

```
1:
2:
3: ----- Assem Tests -----
4:
5: Instruction: add $0, $0, $0
6: Expected (big-endian): 00000000 00000000 00000000 00100000
7: Matched Pattern: big-end
8: Actual Memory:
9: 4: 00000000 00000000 00000000 00100000
10:
11: Instruction: add $1, $2, $3
12: Expected (big-endian): 00000000 01000011 00001000 00100000
13: Matched Pattern: big-end
14: Actual Memory:
15: 4: 00000000 01000011 00001000 00100000
16:
17: Instruction: beq $0, $0, 0
18: Expected (big-endian): 00010000 00000000 00000000 00000000
19: Matched Pattern: big-end
20: Actual Memory:
21: 4: 00010000 00000000 00000000 00000000
22:
23: Instruction: beq $1, $2, 2
24: Expected (big-endian): 00010000 00100010 00000000 00000010
25: Matched Pattern: big-end
26: Actual Memory:
27: 4: 00010000 00100010 00000000 00000010
28:
29: Instruction: beq $0, $0, -1
30: Expected (big-endian): 00010000 00000000 11111111 11111111
31: Matched Pattern: big-end
32: Actual Memory:
33: 4: 00010000 00000000 11111111 11111111
34:
35: Instruction: beq $1, $3, -3
36: Expected (big-endian): 00010000 00100011 11111111 11111101
37: Matched Pattern: big-end
38: Actual Memory:
39: 4: 00010000 00100011 11111111 11111101
40:
41: Instruction: jr $0
42: Expected (big-endian): 00000000 00000000 00000000 00001000
43: Matched Pattern: big-end
44: Actual Memory:
45: 4: 00000000 00000000 00000000 00001000
46:
47: Instruction: jr $31
48: Expected (big-endian): 00000011 11100000 00000000 00001000
49: Matched Pattern: big-end
50: Actual Memory:
51: 4: 00000011 11100000 00000000 00001000
52:
53: Instruction: jal 0
54: Expected (big-endian): 00001100 00000000 00000000 00000000
55: Matched Pattern: big-end
56: Actual Memory:
57: 4: 00001100 00000000 00000000 00000000
58:
59: Instruction: jal 2
60: Expected (big-endian): 00001100 00000000 00000000 00000010
61: Matched Pattern: big-end
62: Actual Memory:
```

```
63: 4: 00001100 00000000 00000000 00000010
64:
65: Instruction: jal -1
66: Expected (big-endian): 00001111 11111111 11111111 11111111
67: Matched Pattern: big-end
68: Actual Memory:
69: 4: 00001111 11111111 11111111 11111111
70:
71: Instruction: lui $0, 0
72: Expected (big-endian): 00111100 00000000 00000000 00000000
73: Matched Pattern: big-end
74: Actual Memory:
75: 4: 00111100 00000000 00000000 00000000
76:
77: Instruction: lui $4, 17
78: Expected (big-endian): 00111100 00000100 00000000 00010001
79: Matched Pattern: big-end
80: Actual Memory:
81: 4: 00111100 00000100 00000000 00010001
82:
83: Instruction: ori $0, $0, 0
84: Expected (big-endian): 00110100 00000000 00000000 00000000
85: Matched Pattern: big-end
86: Actual Memory:
87: 4: 00110100 00000000 00000000 00000000
88:
89: Instruction: ori $1, $2, 3
90: Expected (big-endian): 00110100 01000001 00000000 00000011
91: Matched Pattern: big-end
92: Actual Memory:
93: 4: 00110100 01000001 00000000 00000011
94:
95: Instruction: li $0, 0
96: Expected (big-endian): 00111100 00000000 00000000 00000000 00110100 00000000 0
00000000 00000000
97: Matched Pattern: big-end
98: Actual Memory:
99: 4: 00111100 00000000 00000000 00000000
100: 8: 00110100 00000000 00000000 00000000
101:
102: Instruction: lw $0, 0($0)
103: Expected (big-endian): 10001100 00000000 00000000 00000000
104: Matched Pattern: big-end
105: Actual Memory:
106: 4: 10001100 00000000 00000000 00000000
107:
108: Instruction: lw $1, 4($2)
109: Expected (big-endian): 10001100 01000001 00000000 00000100
110: Matched Pattern: big-end
