

KIẾN TRÚC TẬP LỆNH MIPS

STORM TEAM

NHÓM TẬP LỆNH LOGIC



MIPS là gì?

Một dạng tập lệnh trong Kiến trúc máy tính RISC (Reduced Instruction Set Computer)

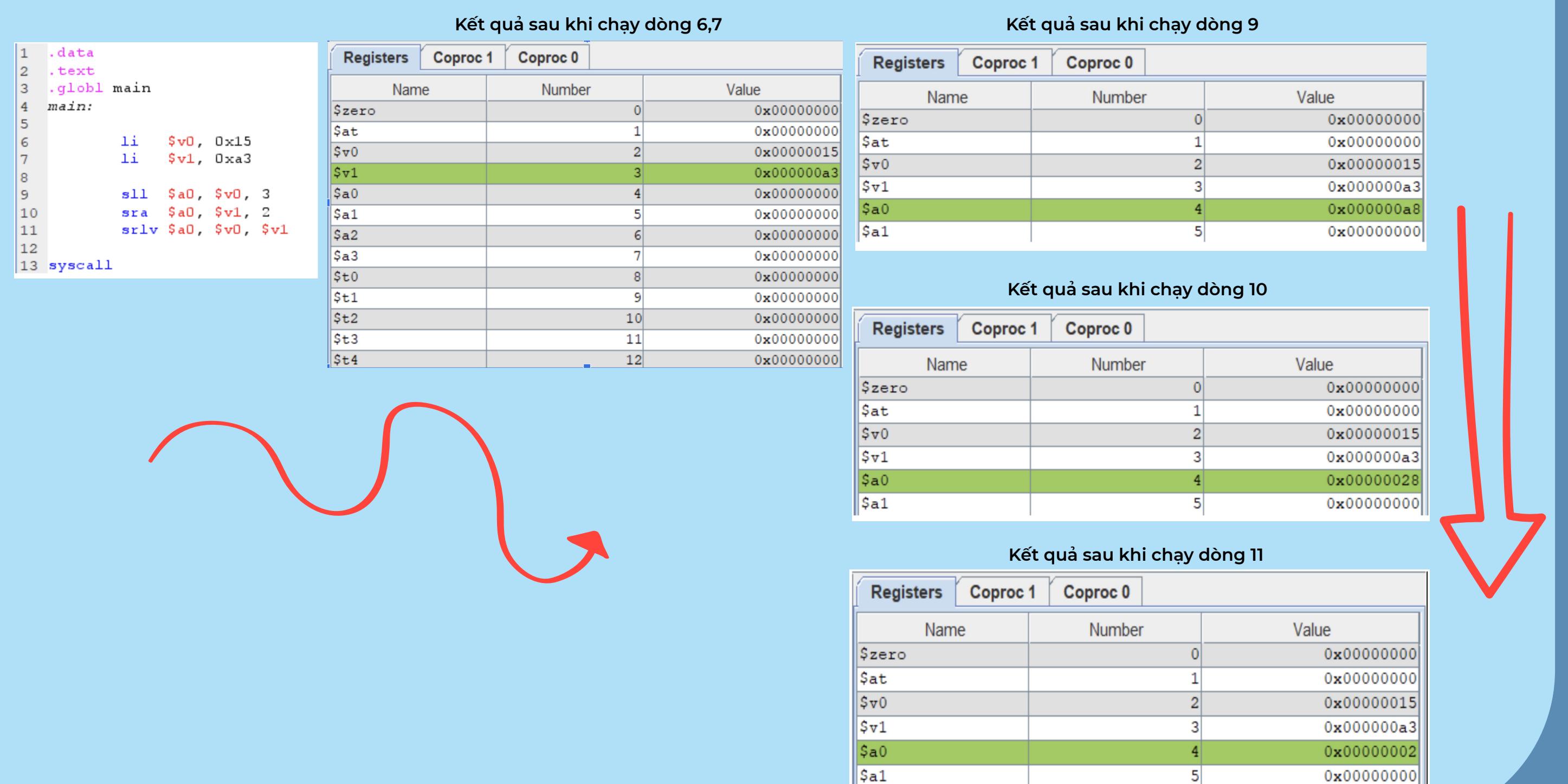
1980 MIPS Technologies Tập trung vào hiệu suất thực thi

