

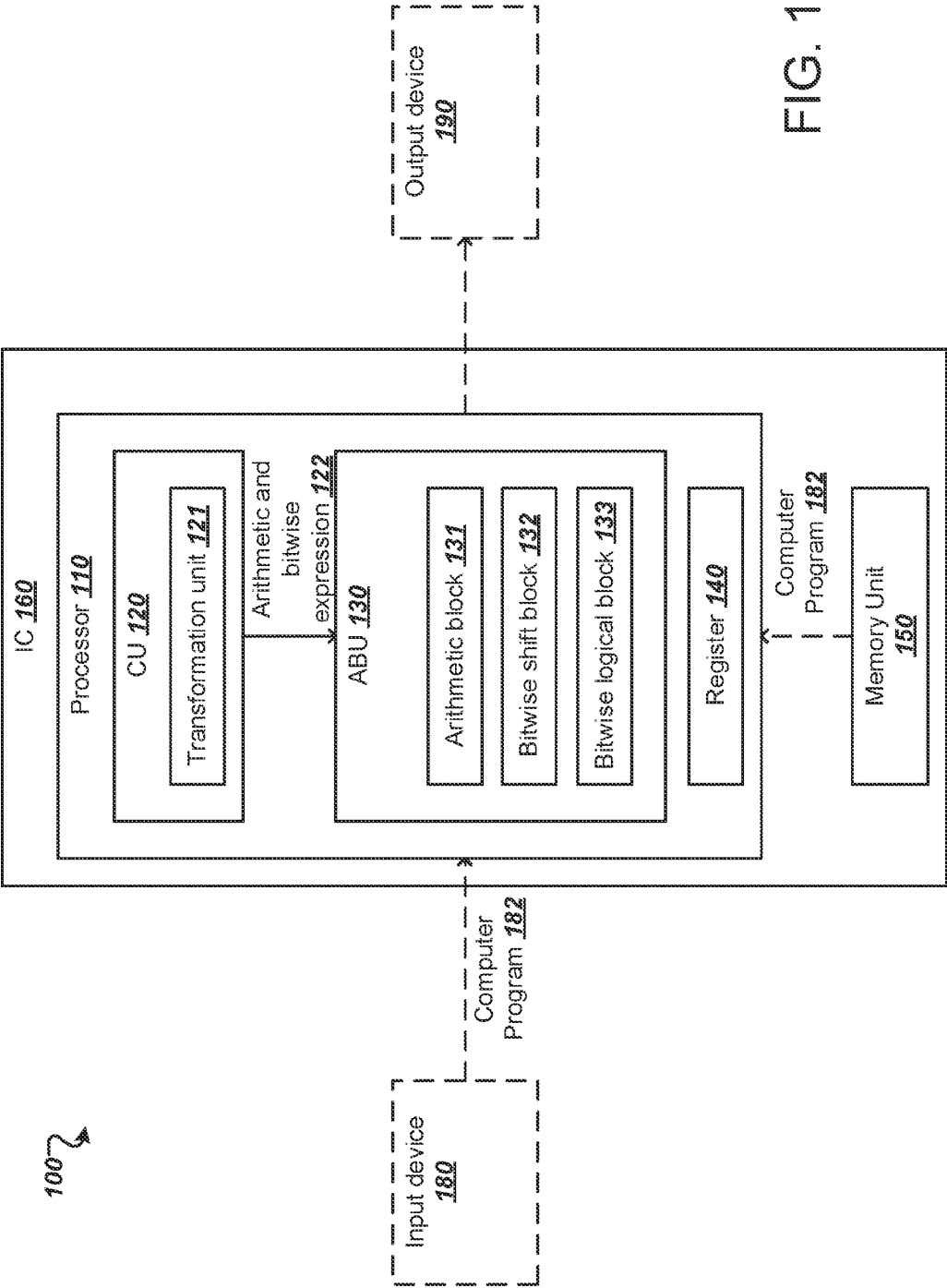


FIG. 1

2/21

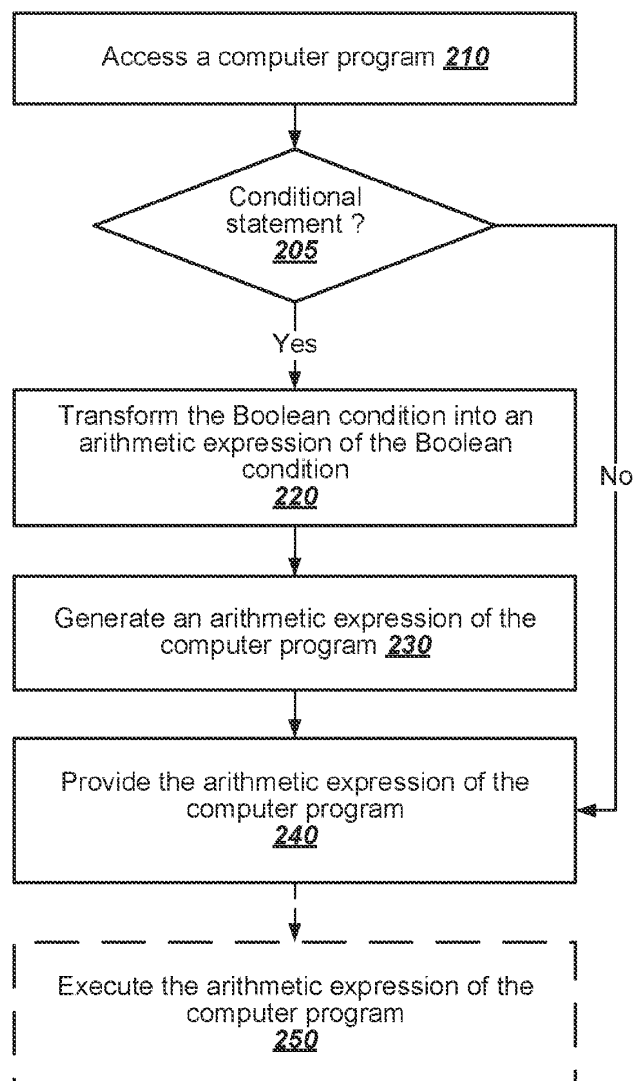


FIG. 2

310 ↪

```
If (boolean condition 301) Then
  (consequent task 302)
Else
  (alternative task 303)
End If
```

