

Lab Report-10

ECPE 170 – Computer Systems and Networks – Fall 2021

Name: Vivek Kumar Maheshwari

Lab Topic: MIPS Assembly Programming(Basic) (Lab #: 10)

Question #1: Take two screenshots of the MIPS register panel: one before your program runs, and one after your program finishes. Put the register panel in Decimal mode (right-click) so it is easy to see register values.

Answer:

FP Regs	Int Regs [10]	Data	Text
Int Regs [10]			
R0 [r0] = 0 R1 [at] = 0 R2 [v0] = 0 R3 [v1] = 0 R4 [a0] = 1 R5 [a1] = 2147480408 R6 [a2] = 2147480416 R7 [a3] = 0 R8 [t0] = 0 R9 [t1] = 0 R10 [t2] = 0 R11 [t3] = 0 R12 [t4] = 0 R13 [t5] = 0 R14 [t6] = 0 R15 [t7] = 0 R16 [s0] = 0 R17 [s1] = 0 R18 [s2] = 0 R19 [s3] = 0 R20 [s4] = 0 R21 [s5] = 0 R22 [s6] = 0 R23 [s7] = 0 R24 [t8] = 0 R25 [t9] = 0 R26 [x0] = 0 R27 [x1] = 0 R28 [gp] = 268468224 R29 [sp] = 2147480404 R30 [s8] = 0 R31 [ra] = 0	<pre> User Text Segment [00400000]..[00440000] [00400000] 8f840000 lw \$4, 0(\$29) ; 183: lw \$a0 \$0(\$sp) # argc [00400004] 27a50004 addiu \$5, \$29, 4 ; 184: addiu \$a1 \$sp 4 # argv [00400008] 24a60004 addiu \$6, \$29, 4 ; 185: addiu \$a2 \$a1 4 # envp [0040000c] 00041080 sll \$2, \$4, 2 ; 186: sll \$v0 \$a0 2 [00400010] 00c23021 addu \$6, \$6, \$2 ; 187: addu \$a2 \$a2 \$v0 [00400014] 0c100009 jal 0x00400024 [main] ; 188: jal main [00400018] 00000000 nop ; 189: nop [00400020] 0000000c syscall ; 190: li \$v0 10 [00400024] 2008000f addi \$8, \$0, 15 ; 192: syscall # syscall 10 (exit) [00400028] 2009000a addi \$9, \$0, 10 ; 193: addi \$t1,\$zero,10 [0040002c] 200a0007 addi \$10, \$0, 7 ; 194: addi \$t2,\$zero,7 [00400030] 200b0002 addi \$11, \$0, 2 ; 195: addi \$t3,\$zero,2 [00400034] 200c0012 addi \$12, \$0, 18 ; 196: addi \$t4,\$zero,18 [00400038] 200dfffd addi \$13, \$0, -3 ; 197: addi \$t5,\$zero,-3 [0040003c] 01097020 add \$14, \$8, \$9 ; 198: add \$t6,\$t0,\$t1 [00400040] 014b7822 sub \$15, \$10, \$11 ; 199: sub \$t7,\$t2,\$t3 [00400044] 018dc020 add \$24, \$12, \$13 ; 200: sub \$t8,\$t4,\$t5 [00400048] 010ac822 sub \$25, \$8, \$10 ; 201: add \$t9,\$t0,\$t2 [0040004c] 01cf8020 add \$16, \$14, \$15 ; 202: add \$t10,\$t6,\$t7 [00400050] 03198822 sub \$17, \$24, \$25 ; 203: sub \$t11,\$t8,\$t9 [00400054] 02119020 add \$18, \$16, \$17 ; 204: add \$t12,\$t9,\$t11 [00400058] 3c011001 lui \$1, 4097 [Z] ; 205: sw \$v0 \$a0 10 # Sets \$v0 to "10" to select exit syscall [0040005c] ac320000 sw \$18, 0(\$1) ; 206: sw \$s0 \$s1 # But we need to use these registers [00400060] 3402000a ori \$2, \$0, 10 ; 207: add \$s2,\$s0,\$s1 [00400064] 0000000c syscall ; 208: sw \$s2,Z ; 209: syscall # Exit Kernel Text Segment [80000000]..[80010000] [80000180] 0001d821 addu \$27, \$0, \$1 ; 90: move \$k1 \$at # Save \$at [80000184] 3c019000 lui \$1, -28672 ; 92: sw \$v0 \$s1 # Not re-entrant and we can't trust \$sp [80000188] ac220020 sw \$2, \$12(\$1) ; 93: sw \$s0 \$s2 # But we need to use these registers [8000018c] 3c019000 lui \$1, -28672 ; 94: mfc0 \$k0 \$t3 # Cause register [80000190] ac240204 sw \$4, \$16(\$1) ; 95: mfc0 \$k0 \$t3 # Cause register </pre>		