Tiêu chuẩn trong Kiến trúc máy tính Ứng dụng rộng rãi trong máy tính cá nhân

PHÉP DỊCH LOGIC

Là phép dịch chuyển các bit qua lại từ trái sang phải hoặc từ phải sang trái, sau khi dịch chuyển sẽ thêm các bit 0 hoặc 1 tùy theo phép toán sử dụng

CÚ PHÁP CÂU LỆNH	OP	RD	RS	SHAMT	FUNCT	PHÉP TOÁN	
						THANH GHI TOÁN HẠNG ĐÍCH	THANH GHI TOÁN HẠNG NGUỒN
BẢNG MỘT SỐ LỆNH DỊCH LOGIC							
	SLL	Dịch trái logic, các bit trống được gán bit 0 Ví dụ: gán giá trị 0x005 vào thanh ghi \$v0; sau khi chạy câu lệnh "sll \$v1, \$v0, 2", số bit của thanh ghi \$v0 dịch trái 2 bit và kết quả 0x014 được lưu vào thanh ghi \$v1					
	SRL	Dịch phải logic, các bit trống được gán bit 0 Ví dụ: gán giá trị 0x005 vào thanh ghi \$v0; sau khi chạy câu lệnh "srl \$v1, \$v0, 2", số bit của thanh ghi \$v0 dịch phải 2 bit và kết quả 0x001 được lưu vào thanh ghi \$v1 \$v0 dịch trái 2 bit và kết quả 0x014 được lưu vào thanh ghi \$v1					
ví dụ:	SRA	Dịch phải logic, các bit trống được gán cùng bit với bit đầu trước khi dịch Ví dụ: gán giá trị 0xffff vào thanh ghi \$v0; sau khi chạy lệnh "sra \$v1, \$v0, 2", số bit của thanh ghi \$v0 dịch phải 2 bit, các bit trống gán cùng bit với bit đầu của 0xffff và kết quả 0xffd được lưu vào thanh ghi \$v1					