FIG. 3

4/21

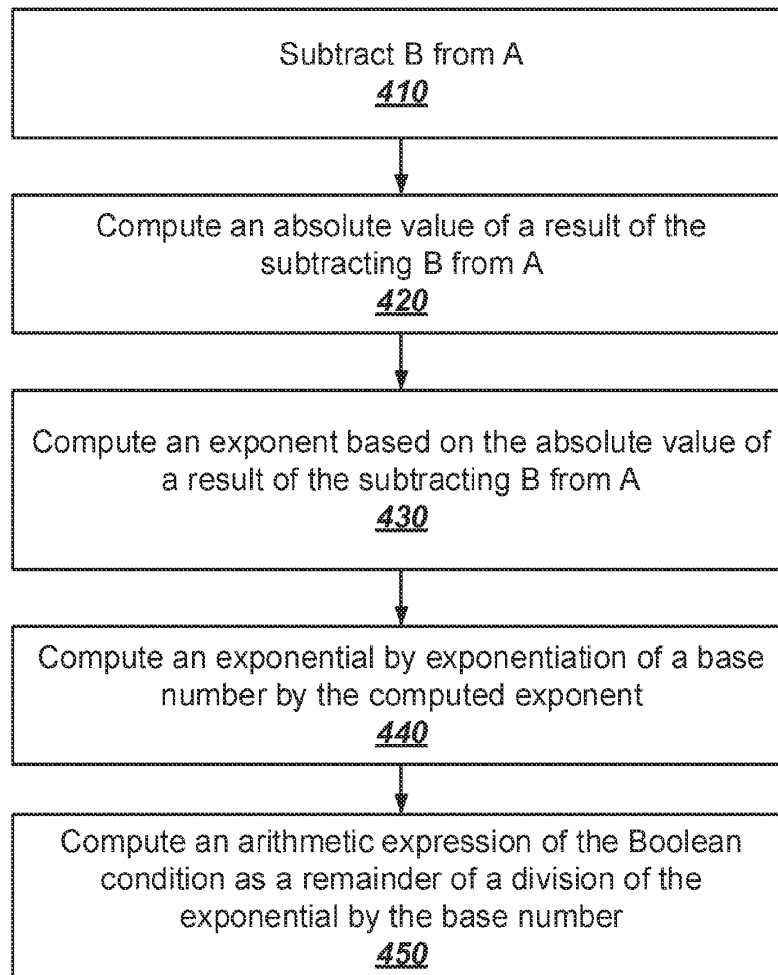


FIG. 4

5/21

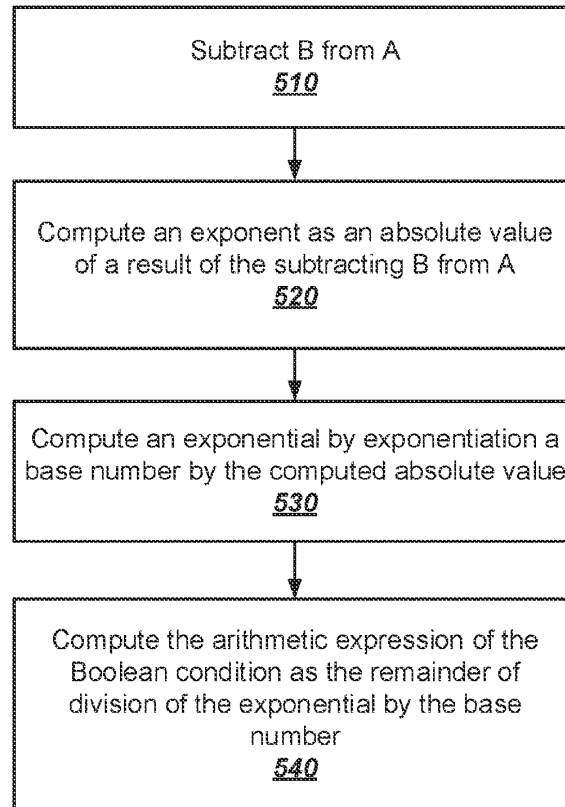


FIG. 5

6/21

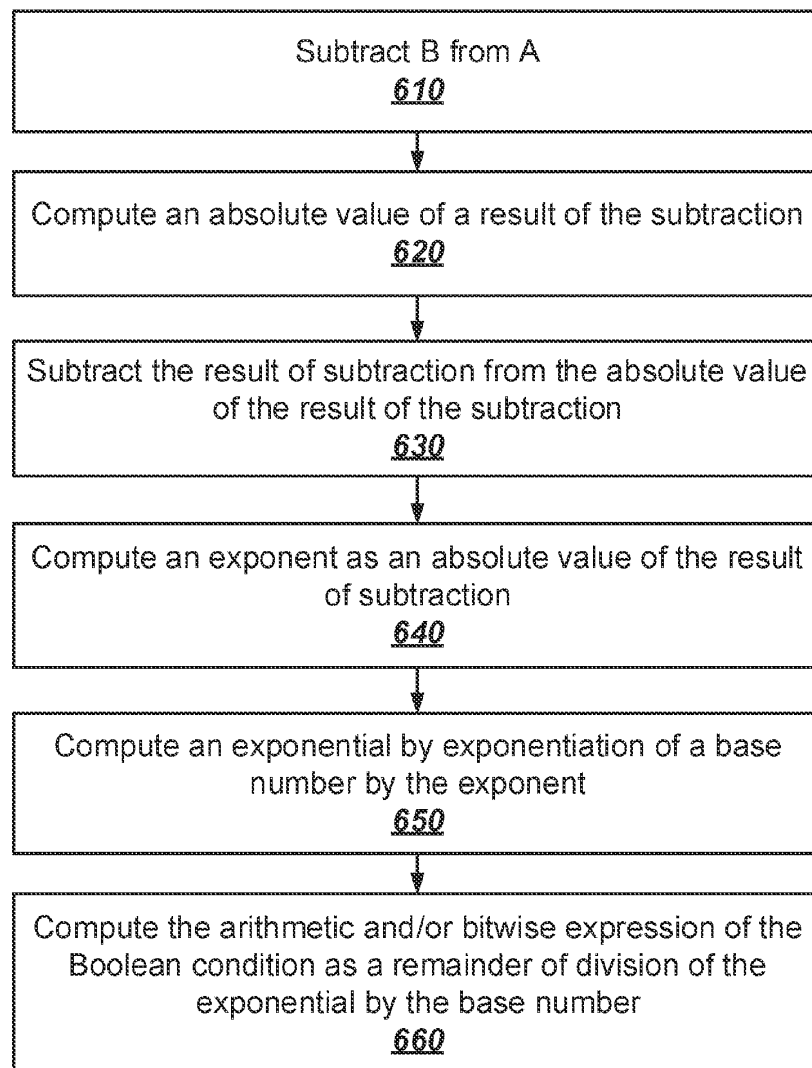


FIG. 6

7/21

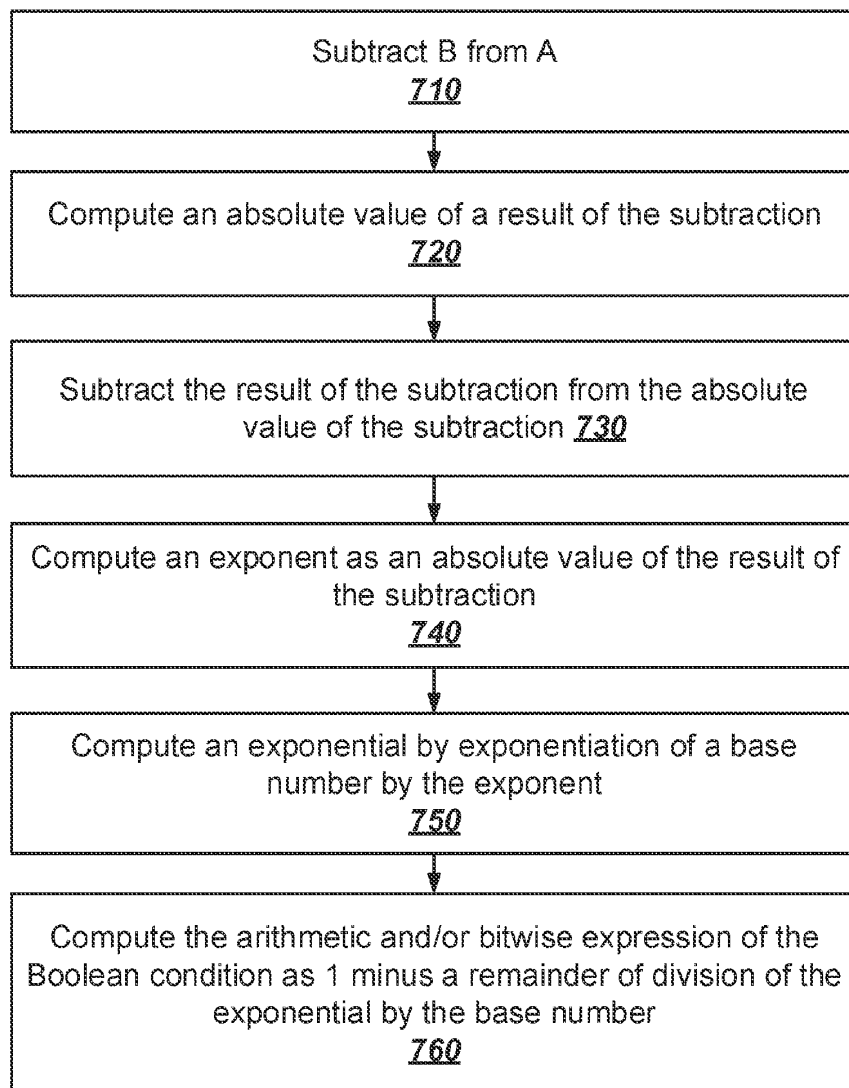


FIG. 7A

8/21

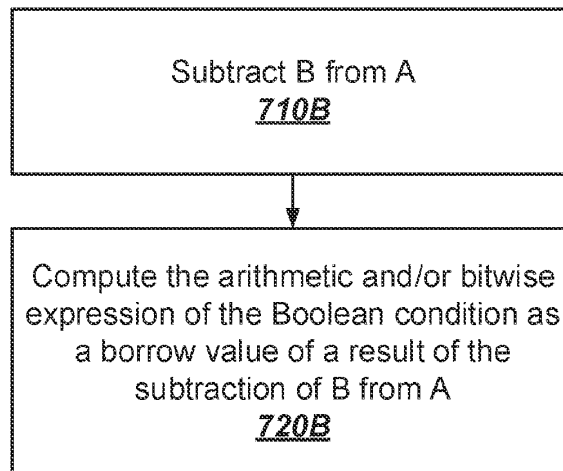


FIG. 7B

9/21

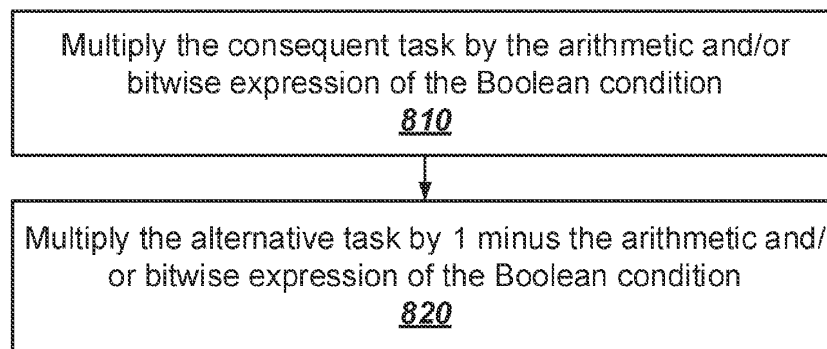


FIG. 8

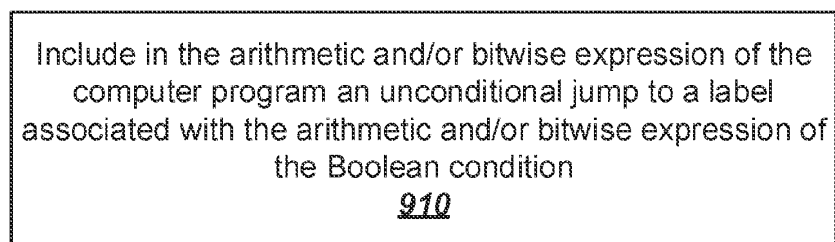


FIG. 9

10/21

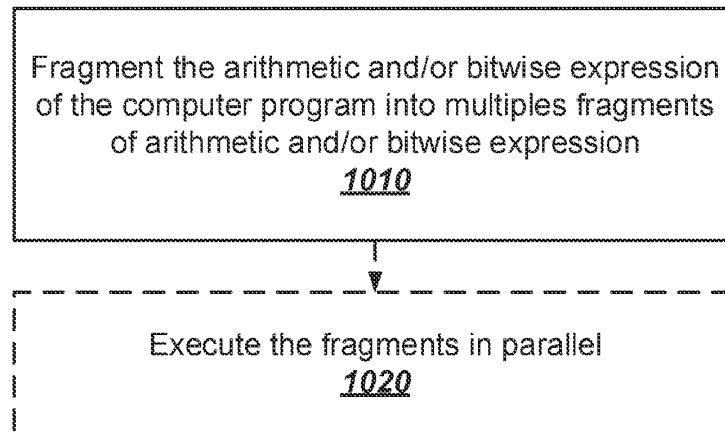