FP Regs	Int Regs [10]	Data	Text
Int Regs [10]			
R0 [r0] = 0 R1 [at] = 268500992 R2 [v0] = 10 R3 [v1] = 0 R4 [a0] = 1 R5 [a1] = 2147480408 R6 [a2] = 2147480416 R7 [a3] = 0 R8 [t0] = 15 R9 [t1] = 10 R10 [t2] = 7 R11 [t3] = 2 R12 [t4] = 18 R13 [t5] = -3 R14 [t6] = 25 R15 [t7] = 5 R16 [s0] = 30 R17 [s1] = 7 R18 [s2] = 37 R19 [s3] = 0 R20 [s4] = 0 R21 [s5] = 0 R22 [s6] = 0 R23 [s7] = 0 R24 [t8] = 15 R25 [t9] = 6 R26 [x0] = 0 R27 [x1] = 0 R28 [gp] = 268468224 R29 [sp] = 2147480404 R30 [s8] = 0 R31 [ra] = 4194328	<pre> User Text Segment [00400000]..[00440000] [00400000] 8fa40000 lw \$4, 0(\$29) ; 183: lw \$a0 \$0(\$sp) # argc [00400004] 27a50004 addiu \$5, \$29, 4 ; 184: addiu \$a1 \$sp 4 # argv [00400008] 24a60004 addiu \$6, \$29, 4 ; 185: addiu \$a2 \$a1 4 # envp [0040000c] 00041080 sll \$2, \$4, 2 ; 186: sll \$v0 \$a0 2 [00400010] 00c23021 addu \$6, \$6, \$2 ; 187: addu \$a2 \$a2 \$v0 [00400014] 0c100009 jal 0x00400024 [main] ; 188: jal main [00400018] 00000000 nop ; 189: nop [00400020] 0000000c syscall ; 190: li \$v0 10 [00400024] 2008000f addi \$8, \$0, 15 ; 192: syscall # syscall 10 (exit) [00400028] 2009000a addi \$9, \$0, 10 ; 193: addi \$t1,\$zero,10 [0040002c] 200a0007 addi \$10, \$0, 7 ; 194: addi \$t2,\$zero,7 [00400030] 200b0002 addi \$11, \$0, 2 ; 195: addi \$t3,\$zero,2 [00400034] 200c0012 addi \$12, \$0, 18 ; 196: addi \$t4,\$zero,18 [00400038] 200dfffd addi \$13, \$0, -3 ; 197: addi \$t5,\$zero,-3 [0040003c] 01097020 add \$14, \$8, \$9 ; 198: add \$t6,\$t0,\$t1 [00400040] 014b7822 sub \$15, \$10, \$11 ; 199: sub \$t7,\$t2,\$t3 [00400044] 018dc020 add \$24, \$12, \$13 ; 200: sub \$t8,\$t4,\$t5 [00400048] 010ac822 sub \$25, \$8, \$10 ; 201: add \$t9,\$t0,\$t2 [0040004c] 01cf8020 add \$16, \$14, \$15 ; 202: add \$t10,\$t6,\$t7 [00400050] 03198822 sub \$17, \$24, \$25 ; 203: sub \$t11,\$t8,\$t9 [00400054] 02119020 add \$18, \$16, \$17 ; 204: add \$t12,\$t9,\$t11 [00400058] 3c011001 lui \$1, 4097 [Z] ; 205: sw \$v0 \$a0 10 # Sets \$v0 to "10" to select exit syscall [0040005c] ac320000 sw \$18, 0(\$1) ; 206: sw \$s0 \$s1 # But we need to use these registers [00400060] 3402000a ori \$2, \$0, 10 ; 207: add \$s2,\$s0,\$s1 [00400064] 0000000c syscall ; 208: sw \$s2,Z ; 209: syscall # Exit Kernel Text Segment [80000000]..[80010000] [80000180] 0001d821 addu \$27, \$0, \$1 ; 90: move \$k1 \$at # Save \$at [80000184] 3c019000 lui \$1, -28672 ; 92: sw \$v0 \$s1 # Not re-entrant and we can't trust \$sp [80000188] ac220020 sw \$2, \$12(\$1) ; 93: sw \$s0 \$s2 # But we need to use these registers [8000018c] 3c019000 lui \$1, -28672 ; 94: mfc0 \$k0 \$t3 # Cause register [80000190] ac240204 sw \$4, \$16(\$1) ; 95: mfc0 \$k0 \$t3 # Cause register </pre>		

Question #2: Take two screenshots of the MIPS memory panel (data tab): one before your program runs, and one after your program finishes. Put the memory panel in Decimal mode (right-click), so it is easy to see memory values. In the after-execution capture, circle the memory location (not register) that contains the final calculated value of Z.

Answer:

FP Regs	nt Regs [10]	Data	Text
nt Regs [10]		Data	
PC	= 0		User data segment [10000000]..[10040000]
EPC	= 0		[10000000]..[1003ffff] 00000000
Cause	= 0		
BadAddr	= 0		
Status	= 805371664		
HI	= 0		User Stack [7fffff354]..[80000000]
LO	= 0		[7fffff354] 000000001 2147480617 000000000
R0 [r0]	= 0		[7fffff360] 2147483632 2147483550 2147483531 2147483511
R1 [at]	= 0		[7fffff370] 2147483466 2147483443 2147483399 2147483367
R2 [v0]	= 0		[7fffff380] 2147483326 2147483306 2147483283 2147483264
R3 [v1]	= 0		[7fffff390] 2147483236 2147483215 2147483196 2147483169
R4 [a0]	= 1		[7fffff3a0] 2147483148 2147483096 2147483055 2147483022
R5 [a1]	= 2147480408		[7fffff3b0] 2147483009 2147482987 2147482967 2147482949
R6 [a2]	= 2147480416		[7fffff3c0] 2147482932 2147481426 2147481393 2147481376
R7 [a3]	= 0		[7fffff3d0] 2147481290 2147481243 2147481227 2147481203
R8 [t0]	= 0		[7fffff3e0] 2147481169 2147481146 2147481126 2147481094
R9 [t1]	= 0		[7fffff3f0] 2147481078 2147481048 2147481037 2147481029
R10 [t2]	= 0		[7fffff400] 2147481011 2147480980 2147480957 2147480872
R11 [t3]	= 0		[7fffff410] 2147480768 2147480750 2147480696 2147480678
R12 [t4]	= 0		[7fffff420] 0000000000 0000000000 1869098752 1982817645
R13 [t5]	= 0		[7fffff430] 1801811561 1634563435 1768042354 1668637300
R14 [t6]	= 0		i v e k k u m a r / b i t b u c
R15 [t7]	= 0		k e t / 2 0 2 1 _ f a i l _ e c
R16 [s0]	= 0		p e 1 7 0 / 1 a b 1 0 / p a r t
R17 [s1]	= 0		1 . a s m . _ = / u s r / b i n
R18 [s2]	= 0		/ q t s p i m . D B U S _ S E S S
R19 [s3]	= 0		S I O N _ B U S _ A D D R E S S
R20 [s4]	= 0		= u n i x : p a t h = / r u n /
R21 [s5]	= 0		u s e r / 1 0 0 0 / b u s . G D
R22 [s6]	= 0		M S E S S I O N = u b u n t u .
R23 [s7]	= 0		P A T H = / u s r / l o c a l /