ỨNG DỤNG

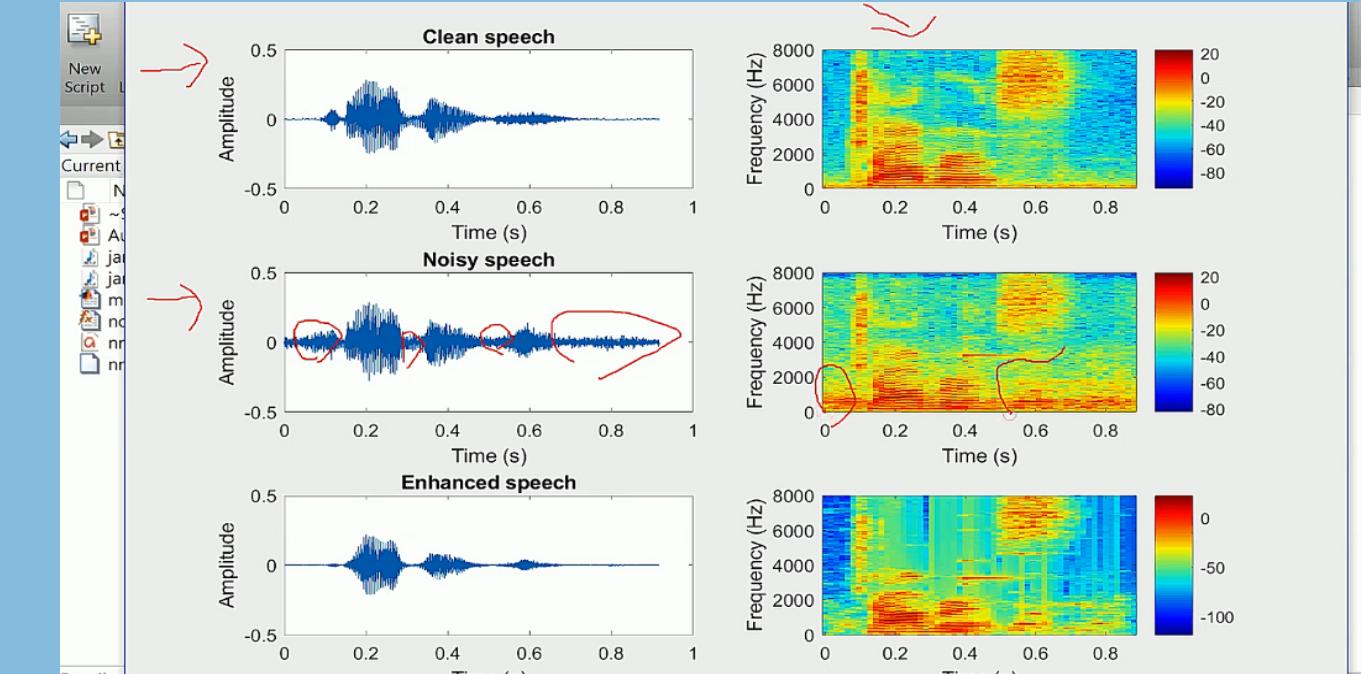
Xử lý dữ liệu bit-level, chẳng hạn như mã hóa/giải mã dữ liệu.



Chuyển đổi giữa kiểu dữ liệu nguyên (integer) và kiểu dữ liệu logic (boolean).

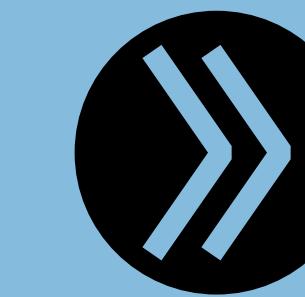


Xử lý dữ liệu nhiễu (noise) trong vi xử lý tín hiệu.

