Craig Shaffer

Dr. Swami Ponpondi

CPSC 300

2-18-2022

Homework 5

1. Draw the memory layout for the word-sized (all elements are 4 byte wide) list, pixel = $\{0x1000, 0x03, 0x56DD23FF\}$. The base address is 0x1FC0. [10]

	Jana	00 00 10 00 00 00 00 00 3 56 00 23 FF		
1	Memory layout for Pixel = { 6x 1600, 0x03,0x56 DD 23 FF} base address = 0x1FC0 Big Endan			
	ox 0 (-Byte Wide-)			
	0x 1FCO 0x 00	Agrical and the property		
	oxIFCI ox 00			
	OXIFC2 10			
	OKITCS OC			
	OKIFUS OB	" and seem complete to the seeplest to a second to the sec		
50.°	ox/FC	an takanika		
	ONIFCE OF OO			
100	OXIFCT OXO3			
	OXIFCE OX 56	AND THE RESERVE TO THE PARTY OF		
	ox1Fc9 anD			
	OXIFEA CX 23			
	ONIFCB ON FF			

2. Use minimum number of MIPS instructions (add, sub : see page 64 of textbook) and write the assembly code for the below Java/Python code expression. Assume all the variables are 32-bit data. You may choose any registers to hold the variable contents. [10 points] x1 = (r + (t + r)) - y + 5; y = x1 + y

2 Mips code for	Assame the variables are in	
X = (r + (++r)) - x +5	The following registers:	
Y= X1+9	x1 → \$50 + → \$52	
	r > 851 Y > \$53	
add \$52, \$52, \$51 # (++r)		
	(tir)) > \$51	
	+r))-y)→\$s(
	(++r))-y)+5 →\$50	
add \$53', \$50, \$53 # (x1+	x) → \$53	

3. Translate the Python code, A[5] = A[4] * 4 + B[3], into MIPS assembly code. You may assume any register for base addresses and variables. [15 points]

		1.09
3	translate to mips	1 . morell 1
	translate to mips A[5] = A[4] * 4 B[3]	
	12 (2) (2)	
• *	Assume A base address in \$50	-> 10 8
	Assume B base address in \$10	
W 4	Assume A + B are word gized lists 4 whe	1 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
8	Start Porc Marges	112740
	1 \$51, 16 (\$50) # load A[4] into \$	51 offset=4×4-16
	16) \$+1, 12(\$+0) # lead B[3] into \$	+1 off set=3x4=12
	511 \$ 51 , \$\$1 , 2 # A[4] x 4 = A[4] x22-	> \$51
	add \$52, \$51, \$+1 # ((A[4]x4) + B[3])->	\$52
	SU \$52, 20 (\$50) # store \$52 in A[6]	Aset - 514220
	S. S.	