FP Regs	nt Regs [10]	Data	Text
Int Regs [10]		Data	
R0 [r0]	= 0		User data segment [10000000]..[10040000]
R1 [at]	= 268500992		[10000000]..[1000ffff] 00000000
R2 [v0]	= 10		[10010000] 0000000037 0000000000 0000000000 0000000000 %
R3 [v1]	= 0		[10010010]..[1003ffff] 00000000
R4 [a0]	= 1		
R5 [a1]	= 2147480408		
R6 [a2]	= 2147480416		
R7 [a3]	= 0		User Stack [7fffff354]..[80000000]
R8 [t0]	= 15		[7fffff354] 000000001 2147480617 0000000000
R9 [t1]	= 10		[7fffff360] 2147483632 2147483550 2147483531 2147483511
R10 [t2]	= 7		[7fffff370] 2147483466 2147483443 2147483399 2147483367
R11 [t3]	= 2		[7fffff380] 2147483326 2147483306 2147483283 2147483264
R12 [t4]	= 18		[7fffff390] 2147483236 2147483215 2147483196 2147483169
R13 [t5]	= -3		[7fffff3a0] 2147483148 2147483096 2147483055 2147483022
R14 [t6]	= 25		[7fffff3b0] 2147483009 2147482987 2147482967 2147482949
R15 [t7]	= 5		[7fffff3c0] 2147482932 2147481426 2147481393 2147481376
R16 [s0]	= 30		[7fffff3d0] 2147481290 2147481243 2147481227 2147481203
R17 [s1]	= 7		[7fffff3e0] 2147481169 2147481146 2147481126 2147481094
R18 [s2]	= 37		[7fffff3f0] 2147481078 2147481048 2147481037 2147481029
R19 [s3]	= 0		[7fffff400] 2147481011 2147480980 2147480957 2147480872
R20 [s4]	= 0		[7fffff410] 2147480768 2147480750 2147480696 2147480678
R21 [s5]	= 0		[7fffff420] 0000000000 0000000000 1869098752 1982817645
R22 [s6]	= 0		[7fffff430] 1801811561 1634563435 1768042354 1668637300
R23 [s7]	= 0		i v e k k u m a r / b i t b u c
R24 [t8]	= 15		k e t / 2 0 2 1 _ f a i l _ e c
R25 [t9]	= 8		p e 1 7 0 / 1 a b 1 0 / p a r t
R26 [k0]	= 0		1 . a s m . _ = / u s r / b i n
R27 [k1]	= 0		/ q t s p i m . D B U S _ S E S S
R28 [gp]	= 268468224		S I O N _ B U S _ A D D R E S S
R29 [sp]	= 2147480404		= u n i x : p a t h = / r u n /
R30 [s8]	= 0		u s e r / 1 0 0 0 / b u s . G D
R31 [ra]	= 4194328		M S E S S I O N = u b u n t u .

Question 3: Take two screenshots of the MIPS register panel: one before your program runs, and one after your program finishes. Put the register panel in Decimal mode (right-click) so it is easy to see register values.

Answer:

FP Regs	Int Regs [10]	Data	Text
Int Regs [10]		Text	<pre>R0 [r0] = 0 R1 [at] = 0 R2 [v0] = 0 R3 [v1] = 0 R4 [a0] = 1 R5 [a1] = 2147480348 R6 [a2] = 2147480356 R7 [a3] = 0 R8 [t0] = 0 R9 [t1] = 0 R10 [t2] = 0 R11 [t3] = 0 R12 [t4] = 0 R13 [t5] = 0 R14 [t6] = 0 R15 [t7] = 0 R16 [w0] = 0 R17 [s1] = 0 R18 [s2] = 0 R19 [s3] = 0 R20 [s4] = 0 R21 [s5] = 0 R22 [s6] = 0 R23 [s7] = 0 R24 [t8] = 0 R25 [t9] = 0 R26 [k0] = 0 R27 [k1] = 0 R28 [gp] = 268468224 R29 [sp] = 2147480344 R30 [s8] = 0 R31 [ra] = 0 [00400000] 8fa40000 lw \$4, 0(\$29) [00400004] 27a50004 addiu \$5, \$29, 4 [00400008] 24a60004 addiu \$6, \$5, 4 [00400010] 00c23021 addu \$6, \$6, \$2 [00400014] 0c100009 jal 0x00400024 [main] [00400018] 00000000 nop [0040001c] 34020000 ori \$2, \$0, 10 [00400020] 00000000 syscall [00400024] 3c011001 lui \$1, 4097 [A] [00400028] 8c300000 lw \$16, 0(\$1) [A] [0040002c] 3c011001 lui \$1, 4097 [B] [00400030] 8c310004 lw \$17, 4(\$1) [B] [00400034] 3c011001 lui \$1, 4097 [C] [00400038] 8c320008 lw \$18, 8(\$1) [C] [0040003c] 0230082a slt \$1, \$17, \$16 [00400040] 14200000 bne \$1, \$0, 44 [label1-0x00400040] [00400044] 20080005 addi \$8, \$0, 5 [00400048] 0248002a slt \$1, \$18, \$8 [0040004c] 14200008 bne \$1, \$0, 32 [label1-0x0040004c] [00400050] 0230082a slt \$1, \$17, \$16 [00400054] 14200003 bne \$1, \$0, 12 [label1-0x00400054] [00400058] 20130003 addi \$19, \$0, 3 [0040005c] 0810001f j 0x0040007c [label4] [00400060] 224a0000 addi \$10, \$18, 1 [00400064] 20900007 addi \$9, \$0, 7 [00400068] 11490003 beq \$10, \$9, 12 [label2-0x00400068] [0040006c] 20130001 addi \$19, \$0, 1 [00400070] 0810001f j 0x0040007c [label4] [00400074] 20130000 addi \$19, \$0, 2 [00400078] 0810001f j 0x0040007c [label4] [0040007c] 200b0001 addi \$11, \$0, 1 [00400080] 200c0002 addi \$12, \$0, 2 [00400084] 126h0004 beq \$19, \$11, 16 [label5-0x00400084]</pre>

FP Regs	Int Regs [10]	Data	Text
Int Regs [10]		Text	<pre>R0 [r0] = 0 R1 [at] = 268500992 R2 [v0] = 10 R3 [v1] = 0 R4 [a0] = 1 R5 [a1] = 2147480348 R6 [a2] = 2147480356 R7 [a3] = 0 R8 [t0] = 5 R9 [t1] = 0 R10 [t2] = 0 R11 [t3] = 1 R12 [t4] = 2 R13 [t5] = 0 R14 [t6] = 0 R15 [t7] = 0 R16 [s0] = 10 R17 [s1] = 15 R18 [s2] = 6 R19 [s3] = 3 R20 [s4] = 0 R21 [s5] = 0 R22 [s6] = 0 R23 [s7] = 0 R24 [t8] = 0 R25 [t9] = 0 R26 [k0] = 0 R27 [k1] = 0 R28 [gp] = 268468224 R29 [sp] = 2147480344 R30 [s8] = 0 R31 [ra] = 4194328 [00400000] 8fa40000 lw \$4, 0(\$29) [00400004] 27a50004 addiu \$5, \$29, 4 [00400008] 24a60004 addiu \$6, \$5, 4 [00400010] 00c23021 addu \$6, \$6, \$2 [00400014] 0c100009 jal 0x00400024 [main] [00400018] 00000000 nop [0040001c] 34020000 ori \$2, \$0, 10 [00400020] 00000000 syscall [00400024] 3c011001 lui \$1, 4097 [A] [00400028] 8c300000 lw \$16, 0(\$1) [A] [0040002c] 3c011001 lui \$1, 4097 [B] [00400030] 8c310004 lw \$17, 4(\$1) [B] [00400034] 3c011001 lui \$1, 4097 [C] [00400038] 8c320008 lw \$18, 8(\$1) [C] [0040003c] 0230082a slt \$1, \$17, \$16 [00400040] 1420000b bne \$1, \$0, 44 [label1-0x00400040] [00400044] 20080005 addi \$8, \$0, 5 [00400048] 0248002a slt \$1, \$18, \$8 [0040004c] 14200008 bne \$1, \$0, 32 [label1-0x0040004c] [00400050] 0230082a slt \$1, \$17, \$16 [00400054] 14200003 bne \$1, \$0, 12 [label1-0x00400054] [00400058] 20130003 addi \$19, \$0, 3 [0040005c] 0810001f j 0x0040007c [label4] [00400060] 224a0001 addi \$10, \$18, 1 [00400064] 20900007 addi \$9, \$0, 7 [00400068] 11490003 beq \$10, \$9, 12 [label2-0x00400068] [0040006c] 20130001 addi \$19, \$0, 1 [00400070] 0810001f j 0x0040007c [label4] [00400074] 20130000 addi \$19, \$0, 2 [00400078] 0810001f j 0x0040007c [label4] [0040007c] 200b0001 addi \$11, \$0, 1 [00400080] 200c0002 addi \$12, \$0, 2 [00400084] 126h0004 beq \$19, \$11, 16 [label5-0x00400084]</pre>

