e



**Problem 1:**

**Computer Science or Information Technology**

Instructor: Dr. G.E. Antoniou

Day, Month, Year

Day

CSIT 502

Department of CSIT

Assessment

Module-6

Hidalgo, Rafael

1. Input and Output in the command line, MARS MIPS simulator, window

(a) Read three–integers from the command line window

(b) ADD the three–integers

(c) Print–out the result in the command line window

• At the end of the problem clearly state the result (decimal)

**Solution (code)**

#read three ints, adds them, then outputs answer in console.

.text

.globl main

main:

# prints "Please input first integer: "

la $a0, prompt1

li $v0, 4

syscall

# inputs the first integer to $t0

li $v0, 5

syscall

move $t0, $v0

# prints "Please input second integer: "

la $a0, prompt2

li $v0, 4

syscall

# inputs the second integer to $t1

li $v0, 5

syscall

move $t1, $v0

# prints "Please input third integer: "

la $a0, prompt3

li $v0, 4

syscall

# inputs the third integer to $t2

li $v0, 5

syscall

move $t2, $v0

# prints "The sum of your three integers is: "

la $a0, prompt4

li $v0, 4

syscall

# Adds first and second integer to $a0

add $a0, $t0, $t1

# Adds first and second and third integer to $a0

add $a0, $a0, $t2

# Prints the result of three integers added together

li $v0, 1

syscall

# exits the program

li $v0, 10

syscall

.data

prompt1:

.asciiz "Please input first integer. "

prompt2:

.asciiz "Please input second integer. "

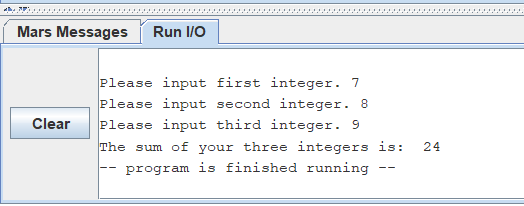
prompt3:

.asciiz "Please input third integer. "

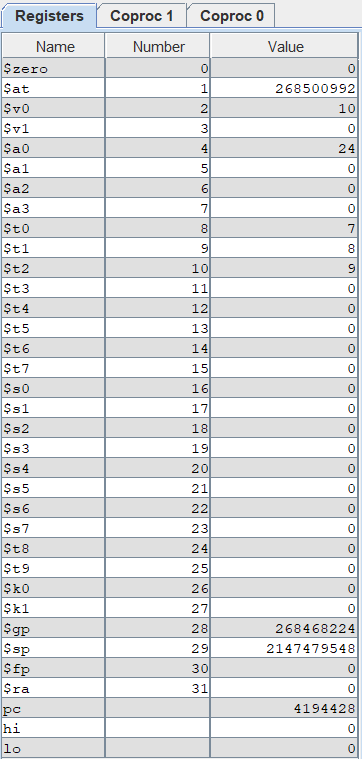
prompt4:

.asciiz "The sum of your three integers is: "

Sample Run (Console):



Sample Run (Registers area), only in decimal:



Result:

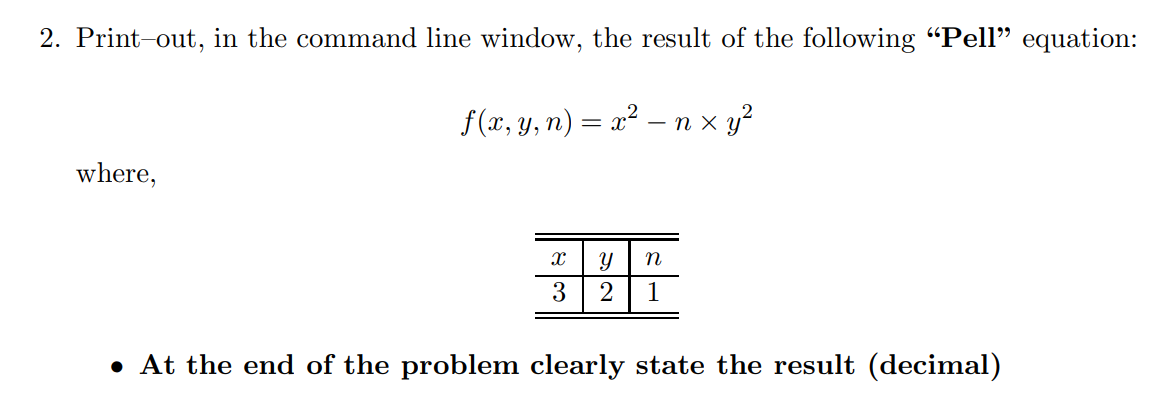
7 + 8 + 9 = 24

As you can see above the proper result is displayed in the command line window.