111: Actual Memory:
112: 4: 10001100 01000001 00000000 00000100
113:
114: Instruction: lw $1, -1($2)
115: Expected (big-endian): 10001100 01000001 11111111 11111111
116: Matched Pattern: big-end
117: Actual Memory:
118: 4: 10001100 01000001 11111111 11111111
119:
120: Instruction: sw $0, 0($0)
121: Expected (big-endian): 10101100 00000000 00000000 00000000
122: Matched Pattern: big-end
123: Actual Memory:
```

```
124: 4: 10101100 00000000 00000000 00000000
125:
126: Instruction: sw $1, 4($2)
127: Expected (big-endian): 10101100 01000001 00000000 00000100
128: Matched Pattern: big-end
129: Actual Memory:
130: 4: 10101100 01000001 00000000 00000100
131:
132: Instruction: sw $1, -1($2)
133: Expected (big-endian): 10101100 01000001 11111111 11111111
134: Matched Pattern: big-end
135: Actual Memory:
136: 4: 10101100 01000001 11111111 11111111
137:
138: Tests Passed: 22 / 22
139:
140:
141: ----- Interp Tests -----
142:
143: Program:
144: li $1, 1
145: add $2, $1, $1
146: [Test Passed]
147:
148: Program:
149: li $1, 3
150: li $2, -1
151: add $3, $1, $2
152: [Test Passed]
153:
154: Program:
155: li $1, 16250367
156: [Test Passed]
157:
158: Program:
159: lui $1, 1
160: [Test Passed]
161:
162: Program:
163: ori $1, $0, 13
164: [Test Passed]
165:
166: Program:
167: ori $1, $0, -1
168: [Test Passed]
169:
170: Program:
171: ori $5, $0, 0
172: ori $5, $5, 2
173: ori $5, $5, 1
174: [Test Passed]
175:
176: Program:
177: ori $1, $0, 20
178: add $5, $0, $0
179: jr $1
180: ori $3, $0, 0
181: ori $5, $0, 1
182: ori $6, $0, 2
183: Register mismatch on $5 - Expected: 0 Actual: 1
184: [Test Failed]
185:
```

```
186: Program:
187: add $5, $0, $0
188: jal 4
189: ori $3, $0, 0
190: ori $4, $0, 1
191: ori $5, $5, 2
192: Register mismatch on $31 - Expected: 8 Actual: 12
193: Register mismatch on $4 - Expected: 0 Actual: 1
194: [Test Failed]
195:
196: Program:
197: add $5, $0, $0
198: jal 16
199: ori $3, $0, 0
200: ori $5, $0, 1
201: ori $5, $5, 2
202: Register mismatch on $31 - Expected: 8 Actual: 12
203: Register mismatch on $5 - Expected: 2 Actual: 0
204: [Test Failed]
205:
206: Program:
207: ori $1, $0, 27
208: sw $1, 40($0)
209: lw $2, 40($0)
210: [Test Passed]
211:
212: Program:
213: ori $1, $0, 25
214: ori $4, $0, 40
215: sw $1, 0($4)
216: lw $2, 0($4)
217: ori $0, $0, 0
218: [Test Passed]
219:
220: Program:
221: ori $1, $0, 25
222: ori $2, $0, 24
223: ori $4, $0, 40
224: sw $1, -4($4)
225: sw $2, 0($4)
226: lw $5, -4($4)
227: lw $6, 0($4)
228: ori $0, $0, 0
229: Unexpected exception : Mips_sim.BadInstruction
230: [Test Failed]
231:
232: Program:
233: add $5, $0, $0
234: beq $0, $0, 3
235: ori $3, $0, 0
236: ori $4, $0, 1
237: ori $5, $0, 2
238: [Test Passed]
239:
240: Program:
241: add $5, $0, $0
242: beq $0, $0, 2
243: ori $3, $0, 0
244: ori $4, $0, 1
245: ori $5, $0, 2
246: Register mismatch on $4 - Expected: 0 Actual: 1
247: [Test Failed]
```

```
248:
249: Program:
250: add $5, $0, $0
251: beq $0, $0, 12
252: ori $3, $0, 0
253: ori $4, $0, 1
254: ori $5, $0, 2
255: Register mismatch on $5 - Expected: 2 Actual: 0
256: [Test Failed]
257:
258: Program:
259: add $5, $0, $0
260: ori $1, $0, 1
261: beq $0, $1, 3
262: ori $3, $0, 0
263: ori $4, $0, 1
264: ori $5, $0, 2
265: [Test Passed]
266:
267: Program:
268: add $5, $0, $0
269: ori $1, $0, 1
270: beq $0, $1, 12
271: ori $3, $0, 0
272: ori $4, $0, 1
273: ori $5, $0, 2
274: [Test Passed]
275:
276:
277:
278: Tests Passed: 12 / 18
```