

1	add:
2	lui  a3, 1
3	addi a3, a3, -100
4	add  a4, a2, a3
5	add  a5, a0, a3
6	add  a6, a1, a3
7	sltu a5, a2, a5
8	sltu a3, a0, a4
9	and  a3, a5, a3
10	sltu a5, a2, a6
11	sltu a4, a1, a4
12	and  a4, a5, a4
13	or   a3, a3, a4
14	mv   a6, zero
15	bnez a3, .LBB0_3
16	addi a6, zero, 996    #ベクトル処理回数の指定
17	mv   a7, a0
18	mv   a5, a1
19	mv   a3, a2
20	addi a4, zero, 996    #ループカウンタの設定
21	.LBB0_2:
22	vsetivli  zero, 4, e32, m1, ta, mu
23	vle32.v  v25, (a7)    #ベクトルロード
24	vle32.v  v26, (a5)    #ベクトルロード
25	vadd.w   v25, v26, v25  #ベクトル加算
26	vse32.v  v25, (a3)    #ベクトルストア
27	addi a4, a4, -4    #ループカウンタをデクリメント
28	addi a3, a3, 16    #アドレス更新
29	addi a5, a5, 16    #アドレス更新
30	addi a7, a7, 16    #アドレス更新
31	bnez a4, .LBB0_2
32	.LBB0_3:
33	addi a4, a6, -999    #余りの要素数を計算
34	slli  a3, a6, 2    #処理済要素分のアドレス計算
35	add  a2, a2, a3    #アドレス更新
36	add  a1, a1, a3    #アドレス更新
37	add  a0, a0, a3    #アドレス更新
38	.LBB0_4:
39	lw   a6, 0(a0)    #ロード
40	lw   a5, 0(a1)    #ロード
41	mv   a3, a4        #
42	add  a4, a5, a6    #加算
43	sw   a4, 0(a2)    #ストア
44	addi a4, a3, 1    #カウンタをインクリメント
45	addi a2, a2, 4    #アドレス更新
46	addi a1, a1, 4    #アドレス更新
47	addi a0, a0, 4    #アドレス更新
48	bgeu  a4, a3, .LBB0_4  #余りがまだあったら再度計算
49	ret