FIG. 10

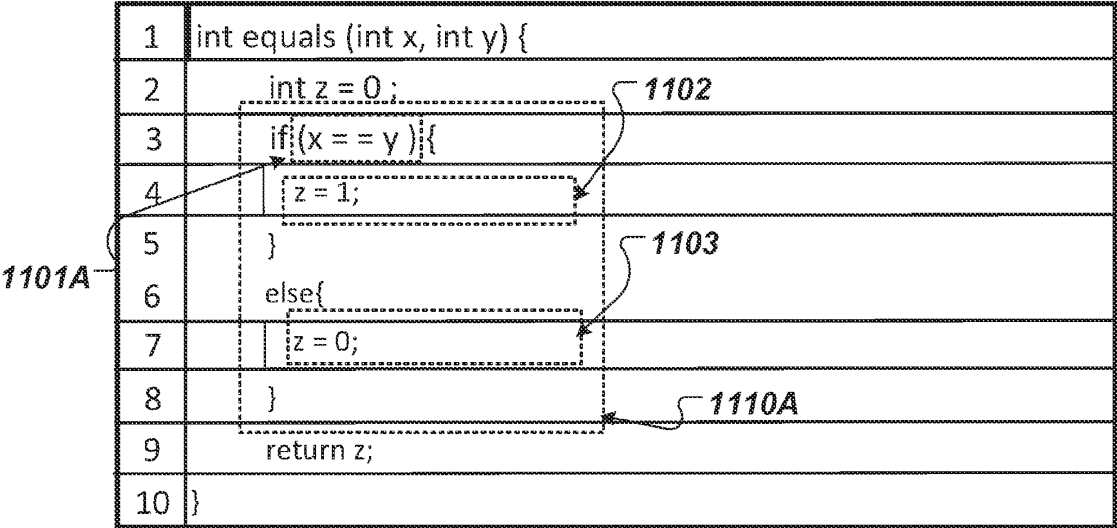


FIG. 11A

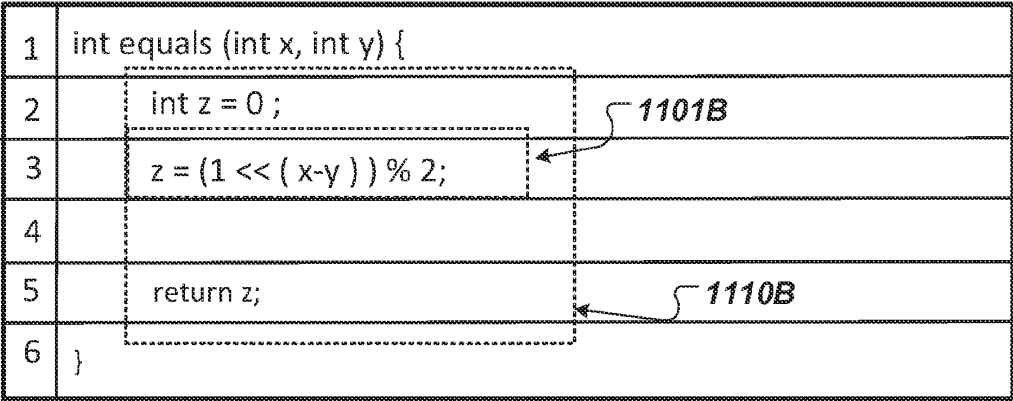


FIG. 11B

12/21

1100C

1	equals (int, int) :
2	push rbp
3	mov rbp, rsp
4	mov DWORD PTR [rbp-20], edi
5	mov DWORD PTR [rbp-24], esi
6	mov DWORD PTR [rbp-4], 0
7	mov eax, DWORD PTR [rbp-20]
8 ✓	cmp eax, DWORD PTR [rbp-24]
9	jne .L2
10	mov DWORD PTR [rbp-4], 1
11	jmp .L3
12	.L2:
13	mov DWORD PTR [rbp-4], 0
14	.L3:
15	mov eax, DWORD PTR [rbp-4],
16	pop rbp
17	ret

FIG. 11C

1100D

1	equals (int, int) :		
2	push	rbp	
3	mov	rbp, rsp	
4	mov	DWORD PTR [rbp-20], edi	
5	mov	DWORD PTR [rbp-24], esi	
6	mov	DWORD PTR [rbp-4], 0	
7	mov	eax, DWORD PTR [rbp-20]	
8	sub	eax, DWORD PTR [rbp-24]	
9	mov	edx, 1	
10	mov	ecx, eax	
11	sal	edx, cl	
12	mov	eax, edx	
13	cdq		
14	shr	edx, 31	
15	add	eax, edx	
16	and	eax, 1	
17	sub	eax, edx	
18	mov	DWORD PTR [rbp-4], eax	
19	mov	eax, DWORD PTR [rbp-4]	
20	pop	rbp	
21	ret		