Question 4: Take two screenshots of the MIPS memory panel (data tab): one before your program runs, and one after your program finishes. Put the memory panel in Decimal mode (right-click), so it is easy to see memory values. In the after-execution capture, circle the memory location (not register) that contains the final calculated value of Z.

Answer:

FP Regs	nt Regs [10]	Data	Text
Int Regs [10]		Data	
PC = 0		User data segment [10000000]..[10040000]	
EPC = 0		[10000000]..[100fffff] 00000000	
Cause = 0		[10010000] 0000000010 0000000015 0000000006 0000000000	
BadVAddr = 0		[10010010]..[1003ffff] 00000000	
Status = 805371664			
HI = 0		User Stack [7fffff318]..[80000000]	
LO = 0		[7fffff318] 0000000001 2147480559	
R0 [r0] = 0		[7fffff320] 0000000000 2147483632 2147483550 2147483531	
R1 [at] = 0		[7fffff330] 2147483511 2147483466 2147483443 2147483399	w . . . J . . . 3
R2 [v0] = 0		[7fffff340] 2147483367 2147483326 2147483306 2147483283 d . . . O . . . < . . .
R3 [v1] = 0		[7fffff350] 2147483264 2147483236 2147483215 2147483196	! k . . . W
R4 [a0] = 1		[7fffff360] 2147483169 2147483148 2147483096 2147483055	E . . . 4 . . . R . . . 1
R5 [a1] = 2147480348		[7fffff370] 2147483022 2147483009 2147482987 2147482967 s . . . Q . . . : . . . & . . .
R6 [a2] = 2147480356		[7fffff380] 2147482949 2147482932 2147481426 2147481393	
R7 [a3] = 0		[7fffff390] 2147481376 2147481290 2147481243 2147481227	
R8 [t0] = 0		[7fffff3a0] 2147481203 2147481169 2147481146 2147481126	
R9 [t1] = 0		[7fffff3b0] 2147481094 2147481078 2147481048 2147481037	
R10 [t2] = 0		[7fffff3c0] 2147481029 2147481011 2147480980 2147480957	
R11 [t3] = 0		[7fffff3d0] 2147480872 2147480768 2147480750 2147480696	(. x /
R12 [t4] = 0		[7fffff3e0] 2147480638 2147480620 0000000000 0788529152	> . . . , /
R13 [t5] = 0		[7fffff3f0] 1701670760 1986623023 1969974117 0796025197	h o m e / v i v e k k u m a r /
R14 [t6] = 0		[7fffff400] 1651796322 1701536629 0808595316 1717514546	b i t b u c k e t / 2 0 2 1 _ f
R15 [t7] = 0		[7fffff410] 1600941153 1701864293 0791689009 0828531052	a l l _ e c p e 1 7 0 / 1 a b 1
R16 [s0] = 0		[7fffff420] 1634742064 0775058546 0007172961 1966030175	o / p a r t 2 . a s m . _ = / u
R17 [s1] = 0		[7fffff430] 1647276659 1898933865 1768977268 1280245869	s r / b i n / q t s p i m . O L
R18 [s2] = 0		[7fffff440] 1146572868 1869098813 1982817645 1801811561	D P W D = / h o m e / v i v e k
R19 [s3] = 0		[7fffff450] 1634563435 1768042354 1668637300 0796157291	k u m a r / b i t b u c k e t /
R20 [s4] = 0		[7fffff460] 0825372722 1818322527 1667587948 0925984112	2 0 2 1 _ f a l l _ e c p e 1 7
R21 [s5] = 0		[7fffff470] 1634749790 0003158370 1398096452 1397052255	0 / l a b 1 0 . D B U S _ S E S
R22 [s6] = 0		[7fffff480] 1313818963 1398096479 1145323871 1397966162	S I O N _ B U S _ A D D R E S S
R23 [s7] = 0		[7fffff490] 1768846653 1634744952 0792553588 0795768178	= u n i x : p a t h = / r u n /
R24 [sp] = 2147480344		[7fffff4a0] 1919251317 0808464687 1969368880 1145503859	u s e r / 1 0 0 0 / b u s . G D
R25 [s8] = 0		[7fffff4b0] 1397052237 1313818963 1969386813 0007697518	M S E S S I O N = u b u n t u .
R26 [k0] = 0		[7fffff4c0] 1213481296 1937059645 1869361010 0795631971	P A T H = / u s r / l o c a l /
R27 [k1] = 0		[7fffff4d0] 1852400243 1937059642 1869361010 0795631971	s b i n : / u s r / l o c a l /
R28 [gp] = 268468224		[7fffff4e0] 0980314466 1920169263 1768059695 1966029422	b i n : / u s r / s b i n : / u
R29 [sp] = 2147480344		[7fffff4f0] 1647276659 0792358505 1852400243 1768042298	s r / b i n : / s b i n : / b i
R30 [s8] = 0		[7fffff500] 10260026077 1731162730 10260026077 1027058642	n : / u s r / a s m o o n : / u