Brief Comments:

The program runs correctly, according to the specifications

**Problem 2:**



**Solution (code)**

# prints out the result of the pell equation (x^2)-n\*(y^2) where x=3, y=2, and n =1

.text

.globl main

main:

# assigns 3 to $t0, which represents x in the pell equation.

li $t0,3

# assigns 2 to $t1, which represents y in the pell equation.

li $t1,2

# assigns 1 to $t2, which represents n in the pell equation.

li $t2,1

# calculates x^2 and puts it in $t3

mul $t3,$t0,$t0

# calculates y^2 and puts it in $t4

mul $t4,$t1,$t1

# calculates n\*(y^2) and puts it in $t5

mul $t5,$t2,$t4

# calculates (x^2)-n\*(y^2) and puts it in $t6

sub $t6,$t3,$t5

# prints "The result of the pell equation is: "

la $a0, prompt

li $v0, 4

syscall

# prints the output of the operation ($t6)

move $a0, $t6

li $v0, 1

syscall

# exits the program

li $v0, 10

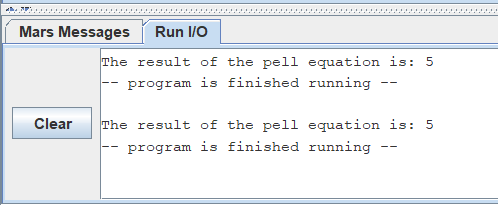
syscall

.data

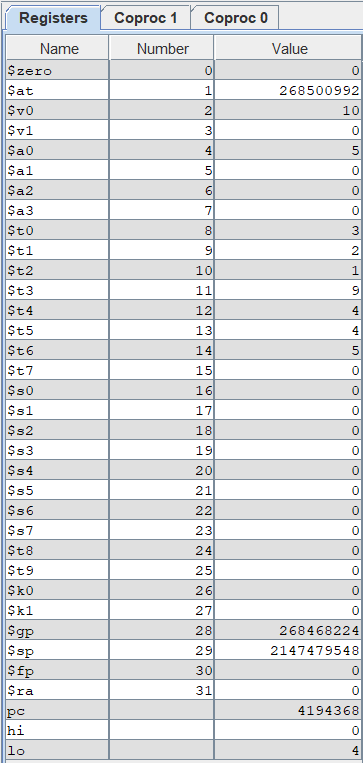
prompt:

.asciiz "The result of the pell equation is: "

Sample Run (Console):



Sample Run (Registers area), only in decimal:



Result:

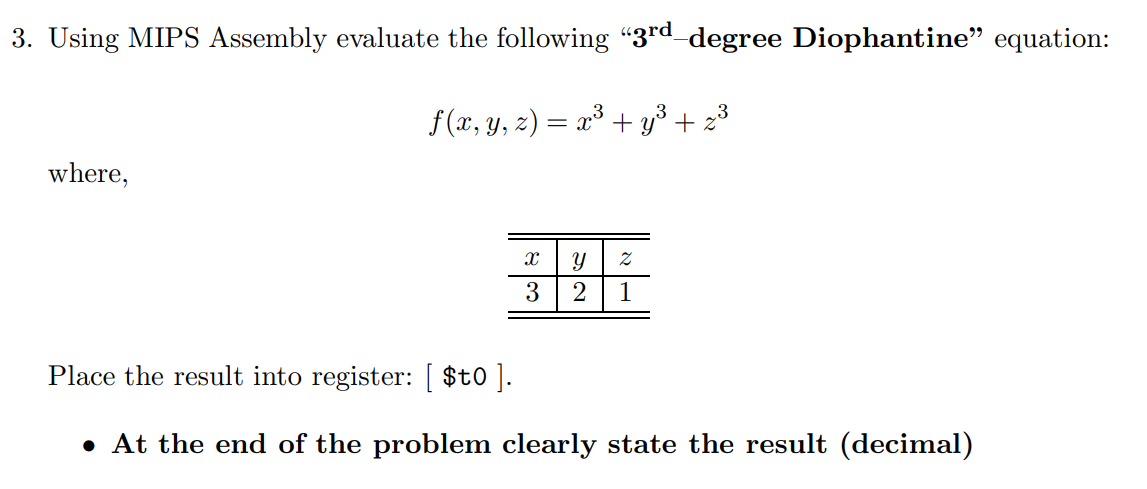
(x^2)-n\*(y^2) where x=3, y=2, and n =1

This equation yields 5, and as can be seen above, the appropriate result is displayed in the command window and in the registers area.

