1, S [I + 1] ≠ 0 のときに+1 される (行番号 41) の変数 C は 2 となる。呼び出された関数 I nit では,N=2, K=2 とな3の先頭から 2 個の要素に 1 を設定する (N-K=0 なので 0 は設定しな 大実行後の配列 S は次のようになる。これで,次の組合せの状態ができ この場合は L=3 であるから このときの引数は配列 S, 変数 L-1, 変数 L は連続する 2 要素の値の組合 5 L-1 は

2

ß	
1]-
,	2
0	ω
ш	4
0	οτ
	_

遍合 合)は、配列sには値を設定せずに、 了する。 となるため, 要素の入替えが行われると, :深の人含なが行われると,変数Rの値が0になるので,繰返し条件の一つであー1 を満たさなくなり,繰返し処理を抜けて,返却値として 0 を返して関数をこう。 ここ る。なお、配列Sの要素中にS[I]=1,S[I+1]=0の組合せが見つからなった(渡された組合せの状態が、この関数のアルゴリズムで得られる最終形である場は、配列Sには値を設定せずに、変数Rは-1のままとなり、関数 Next の返却-1となる。このとき、主プログラム Main では、行番号5の変数Rの値が-1るため、繰返し処理を抜けて、プログラムの実行が終了する。 変数 R の値が 0 になるので、繰返し条件の

- 空欄 の内容は変化しない)。このため、最後に関数 Next の返却値が 0 以外 (-1) になったときには配列 S の内容を印字する必要はない。これらのことから、関数 Next 実行直後に処理を繰り返すかどうかを判定し、処理を継続する場合だけ配列 S の内容を印字する (P) が正解となる。 答群を見ると、(イ)以外は、最初に関数Nextを呼び出す前に、副プログラムDumpを呼び出すので、(イ)以外が正解の候補となる。そして、(ウ)は繰返し処理の中で、関数Nextの実行後にDumpを実行しており、(エ)は繰返し処理の終了後にもDumpを実行している。このため、(ウ)と(エ)では関数Nextの返却値が一1の場合にも配列Sの内容を印字することになる。[プログラムの説明]によると、関数Nextはな返却値として一を返すのは、この関数のアル 』a:配列Sの先頭からΝ個の要素に格納されている値を印字できるようにαの部分を変更する。配列Sには、まず行番号3の関数 Init の呼出しで値が格納される。続いて行番号 4~6 の繰返し処理中で関数 Next を呼び出す都度、そ ゴリズムで得られる最終形の場合であり、配列 S には値を設定しない(配列 S の内容は変化しない)。このため、最後に関数 Next の返却値が 0 以外 (-1)の内容は変化する。したがって、最初に関数 Next を呼び出す前に印字しな と行番号 3 の関数 Init 実行後の配列 s の内容を印字す とができない。
- S[L+1] (その次 ,次に繰返 $35\sim38$) し処そ
- を表すことになる。よって, 空欄 d:与えられた"1, 0, 1, レースする。 (エ) が正解である。 1"を配列 S
- 1, 0, の初期状態として関数 Next 4

初期状態	<u> </u>	0		0	1	ΟΩ
	Οī	4	ω	2	ш	

変数Lの初期値は1であるため,

S[1](S[L])=1, S[2](S[L+1]) となり, この2要素の内容が入れ替わる (行番号 37,

Ø	
0	,_
-	22
ш	ω
0	4
щ	57

配列 S の内容は変化しない。 に 0 をセットする。そして, し条件を判定するが,R=-1~ を終了する。よって、 (=0)次に関数 Init を満たさない。 の内容は変化しない。その後、関数 Next の行番号 40 に戻り である。関数 Init 側では、変数 Kが Oであるため、 を呼び出す。 配列 s の内容は, "0, 1, 1, 0, このとき引数は配列 S, 変数 1-数 L-1 (=0) と C , 1≤K≦N (行番号

空欄 e, f:関数 Next から関数 Init が呼び出されるのは

S[L+1] = 0

合せが最初に見つかった位置であり、変数 c は配列 S[L] の要素が 1 でかつ、S[L+1] の要素が 0 でない場合に+1 されるが、これは 1,0 の組合せを見つい てはし の場合である。そして -1 とcに対応する。処理の概要でも確認したが, のとき,関数 Init が受け取る引数 N と これは 1,0の組合せを見つけ 发 N と K は, 関数 Next 変数 L は, 1,0の組

るまでの、値が1の要素の個数である。連続する2要素が最も早く1,0になるパターンは、配別5の要素が1,0,…の場合であり、このとき、1=1,C=0となるため、関数 Init 側ではN=0,K=0となる。また、最も遅く1と0になるパターンは、"0,1,1,0"の場合であり、このとき、1=4,C=2となるため、関数 Init 側ではN=8,K=2とる。よって、Nの範囲は、0~3,Kの範囲は0~2となり、空欄eは(イ)、空欄fは(ア)が正解である。で組合せの一つを表現する。問題ではK=3なので、1が3個設定されなければならない。解答群を見ると、1が3個あるのは、(イ)と(ウ)である。そして、配列5は行番号3の関数 Init の呼出しによって"1,1,0,0"から始まり、連続する2要素の値の組合せが1,0の場合にその内容を"1,1,0,10"のように入れ替えるため、主プログラム Main の終了時点における配列5、要素は"1,1,1,0,0"から変化しているはずなので、(ウ)は誤りである。 したがって、(イ)が正解となる。なお、配列 S の変化をトレースすると次のようになる。○付き数字の行は、プログラムのαの部分を入れ替えた場合に、副プログラム Dumb で印字される配列 S の内容である。最後の⑩で印字される凡容は(イ)と一致している。