FP Regs	nt Regs [10]	Data	Text
Int Regs [10]		Data	
R0 [r0] = 0		User data segment [10000000]..[10040000]	
R1 [at] = 268500992		[10000000]..[100fffff] 00000000	
R2 [v0] = 10		[10010000] 0000000010 0000000015 0000000006 0000000000	
R3 [v1] = 0		[10010010]..[1003ffff] 00000000	
R4 [a0] = 1			
R5 [a1] = 2147480348			
R6 [a2] = 2147480356			
R7 [a3] = 0		User Stack [7fffff318]..[80000000]	
R8 [t0] = 5		[7fffff320] 0000000001 2147483632 2147483550 2147483531	
R9 [t1] = 0		[7fffff330] 2147483511 2147483466 2147483443 2147483399	w . . . J . . . 3
R10 [t2] = 0		[7fffff340] 2147483367 2147483326 2147483306 2147483283 d . . . O . . . < . . .
R11 [t3] = 1		[7fffff350] 2147483264 2147483236 2147483215 2147483196	! k . . . W
R12 [t4] = 2		[7fffff360] 2147483169 2147483148 2147483096 2147483055	E . . . 4 . . . R . . . 1
R13 [t5] = 0		[7fffff370] 2147483022 2147483009 2147482987 2147482967	
R14 [t6] = 0		[7fffff380] 2147482949 2147482932 2147481426 2147481393	
R15 [t7] = 0		[7fffff390] 2147481376 2147481290 2147481243 2147481227	
R16 [s0] = 10		[7fffff3a0] 2147481203 2147481169 2147481146 2147481126	s . . . Q . . . : . . . & . . .
R17 [s1] = 15		[7fffff3b0] 2147481094 2147481078 2147481048 2147481037	
R18 [s2] = 6		[7fffff3c0] 2147480980 2147480957 0007172961 1966030175	
R19 [s3] = 3		[7fffff3d0] 2147480872 2147480768 2147480750 2147480696	(. x /
R20 [s4] = 0		[7fffff3e0] 1768846653 2147480620 0000000000 0788529152	> . . . , /
R21 [s5] = 0		[7fffff3f0] 1919251317 0808464687 1969368880 1145503859	h o m e / v i v e k k u m a r /
R22 [s6] = 0		[7fffff400] 1600941153 1701864293 0791689009 0828531052	b i t b u c k e t / 2 0 2 1 _ f
R23 [s7] = 0		[7fffff410] 1634742064 0775058546 0007172961 1966030175	a l l _ e c p e 1 7 0 / 1 a b 1
R24 [t8] = 0		[7fffff420] 1647276659 1898933865 1768977268 1280245869	o / p a r t 2 . a s m . _ = / u
R25 [t9] = 0		[7fffff430] 1146572868 1869098813 1982817645 1801811561	s r / b i n / q t s p i m . O L
R26 [k0] = 0		[7fffff440] 1634563435 1768042354 1668637300 0796157291	D P W D = / h o m e / v i v e k
R27 [k1] = 0		[7fffff450] 0825372722 1818322527 1667587948 0925984112	k u m a r / b i t b u c k e t /
R28 [gp] = 268468224		[7fffff460] 1344797920 0003158370 1398096452 1397052255	2 0 2 1 _ f a l l _ e c p e 1 7
R29 [sp] = 2147480344		[7fffff470] 1313818963 1398096479 1145323871 1397966162	0 / l a b 1 0 . D B U S _ S E S
R30 [s8] = 0		[7fffff480] 1852400243 1937059642 1869361010 0795631971	S I O N _ B U S _ A D D R E S S
R31 [ra] = 4194328		[7fffff490] 0980314466 1920169263 1768059695 1966029422	= u n i x : p a t h = / r u n /
		[7fffff4a0] 1919251317 0808464687 1969368880 1145503859	u s e r / 1 0 0 0 / b u s . G D
		[7fffff4b0] 1397052237 1313818963 1969386813 0007697518	M S E S S I O N = u b u n t u .
		[7fffff4c0] 1213481296 1937059645 1869361010 0795631971	P A T H = / u s r / l o c a l /
		[7fffff4d0] 1852400243 1937059642 1869361010 0795631971	s b i n : / u s r / l o c a l /
		[7fffff4e0] 0980314466 1920169263 1768059695 1966029422	b i n : / u s r / s b i n : / u
		[7fffff4f0] 1647276659 0792358505 1852400243 1768042298	s r / b i n : / s b i n : / b i
		[7fffff500] 10260026077 1731162730 10260026077 1027058642	n : / u s r / a s m o o n : / u

Question 5: Take a screenshot of the MIPS register panel after your program finishes. Put the register panel in Decimal mode (right-click) so it is easy to see register values.

Answer:

FP Regs	Int Regs [10]	Data	Text
Int Regs [10]			
R0 [r0] = 0		[00400008] 24a60004 addiu \$6, \$5, 4 ; 185: addiu \$a2 \$a1 4 # envp	
R1 [at] = 268500992		[0040000c] 00041080 sll \$2, \$4, 2 ; 186: sll \$v0 \$a0 2	
R2 [v0] = 10		[00400010] 00c23021 addi \$6, \$6, \$2 ; 187: addu \$a2 \$a2 \$v0	
R3 [v1] = 0		[00400014] 0c100009 jal 0x00400024 [main] ; 188: jal main	
R4 [a0] = 1		[00400018] 00000000 nop ; 189: nop	
R5 [a1] = 2147480408		[0040001c] 3402000a ori \$2, \$0, 10 ; 191: li \$v0 10	
R6 [a2] = 2147480416		[00400024] 00000000 syscall ; 192: syscall # syscall 10 (exit)	
R7 [a3] = 0		[00400028] 8c300000 lui \$1, 4097 [Z] ; 12: lw \$s0, Z # Z = 2	
R8 [t0] = 20		[0040002c] 3c011001 lui \$1, 4097 [i] ; 13: lw \$s1, i # i = 0	
R9 [t1] = 100		[00400030] 8c310004 lw \$17, 4(\$1) [i] ; 14: addi \$t0,\$zero,20 # storing 20 in temporary register	
R10 [t2] = 0		[00400034] 20080014 addi \$8, \$0, 20 ; 17: bgt \$s1, \$t0, endwhile # if i > 20	
R11 [t3] = 0		[00400038] 0111082a slt \$1, \$8, \$17 ; 18: addi \$s0, \$s0, 1 # Z++	
R12 [t4] = 0		[0040003c] 14200000 bne \$1, \$0, 16 [endwhile-0x0040003c] ; 19: addi \$s1, \$s1, 2 # i += 2	
R13 [t5] = 0		[00400040] 22100001 addi \$16, \$16, 1 ; 20: j while	
R14 [t6] = 0		[00400044] 22310002 addi \$17, \$17, 2 ; 21: addi \$s0, \$s0, 100 # storing 100 in temporary register.	
R15 [t7] = 0		[00400048] 08100006 j 0x00400038 [while] ; 22: addi \$t1,\$zero, 100 # storing 100 in temporary register.	
R16 [s0] = 78		[0040004c] 20090064 addi \$9, \$0, 100 ; 23: addi \$t1,\$zero, 100 # storing 100 in temporary register.	
R17 [s1] = 0		[00400050] 0209002a slt \$1, \$16, \$9 ; 24: addi \$t1,\$zero, 100 # storing 100 in temporary register.	
R18 [s2] = 0		[00400054] 10200003 beq \$1, \$0, 12 [endifofwhile-0x00400054] ; 25: bge \$s0, \$t1, endofwhile # checking for exit condition	
R19 [s3] = 0		[00400058] 22100001 addi \$16, \$16, 1 ; 26: addi \$s0, \$s0, 1 # Z++	
R20 [s4] = 0		[0040005c] 08100014 beq \$1, \$0, 12 [endifofwhile-0x00400054] ; 27: j dowhile	
R21 [s5] = 0		[00400060] 0011082a slt \$1, \$0, \$17 ; 28: bne \$s1, \$zero, end # exit statement. i	
R22 [s6] = 0		[00400064] 10200004 beg \$1, \$0, 16 [end-0x00400064] ; 29: addi \$s0, \$s0, 1 # Z--	
R23 [s7] = 0		[00400068] 2210ffff addi \$16, \$16, -1 ; 30: sub \$s0, \$s0, 1 # Z--	
R24 [t8] = 0		[0040006c] 2231ffff addi \$17, \$17, -1 ; 31: sub \$s1, \$s1, 1 # i--	
R25 [t9] = 0		[00400070] 08100018 j 0x00400060 [while2] ; 32: j while2	
R26 [k0] = 0		[00400074] 3c011001 lui \$1, 4097 [Z] ; 33: sw \$s0, Z	
R27 [k1] = 0		[00400078] ac300000 sw \$16, 0(\$1) [Z] ; 34: j while2	
R28 [gp] = 268468224		[0040007c] 3c011001 lui \$1, 4097 [i] ; 35: sw \$s1, Z	
R29 [sp] = 2147480404		[00400080] ac310004 su \$17, 4(\$1) [i] ; 36: sw \$s1, i	
R30 [s8] = 0		[00400084] 3402000a ori \$2, \$0, 10 ; 37: sw \$s1, i	
R31 [ra] = 4194328		[00400088] 0000000c syscall ; 38: li \$v0, 10 # Sets \$v0 to "10" to select exit syscall	
		[0040008c] ; 39: syscall # Exit	
Kernel Text Segment [80000000]..[80010000]			