FIG. 11D

14/21

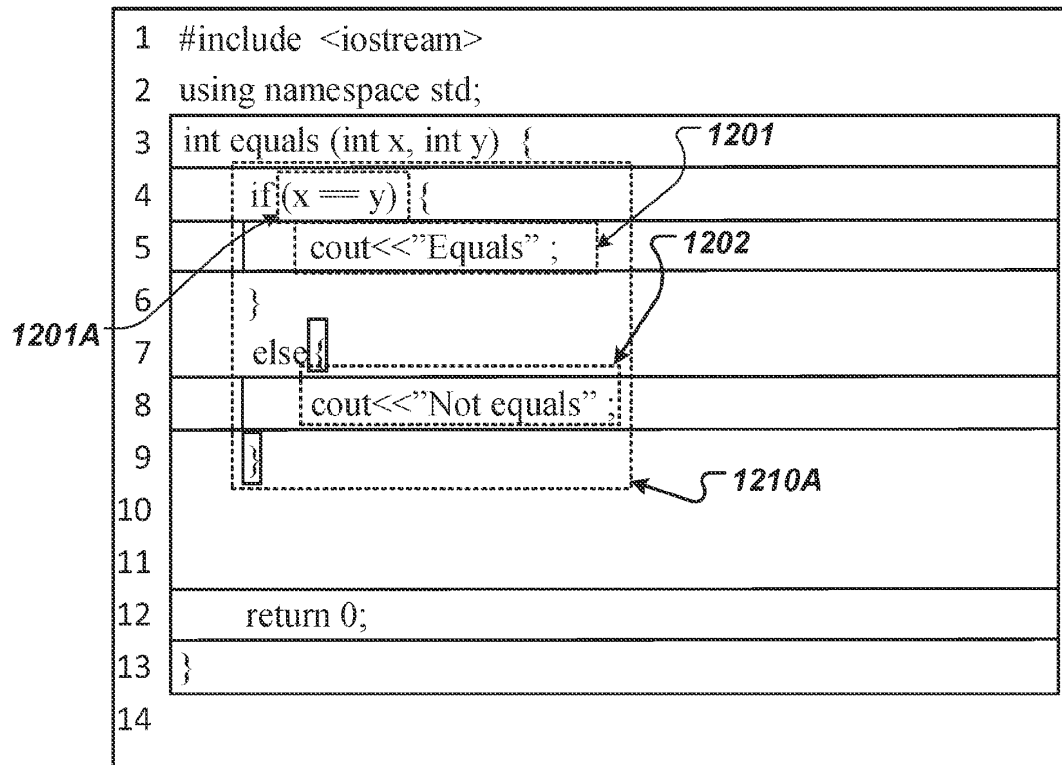


FIG. 12A

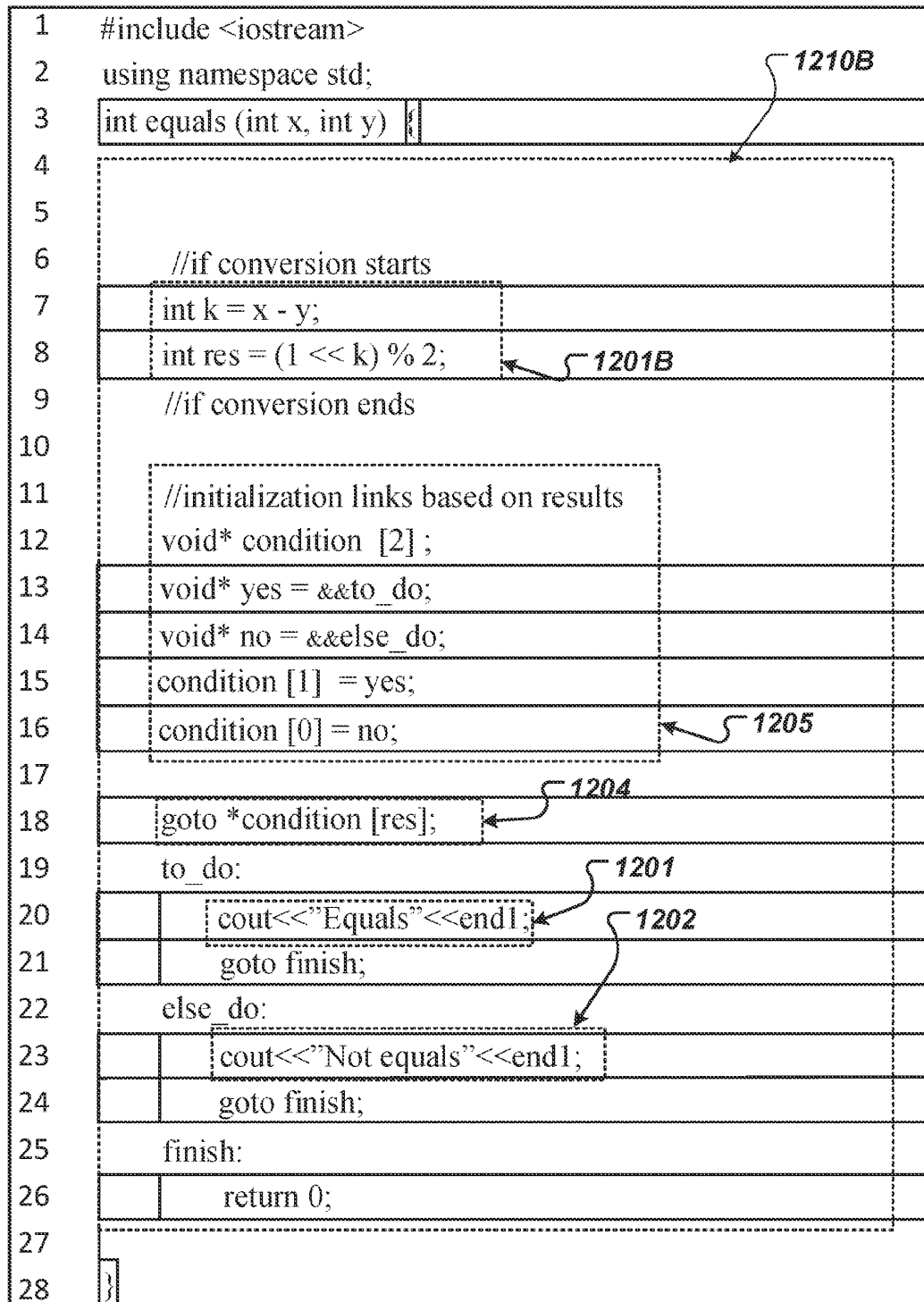


FIG. 12B

1200D

1	.LC0:	
2	.string	"Equals"
3	.LC1:	
4	.string	"Not equals"
5	equal (int, int):	
6	push	rbp
7	mov	rbp, rsp
8	sub	rsp, 64
9	mov	DWORD PTR [rbp-52], edi
10	mov	DWORD PTR [rbp-56], esi
11	mov	eax, DWORD PTR [rbp-52]
12	sub	eax, DWORD PTR [rbp-56]
13	mov	DWORD PTR [rbp-4], eax
14	mov	eax, DWORD PTR [rbp-4]
15	mov	edx, 1
16	mov	ecx, eax
17	sar	edx, cl
18	mov	eax, edx
19	cdq	
20	shr	edx, 31
21	add	eax, edx
22	and	eax, 1
23	sub	eax, edx
24	mov	DWORD PTR [rbp-8], eax
25	mov	QWORD PTR [rbp-16], OFFSET FLAT: .L2
26	mov	QWORD PTR [rbp-24], OFFSET FLAT: .L3
27	mov	rax, QWORD PTR [rbp-16]
28	mov	QWORD PTR [rbp-40], rax
29	mov	rax, QWORD PTR [rbp-24]
30	mov	QWORD PTR [rbp-48], rax

FIG. 12D

31	mov	eax, DWORD PTR (rbp-8)
32	cldq	
33	mov	rax, QWORD PTR (rbp-48+rax*8)
34	nop	
35	jmp	rax
36	L2:	
37	mov	esi, OFFSET FLAT: _LCO
38	mov	edi, OFFSET FLAT: _ES4cout