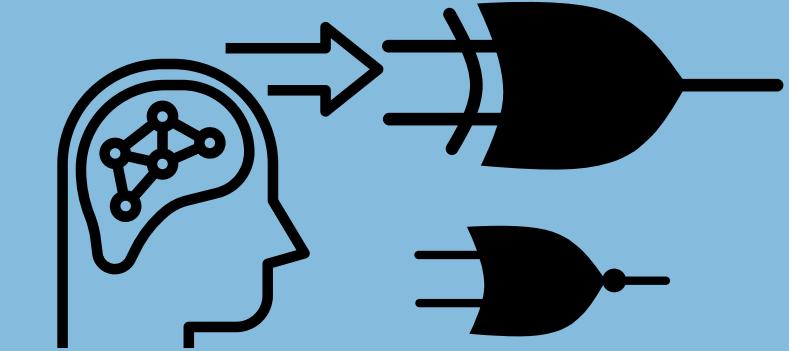


Phân loại

PHÉP DỊCH LOGIC



PHÉP TOÁN LOGIC



PHÉP TOÁN LOGIC

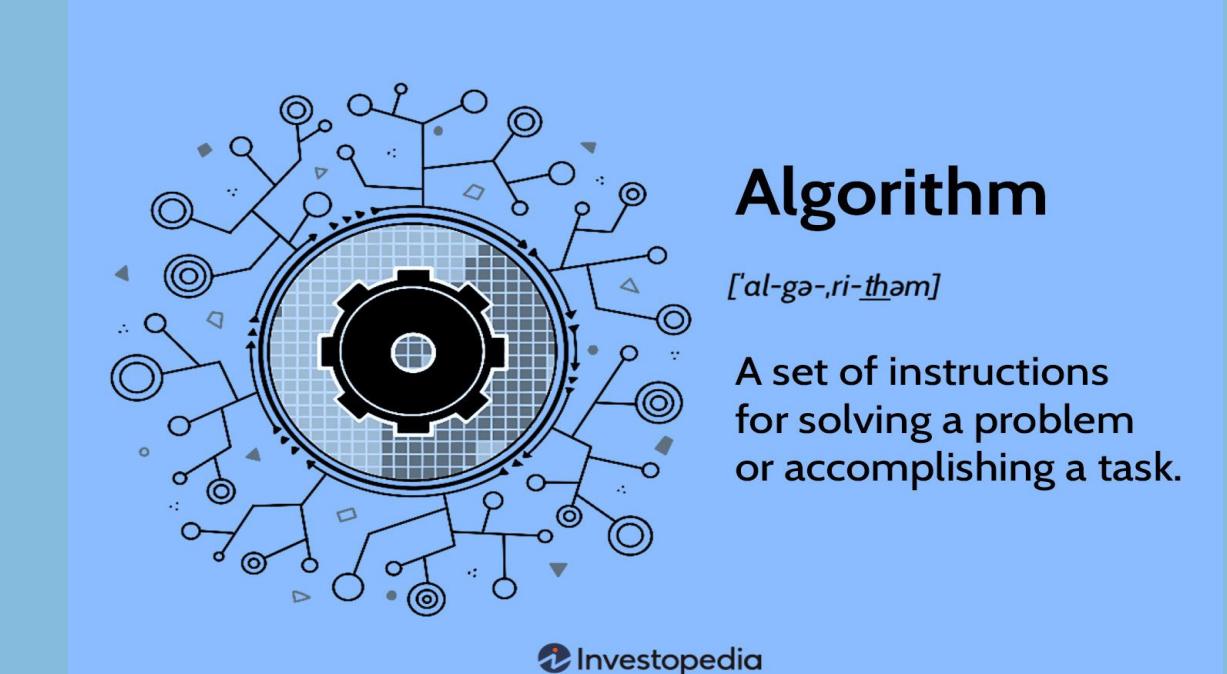
Cho phép tính toán của hai dãy bit giữa hai thanh ghi và lưu kết quả vào thanh thứ ba

CÚ PHÁP CÂU LỆNH	OP	RD	RS	RT
	PHÉP TOÁN	THANH GHI TOÁN HẠNG ĐÍCH	THANH GHI TOÁN HẠNG NGUỒN	THANH GHI THỨ HAI HOẶC HẰNG SỐ
BẢNG MỘT SỐ PHÉP TOÁN LOGIC	A	B	A and B	A or B
	0	0	0	0
	0	1	0	1
	1	0	0	1
	1	1	1	1

note: đảo ngược bit 0 thành 1 và ngược lại
Các phép toán ori, andi, xorl, norl, nandi: tương tự các phép toán trên, nhưng thanh ghi rt là 1 hằng số thay vì ô nhớ