Question 6: Take a screenshot of the MIPS memory panel (data tab) after your program finishes. Put the memory panel in Decimal mode (right-click), so it is easy to see memory values. Circle the memory location (not register) that contains the final calculated values of I and Z.

Answer:

FP Regs	Int Regs [10]	Data	Text
Int Regs [10]			
R0 [r0] = 0		User data segment [10000000]..[10040000]	
R1 [at] = 268500992		[10000000]..[100ffff] 00000000	
R2 [v0] = 10		[10010000]..[1000000078] 0000000000 0000000000 0000000000 N	
R3 [v1] = 0		[10010010]..[1003ffff] 00000000	
R4 [a0] = 1		User Stack [7ffff354]..[80000000]	
R5 [a1] = 2147480408		[7ffff354] 0000000001 2147480617 0000000000	
R6 [a2] = 2147480416		[7ffff360] 2147483632 2147483550 2147483531 2147483511	
R7 [a3] = 0		[7ffff370] 2147483466 2147483443 2147483399 2147483367	
R8 [t0] = 20		[7ffff380] 2147483326 2147483306 2147483283 2147483264	
R9 [t1] = 100		[7ffff390] 2147483236 2147483215 2147483196 2147483169	
R10 [t2] = 0		[7ffff3a0] 2147483148 2147483096 2147483055 2147483022	
R11 [t3] = 0		[7ffff3b0] 2147483009 2147482987 2147482967 2147482949	
R12 [t4] = 0		[7ffff3c0] 2147482932 2147481426 2147481393 2147481376	
R13 [t5] = 0		[7ffff3d0] 2147481290 2147481243 2147481227 2147481203	
R14 [t6] = 0		[7ffff3e0] 2147481169 2147481146 2147481126 2147481094	
R15 [t7] = 0		[7ffff3f0] 2147481078 2147481048 2147481037 2147481029	
R16 [s0] = 78		[7ffff400] 2147481011 2147480980 2147480957 2147480872	
R17 [s1] = 0		[7ffff410] 2147480768 2147480750 2147480696 2147480678	
R18 [s2] = 0		[7ffff420] 0000000000 0000000000 1869098752 1982817645	
R19 [s3] = 0		[7ffff430] 1801811561 1634563435 1768042354 1668637300	i v e k k u m a r / b i t b u c
R20 [s4] = 0		[7ffff440] 0796157291 0825372722 1818322527 1667587948	k e t / 2 0 2 1 _ f a i l _ e c
R21 [s5] = 0		[7ffff450] 0925984112 163479920 0791687522 1953653104	p e 1 7 0 / 1 a b 1 0 / p a r t
R22 [s6] = 0		[7ffff460] 1935748659 1029636205 1920169263 1852400175	3 . a s m . _ = / u s r / b i n
R23 [s7] = 0		[7ffff470] 1937010991 0007170416 1398096452 1397052255	/ q t s p i m . D B U S _ S E S S
R24 [t8] = 0		[7ffff480] 1313818963 1398096479 1145323871 1397966162	S I O N _ B U S _ A D D R E S S
R25 [t9] = 0		[7ffff490] 1768846653 1634744952 0792553588 0795768178	= u n i x : p a t h = / r u n /
R26 [k0] = 0		[7ffff4a0] 1919251317 0808464687 169368880 1145503859	u s e r / 1 0 0 0 / b u s . G D
R27 [k1] = 0		[7ffff4b0] 1397052237 1313818963 1969386813 007697518	M S E S S I O N = u b u n t u .
R28 [gp] = 268468224		[7ffff4c0] 1213481296 1937059645 1869361010 0795631971	P A T H = / u s r / l o c a l /
R29 [sp] = 2147480404		[7ffff4d0] 1852400243 1937059642 1869361010 0795631971	s b i n : / u s r / l o c a l /
R30 [s8] = 0		[7ffff4e0] 0980314466 1920169263 1768059695 1966029422	b i n : / u s r / s b i n : / u
R31 [ra] = 4194328		[7ffff4f0] 1647276659 0792358505 1852400243 1768042292	s r / b i n : / s b i n : / b i
		[7ffff500] 1966029422 1731162739 1936026977 1937059642	n : / u s r / g a m e s : / u s
		[7ffff510] 1869361010 0795631971 1701667175 1932474995	r / l o c a l / g a m e s : / s
		[7ffff520] 0795894126 0007235938 1958050672 1096040772	n a p / b i n . X D G _ D A T A
		[7ffff530] 1380533343 1966030163 1932489331 1701994856	- D I R S = / u s r / s h a r e
		[7ffff540] 1060386700 0008776046 1000160622 1668245757	/ .. b .. n .. t .. / u .. c .. / l .. o ..