39	call	std::basic_ostream<char, std::char_traits<char>>::{std::basic_ostream<char, std::char_traits<char>> &, char const*}
40	mov	esi, OFFSET FLAT: _ZS4end11eslflchar_traitsEERSu13basic_ostreamIT_0_ES6
41	mov	rdi, rax
42	call	std::basic_ostream<char, std::char_traits<char>>::operator<<({std::basic_ostream<char, std::char_traits<char>> &(*)}{std::basic_ostream<char, std::char_traits<char>> &(*)}
43	jmp	_L5
44	L3:	
45	mov	esi, OFFSET FLAT: _LCI
46	mov	edi, OFFSET FLAT: _ZS4cout
47	call	std::basic_ostream<char, std::char_traits<char>> & std::operator<<({std::char_traits<char>> &(*)}{std::basic_ostream<char, std::char_traits<char>> &, char const*}
48	mov	esi, OFFSET FLAT: _ZS4end11eslflchar_traitsEERSu13basic_ostreamIT_0_ES6
49	mov	rdi, rax
50	call	std::basic_ostream<char, std::char_traits<char>>::operator<<({std::basic_ostream<char, std::char_traits<char>> &(*)}{std::basic_ostream<char, std::char_traits<char>> &(*)}
51	nop	
52	L5:	
53	mov	eax, 0
54	leave	
55	ret	

FIG. 12D (cont)