VỊ ĐỊA:	AND	OR	XOR	NOR	NAND																																																																																																																																																													
	<table border="1"><thead><tr><th>Text Segment</th><th>Btgt Address</th><th>Code</th><th>Basic</th><th>Source</th></tr></thead><tbody><tr><td></td><td>0x04000000</td><td>0x24020011add1a 23,25,0x0011</td><td>61 11 \$v0, \$v1</td><td>\$zero</td><td>0 0x00000000</td></tr><tr><td></td><td>0x04000000</td><td>0x24020011add1a 23,25,0x0011</td><td>71 11 \$v1, \$v0</td><td>\$at</td><td>1 0x00000001</td></tr><tr><td></td><td>0x04000000</td><td>0x24020011add1a 23,25,0x0011</td><td>81 11 \$v0, \$v0, \$v1</td><td>\$v0</td><td>2 0x00000002</td></tr><tr><td></td><td>0x04000000</td><td>0x24020011add1a 23,25,0x0011</td><td>91 11 \$v1, \$v1</td><td>\$v1</td><td>3 0x00000003</td></tr><tr><td></td><td>0x04000000</td><td>0x432024and 54,52,23</td><td>61 and \$v0, \$v0, \$v1</td><td>\$and</td><td>4 0x00000004</td></tr><tr><td></td><td>0x04000000</td><td>0x432024and 54,52,23</td><td>71 and \$v1, \$v0, \$v1</td><td>\$and</td><td>5 0x00000005</td></tr><tr><td></td><td>0x04000000</td><td>0x432024and 54,52,23</td><td>81 and \$v0, \$v0, \$v1</td><td>\$and</td><td>6 0x00000006</td></tr><tr><td></td><td>0x04000000</td><td>0x432024and 54,52,23</td><td>91 and \$v1, \$v1</td><td>\$and</td><td>7 0x00000007</td></tr><tr><td></td><td>0x04000000</td><td>0x432024or 54,52,23</td><td>61 or \$v0, \$v0, \$v1</td><td>\$or</td><td>8 0x00000008</td></tr><tr><td></td><td>0x04000000</td><td>0x432024or 54,52,23</td><td>71 or \$v1, \$v0, \$v1</td><td>\$or</td><td>9 0x00000009</td></tr><tr><td></td><td>0x04000000</td><td>0x432024or 54,52,23</td><td>81 or \$v0, \$v0, \$v1</td><td>\$or</td><td>10 0x0000000a</td></tr><tr><td></td><td>0x04000000</td><td>0x432024or 54,52,23</td><td>91 or \$v1, \$v1</td><td>\$or</td><td>11 0x0000000b</td></tr><tr><td></td><td>0x04000000</td><td>0x432024xor 54,52,23</td><td>61 xor \$v0, \$v0, \$v1</td><td>\$xor</td><td>12 0x0000000c</td></tr><tr><td></td><td>0x04000000</td><td>0x432024xor 54,52,23</td><td>71 xor \$v1, \$v0, \$v1</td><td>\$xor</td><td>13 0x0000000d</td></tr><tr><td></td><td>0x04000000</td><td>0x432024xor 54,52,23</td><td>81 xor \$v0, \$v0, \$v1</td><td>\$xor</td><td>14 0x0000000e</td></tr><tr><td></td><td>0x04000000</td><td>0x432024xor 54,52,23</td><td>91 xor \$v1, \$v1</td><td>\$xor</td><td>15 0x0000000f</td></tr><tr><td></td><td>0x04000000</td><td>0x432024nori 54,52,23</td><td>61 nori \$v0, \$v0, \$v1</td><td>\$nori</td><td>16 0xffffffff</td></tr><tr><td></td><td>0x04000000</td><td>0x432024nori 54,52,23</td><td>71 nori \$v1, \$v0, \$v1</td><td>\$nori</td><td>17 0xffffffff</td></tr><tr><td></td><td>0x04000000</td><td>0x432024nori 54,52,23</td><td>81 nori \$v0, \$v0, \$v1</td><td>\$nori</td><td>18 0xffffffff</td></tr><tr><td></td><td>0x04000000</td><td>0x432024nori 54,52,23</td><td>91 nori \$v1, \$v1</td><td>\$nori</td><td>19 0xffffffff</td></tr><tr><td></td><td>0x04000000</td><td>0x432024nand 54,52,23</td><td>61 nand \$v0, \$v0, \$v1</td><td>\$nand</td><td>20 0xffffffff</td></tr><tr><td></td><td>0x04000000</td><td>0x432024nand 54,52,23</td><td>71 nand \$v1, \$v0, \$v1</td><td>\$nand</td><td>21 0xffffffff</td></tr><tr><td></td><td>0x04000000</td><td>0x432024nand 54,52,23</td><td>81 nand \$v0, \$v0, \$v1</td><td>\$nand</td><td>22 0xffffffff</td></tr><tr><td></td><td>0x04000000</td><td>0x432024nand 54,52,23</td><td>91 nand \$v1, \$v1</td><td>\$nand</td><td>23 0xffffffff</td></tr><tr><td></td><td>0x04000000</td><td>0x432024naycall</td><td>101 syscall</td><td>\$syscall</td><td>24 0x00000000</td></tr><tr><td></td><td>0x04000000</td><td>0x432024naycall</td><td>111 syscall</td><td>\$syscall</td><td>25 0x00000000</td></tr></tbody></table>	Text Segment	Btgt Address	Code	Basic	Source		0x04000000	0x24020011add1a 23,25,0x0011	61 11 \$v0, \$v1	\$zero	0 0x00000000		0x04000000	0x24020011add1a 23,25,0x0011	71 11 \$v1, \$v0	\$at	1 0x00000001		0x04000000	0x24020011add1a 23,25,0x0011	81 11 \$v0, \$v0, \$v1	