Question 7: Take a screenshot of the MIPS register panel after your program finishes. Put the register panel in Decimal mode (right-click) so it is easy to see register values.

Answer:

FP Regs	Int Regs [10]	Data	Text
Int Regs [10]		Text	
R0 [r0] = 0	[0040001c] 3402000a ori \$2, \$0, 10 ; 191: li \$v0 10		
R1 [at] = 1	[00400020] 0000000c syscall ; 192: syscall # syscall 10 (exit)		
R2 [v0] = 10	[00400024] 3c011001 lui \$1, 4097 [A] ; 12: la \$s0, A # s0 = &A[0]		
R3 [v1] = 0	[00400028] 34300000 ori \$16, \$1, 0 [A]		
R4 [a0] = 1	[0040002c] 3c011001 lui \$1, 4097 [B] ; 13: la \$s1, B # s1 = &B[0]		
R5 [a1] = 2147480348	[00400030] 34310014 ori \$17, \$1, 20 [B]		
R6 [a2] = 2147480356	[00400034] 20120000 addi \$18, \$0, 0 ; 14: addi \$s2, \$zero, 0 # i = 0		
R7 [a3] = 0	[00400038] 20080000 addi \$8, \$0, 5 ; 15: addi \$t0, \$zero, 5 # for the loop		
R8 [t0] = 5	[0040003c] 02480082a slt \$1, \$18, \$8 ; 17: bge \$s2, \$t0, endfor		
R9 [t1] = 16	[00400040] 1020000a beq \$1, \$0, 40 [endfor-0x00400040]		
R10 [t2] = 5	[00400044] 02524820 add \$9, \$18, \$18 ; 18: add \$t1, \$s2, \$s2 # 2 * i		
R11 [t3] = 17	[00400048] 01294820 add \$9, \$9, \$9 ; 19: add \$t1, \$t1, \$t1 # 4 * i or 4 bytes * i		
R12 [t4] = 0	[0040004c] 02099820 add \$19, \$16, \$9 ; 20: add \$s3, \$s0, \$t1 # (A + i)		
R13 [t5] = 13	[00400050] 0229a020 add \$20, \$17, \$9 ; 21: add \$t2, \$s1, \$t1 # (B + i)		
R14 [t6] = 0	[00400054] 8e8a0000 lw \$10, 0(\$20) ; 22: lw \$t2, 0(\$s4) # *(B + i)		
R15 [t7] = 0	[00400058] 214b000c addi \$11, \$10, 12 ; 23: addi \$t3, \$t2, 12 # temp = B[i] + C		
R16 [s0] = 268500992	[0040005c] ae6b0000 ss \$11, 0(\$19) ; 24: sw \$t3, 0(\$s3) # A[i1] = temp		
R17 [s1] = 268501012	[00400060] 22520001 addi \$18, \$18, 1 ; 25: addi \$s2, \$s2, 1 # i++		
R18 [s2] = -1	[00400064] 0810000f j 0x0040003c [for] ; 26: j for		
R19 [s3] = 268501008	[00400068] 2252ffff addi \$18, \$18, -1 ; 29: sub \$s2, \$s2, 1 # i--		
R20 [s4] = 268501028	[0040006c] 0240082a slt \$1, \$18, \$0 ; 31: bit \$s2, \$zero, endwhile		
R21 [s5] = 268500992	[00400070] 14200009 bne \$1, \$0, 36 [endwhile-0x00400070]		
R22 [s6] = 26	[00400074] 02526020 add \$12, \$18, \$18 ; 32: add \$t4, \$s2, \$s2 # 2 * i		
R23 [s7] = 0	[00400078] 018c6020 add \$12, \$12, \$12 ; 33: add \$t4, \$t4, \$t4 # 4 * i or 4 bytes * i		
R24 [t8] = 0	[0040007c] 020ca820 add \$21, \$16, \$12 ; 34: add \$s5, \$s0, \$t4 # (A + i)		
R25 [t9] = 0	[00400080] 8ead0000 lw \$13, 0(\$21) ; 35: lw \$t5, 0(\$s5) # *(A + i)		
R26 [k0] = 0	[00400084] 01abd020 add \$22, \$13, \$13 ; 36: add \$s6, \$t5, \$t5 # A[i] * 2		
R27 [k1] = 0	[00400088] ae6b0000 ss \$22, 0(\$21) ; 37: sw \$s6, 0(\$s5) # A[i] = A[i] * 2		
R28 [gp] = 268468224	[0040008c] 2252ffff addi \$18, \$18, -1 ; 38: sub \$s2, \$s2, 1 # i--		
R29 [sp] = 2147480344	[00400090] 0810001b j 0x0040006c [while] ; 39: j while		
R30 [s8] = 0	[00400094] 3402000a ori \$2, \$0, 10 ; 50: li \$v0, 10 # Sets \$v0 to "10" to select exit syscall		
R31 [ra] = 4194328	[00400098] 0000000c syscall ; 51: syscall # Exit		

Question 8: Take a screenshot of the MIPS memory panel (data tab) after your program finishes. Put the memory panel in Decimal mode (right-click), so it is easy to see memory values. Circle the final values of array A.

Answer:

FP Regs	Int Regs [10]	Data	Text
Int Regs [10]		Data	
R0 [r0] = 0	[10000000..1000ffff] 00000000 User data segment [10000000..10040000]		
R1 [at] = 1	[10010000] 000000026 000000028 000000030 000000032		
R2 [v0] = 10	[10010010] 000000034 000000001 000000002 000000003 "		
R3 [v1] = 0	[10010020] 000000004 000000005 000000000 000000000		
R4 [a0] = 1	[10010030]..[1003ffff] 00000000		
R5 [a1] = 2147480348			
R6 [a2] = 2147480356			
R7 [a3] = 0			
R8 [t0] = 5			
R9 [t1] = 16			
R10 [t2] = 5			
R11 [t3] = 17			
R12 [t4] = 0			
R13 [t5] = 13			
R14 [t6] = 0			
R15 [t7] = 0			
R16 [s0] = 268500992			
R17 [s1] = 268501012			
R18 [s2] = -1			
R19 [s3] = 268501008			
R20 [s4] = 268501028			
R21 [s5] = 268500992			
R22 [s6] = 26			
R23 [s7] = 0			
R24 [t8] = 0			
R25 [t9] = 0			
R26 [k0] = 0			
R27 [k1] = 0			
R28 [gp] = 268468224			
R29 [sp] = 2147480344			
R30 [s8] = 0			
R31 [ra] = 4194328			

User data segment [10000000..10040000]

```

[10000000..1000ffff] 00000000
[10010000] 000000026 000000028 000000030 000000032 . . . . .
[10010010] 000000034 000000001 000000002 000000003 " . . . . .
[10010020] 000000004 000000005 000000000 000000000 . . . . .
[10010030]..[1003ffff] 00000000 . . . . .