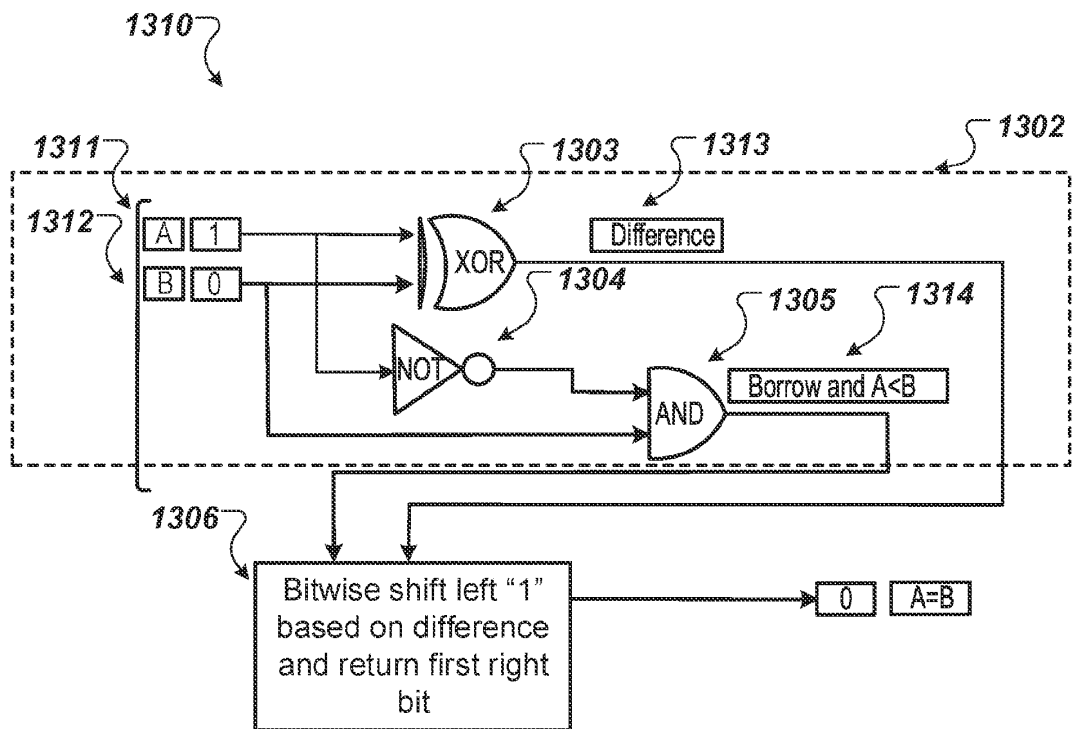


FIG. 13A

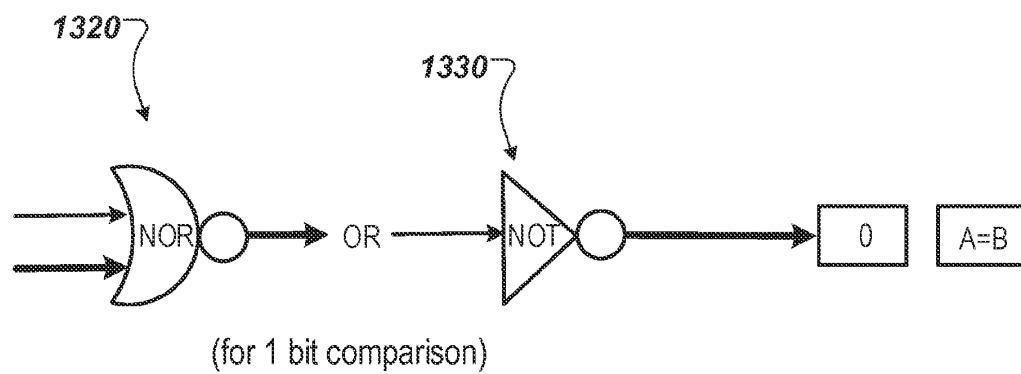


FIG. 13B

20/21