\$v0	2 0x00000002		0x04000000	0x24020011add1a 23,25,0x0011	91 11 \$v1, \$v1	\$v1	3 0x00000003		0x04000000	0x432024and 54,52,23	61 and \$v0, \$v0, \$v1	\$and	4 0x00000004		0x04000000	0x432024and 54,52,23	71 and \$v1, \$v0, \$v1	\$and	5 0x00000005		0x04000000	0x432024and 54,52,23	81 and \$v0, \$v0, \$v1	\$and	6 0x00000006		0x04000000	0x432024and 54,52,23	91 and \$v1, \$v1	\$and	7 0x00000007		0x04000000	0x432024or 54,52,23	61 or \$v0, \$v0, \$v1	\$or	8 0x00000008		0x04000000	0x432024or 54,52,23	71 or \$v1, \$v0, \$v1	\$or	9 0x00000009		0x04000000	0x432024or 54,52,23	81 or \$v0, \$v0, \$v1	\$or	10 0x0000000a		0x04000000	0x432024or 54,52,23	91 or \$v1, \$v1	\$or	11 0x0000000b		0x04000000	0x432024xor 54,52,23	61 xor \$v0, \$v0, \$v1	\$xor	12 0x0000000c		0x04000000	0x432024xor 54,52,23	71 xor \$v1, \$v0, \$v1	\$xor	13 0x0000000d		0x04000000	0x432024xor 54,52,23	81 xor \$v0, \$v0, \$v1	\$xor	14 0x0000000e		0x04000000	0x432024xor 54,52,23	91 xor \$v1, \$v1	\$xor	15 0x0000000f		0x04000000	0x432024nori 54,52,23	61 nori \$v0, \$v0, \$v1	\$nori	16 0xffffffff		0x04000000	0x432024nori 54,52,23	71 nori \$v1, \$v0, \$v1	\$nori	17 0xffffffff		0x04000000	0x432024nori 54,52,23	81 nori \$v0, \$v0, \$v1	\$nori	18 0xffffffff		0x04000000	0x432024nori 54,52,23	91 nori \$v1, \$v1	\$nori	19 0xffffffff		0x04000000	0x432024nand 54,52,23	61 nand \$v0, \$v0, \$v1	\$nand	20 0xffffffff		0x04000000	0x432024nand 54,52,23	71 nand \$v1, \$v0, \$v1	\$nand	21 0xffffffff		0x04000000	0x432024nand 54,52,23	81 nand \$v0, \$v0, \$v1	\$nand	22 0xffffffff		0x04000000	0x432024nand 54,52,23	91 nand \$v1, \$v1	\$nand	23 0xffffffff		0x04000000	0x432024naycall	101 syscall	\$syscall	24 0x00000000		0x04000000	0x432024naycall	111 syscall	\$syscall	25 0x00000000
Text Segment	Btgt Address	Code	Basic	Source																																																																																																																																																														
	0x04000000	0x24020011add1a 23,25,0x0011	61 11 \$v0, \$v1	\$zero	0 0x00000000																																																																																																																																																													
	0x04000000	0x24020011add1a 23,25,0x0011	71 11 \$v1, \$v0	\$at	1 0x00000001																																																																																																																																																													
	0x04000000	0x24020011add1a 23,25,0x0011	81 11 \$v0, \$v0, \$v1	\$v0	2 0x00000002																																																																																																																																																													
	0x04000000	0x24020011add1a 23,25,0x0011	91 11 \$v1, \$v1	\$v1	3 0x00000003																																																																																																																																																													
	0x04000000	0x432024and 54,52,23	61 and \$v0, \$v0, \$v1	\$and	4 0x00000004																																																																																																																																																													
	0x04000000	0x432024and 54,52,23	71 and \$v1, \$v0, \$v1	\$and	5 0x00000005																																																																																																																																																													
	0x04000000	0x432024and 54,52,23	81 and \$v0, \$v0, \$v1	\$and	6 0x00000006																																																																																																																																																													
	0x04000000	0x432024and 54,52,23	91 and \$v1, \$v1	\$and	7 0x00000007																																																																																																																																																													
	0x04000000	0x432024or 54,52,23	61 or \$v0, \$v0, \$v1	\$or	8 0x00000008																																																																																																																																																													