```

User Stack [7fffff318]..[80000000]

```

[7fffff318] 000000001 2147480559
[7fffff320] 000000000 2147483632 2147483550 2147483531 . . . . .
[7fffff330] 2147483511 2147483466 2147483443 2147483399 w . . . J . . . 3 . . . .
[7fffff340] 2147483367 2147483326 2147483306 2147483283 . . . . .
[7fffff350] 2147483264 2147483236 2147483215 2147483196 . . . . d . . . O . . . < . . .
[7fffff360] 2147483169 2147483148 2147483096 2147483055 ! . . . . .
[7fffff370] 2147483022 2147483009 2147482987 2147482967 . . . . . k . . . W . . .
[7fffff380] 2147482949 2147482932 2147481426 2147481393 E . . . 4 . . . R . . . 1 . . .
[7fffff390] 2147481376 2147481290 2147481243 2147481227 . . . . .
[7fffff3a0] 2147481203 2147481169 2147481146 2147481126 s . . . Q . . . . & . . .
[7fffff3b0] 2147481094 2147481078 2147481048 2147481037 . . . . .
[7fffff3c0] 2147481029 2147481011 2147480980 2147480957 . . . . .
[7fffff3d0] 2147480872 2147480768 2147480750 2147480696 ( . . . . . x . . .
[7fffff3e0] 2147480638 2147480620 000000000 0788529152 > . . . , . . . . . / h o m e / v i v e k k u m a r / b i t b u c k e t / 2 0 2 1 _ f
[7fffff3f0] 1701670760 1986623023 1969974117 0796025197 a l l _ e c p e 1 7 0 / l a b 1 0 / p a r t 4 . a s m . _ = / u s r / b i n / q t s p i m . o L D P W D = / h o m e / v i v e k k u m a r / b i t b u c k e t / 2 0 2 1 _ f a l l _ e c p e 1 7 0 / l a b 1 0 . D B U S _ S E S S I O N _ B U S _ A D D R E S S = u n i x : p a t h = / r u n / u s e r / 1 0 0 0 / b u s . G D M S E S S I O N = u b u n t u .

```

Question 9:

FP Regs	Int Regs [16]	Data	Text
	Int Regs [16]		
R0 [r0] = 0			
R1 [at] = 10010000			
R2 [v0] = a			
R3 [v1] = 0			
R4 [a0] = 10010104			
R5 [a1] = 100			
R6 [a2] = 7ffff360			
R7 [a3] = 0			
R8 [t0] = 3			
R9 [t1] = 10010003			
R10 [t2] = 65			
R11 [t3] = 0			
R12 [t4] = 0			
R13 [t5] = 0			
R14 [t6] = 0			
R15 [t7] = 0			
R16 [s0] = 10010000			
R17 [s1] = 10010003			
R18 [s2] = 65			
R19 [s3] = 0			
R20 [s4] = 0			
R21 [s5] = 0			
R22 [s6] = 0			
R23 [s7] = 0			
R24 [t8] = 0			
R25 [t9] = 0			
R26 [k0] = 0			
R27 [k1] = 0			
R28 [gp] = 10008000			
R29 [sp] = 7ffff354			
R30 [s8] = 0			
R31 [ra] = 400018			
		User data segment [10000000]..[10040000]	
		[10000000]..[1000ffff] 00000000	s o m e b o d y
		[10010000] 656d6f73 79646f62 0000000a 00000000	
		[10010010]..[100100ff] 00000000	
		[10010100] 10010003 6946000a 20747372 6374616d F i r s t m a t c
		[10010110] 74612068 64646120 73736572 68540020	h a t a d d r e s s . T h
		[10010120] 616d2065 69686374 6320676e 61726168	e m a t c h i n g c h a r a
		[10010130] 72657463 20736920 206f4e00 6374616d	c t e r i s . N o m a t c
		[10010140] 6f662068 0a646e75 00000000 00000000	h f o u n d
		[10010150]..[1003ffff] 00000000	
		User Stack [7ffff354]..[80000000]	
		[7ffff354] 00000001 7ffff429 00000000)
		[7ffff360] 7fffffff0 7fffff9e 7fffff8b 7fffff77 w . . .
		[7ffff370] 7fffff4a 7fffff33 7fffff07 7ffffee7	J . . . 3
		[7ffff380] 7fffffebe 7fffffea 7fffffe93 7ffffe80
		[7ffff390] 7fffffe64 7fffffe4f 7fffffe3c 7fffffe21	d . . . 0 . . . < . . . ! . .
		[7ffff3a0] 7fffffe0c 7fffffd8 7fffffdaf 7fffffd8e
		[7ffff3b0] 7fffffd81 7fffffd6b 7ffffd57 7ffffd45 k . . . w . . . E . . .
		[7ffff3c0] 7ffffd34 7fffff752 7fffff731 7fffff720	4 . . . R . . . 1
		[7ffff3d0] 7fffff6ca 7fffff69b 7fffff68b 7fffff673 s . . .
		[7ffff3e0] 7fffff651 7fffff63a 7fffff626 7fffff606	Q . . . : . . . &
		[7ffff3f0] 7fffff5f6 7fffff5d8 7fffff5cd 7fffff5c5
		[7ffff400] 7fffff5b3 7fffff594 7fffff57d 7fffff528) . . . (. . .
		[7ffff410] 7fffff4c0 7fffff4ae 7fffff478 7fffff466 x . . . f . . .
		[7ffff420] 00000000 00000000 6f682f00 762f656d / h o m e / v
		[7ffff430] 6b657669 616d756b 69622f72 63756274	i v e k k u m a r / b i t b u c
		[7ffff440] 2f74656b 31323032 6c61665f 63655f6c	k e t / 2 0 2 1 _ f a l l _ e c
		[7ffff450] 37316570 616c2f30 2f303162 74726170	p e 1 7 0 / l a b 1 0 / p a r t
		[7ffff460] 73612e35 3d5f006d 7273752f 6e69622f	5 . a s m . _ = / u s r / b i n
		[7ffff470] 7374712f 006d6970 53554244 5345535f	/ q t s p i m . D B U S _ S E S
		[7ffff480] 4e4f4953 5355425f 4444415f 53534552	S I O N _ B U S _ A D D R E S S
		[7ffff490] 696e753d 61703a78 2f3d6874 2f6e7572	= u n i x : p a t h = / r u n /
		[7ffff4a0] 72657375 3030312f 75622f30 44470073	u s e r / 1 0 0 0 / b u s . G D
		[7ffff4b0] 5345534d 4e4f4953 7562753d 0075746e	M S E S S I O N = u b u n t u .
		[7ffff4c0] 10008000 7775074 2f303162 74726170	n a t u r e / u n x / 1 - - - /