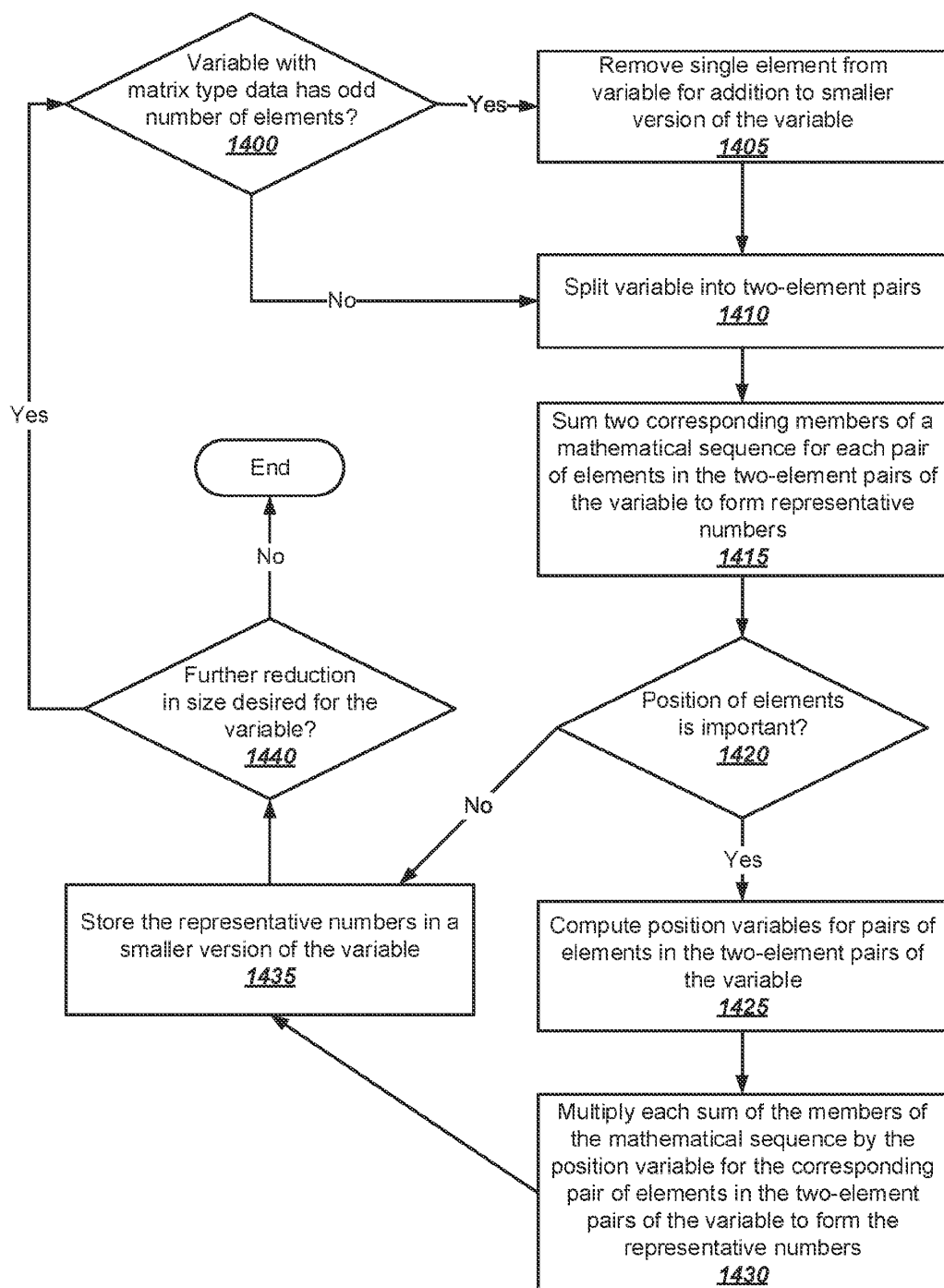


FIG. 14

21/21

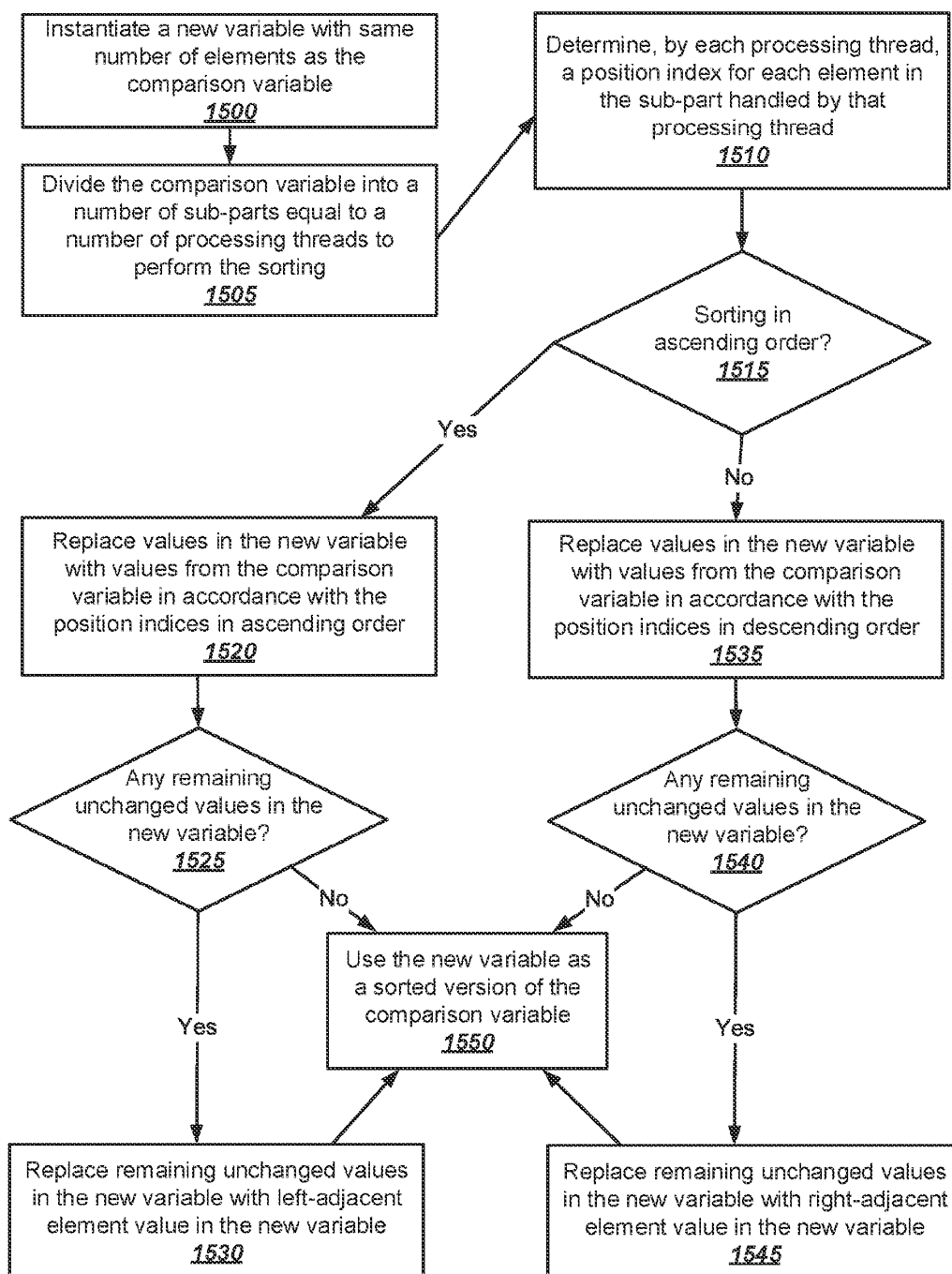


FIG. 15