	0x04000000	0x432024or 54,52,23	71 or \$v1, \$v0, \$v1	\$or	9 0x00000009																																																																																																																																																													
	0x04000000	0x432024or 54,52,23	81 or \$v0, \$v0, \$v1	\$or	10 0x0000000a																																																																																																																																																													
	0x04000000	0x432024or 54,52,23	91 or \$v1, \$v1	\$or	11 0x0000000b																																																																																																																																																													
	0x04000000	0x432024xor 54,52,23	61 xor \$v0, \$v0, \$v1	\$xor	12 0x0000000c																																																																																																																																																													
	0x04000000	0x432024xor 54,52,23	71 xor \$v1, \$v0, \$v1	\$xor	13 0x0000000d																																																																																																																																																													
	0x04000000	0x432024xor 54,52,23	81 xor \$v0, \$v0, \$v1	\$xor	14 0x0000000e																																																																																																																																																													
	0x04000000	0x432024xor 54,52,23	91 xor \$v1, \$v1	\$xor	15 0x0000000f																																																																																																																																																													
	0x04000000	0x432024nori 54,52,23	61 nori \$v0, \$v0, \$v1	\$nori	16 0xffffffff																																																																																																																																																													
	0x04000000	0x432024nori 54,52,23	71 nori \$v1, \$v0, \$v1	\$nori	17 0xffffffff																																																																																																																																																													
	0x04000000	0x432024nori 54,52,23	81 nori \$v0, \$v0, \$v1	\$nori	18 0xffffffff																																																																																																																																																													
	0x04000000	0x432024nori 54,52,23	91 nori \$v1, \$v1	\$nori	19 0xffffffff																																																																																																																																																													
	0x04000000	0x432024nand 54,52,23	61 nand \$v0, \$v0, \$v1	\$nand	20 0xffffffff																																																																																																																																																													
	0x04000000	0x432024nand 54,52,23	71 nand \$v1, \$v0, \$v1	\$nand	21 0xffffffff																																																																																																																																																													
	0x04000000	0x432024nand 54,52,23	81 nand \$v0, \$v0, \$v1	\$nand	22 0xffffffff																																																																																																																																																													
	0x04000000	0x432024nand 54,52,23	91 nand \$v1, \$v1	\$nand	23 0xffffffff																																																																																																																																																													
	0x04000000	0x432024naycall	101 syscall	\$syscall	24 0x00000000																																																																																																																																																													
	0x04000000	0x432024naycall	111 syscall	\$syscall	25 0x00000000																																																																																																																																																													

Thực hiện các phép toán logic để xây dựng các thuật toán phức tạp hơn.



STORM TEAM