Brief Comments:

The program runs correctly, according to the specifications

**Problem 3:**



**Solution (code)**

# prints out the result of the 3rd–degree Diophantine equation x^3 + y^3 + z^3 where x = 3, y = 2 and z = 1

.text

.globl main

main:

# assigns 3 to $t0, which represents x in the 3rd–degree Diophantine equation.

li $t0,3

# assigns 2 to $t1, which represents y in the 3rd–degree Diophantine equation.

li $t1,2

# assigns 1 to $t2, which represents z in the 3rd–degree Diophantine equation.

li $t2,1

# calculates x^2 and puts it in $t3

mul $t3,$t0,$t0

# calculates x^3 and puts it in $t3

mul $t3,$t3,$t0

# calculates y^2 and puts it in $t4

mul $t4,$t1,$t1

# calculates y^3 and puts it in $t4

mul $t4,$t4,$t1

# calculates z^2 and puts it in $t5

mul $t5,$t2,$t2

# calculates z^3 and puts it in $t5

mul $t5,$t5,$t2

# adds x^3 + y^3 and puts it in $t6

add $t6,$t3,$t4

# adds x^3 + y^3 + z^3 and puts it in $t0

add $t0,$t6,$t5

# prints "The result of the 3rd degree Diophantine equation is: "

la $a0, prompt

li $v0, 4

syscall

# prints the output of the operation ($t0)

move $a0, $t0

li $v0, 1

syscall

# exits the program

li $v0, 10

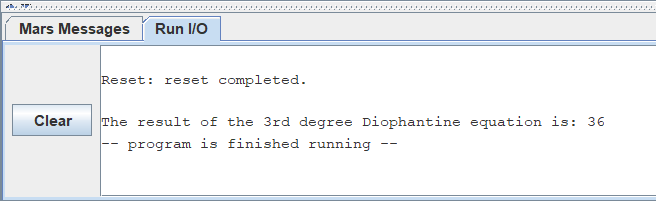
syscall

.data

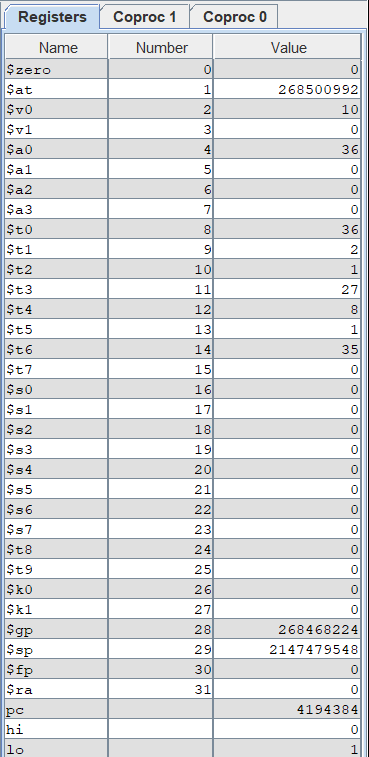
prompt:

.asciiz "The result of the 3rd degree Diophantine equation is: "

Sample Run (Console):



Sample Run (Registers area), only in decimal:



Result:

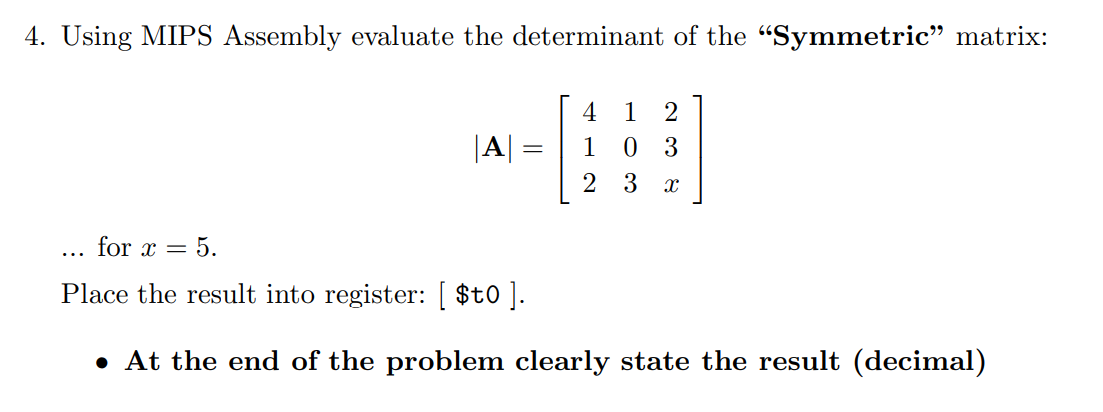
x^3 + y^3 + z^3 where x = 3, y = 2 and z = 1

This equation yields 36, and as can be seen above, the appropriate result is displayed in the command window and in the registers area. Also note the result was put in register $t0

Brief Comments:

The program runs correctly, according to the specifications

**Problem 4:**



**Solution (code)**

# prints out the result of the determinant utilizing the "determinant" equation aei + bfg + cdh - afh - bdi - ceg

# where a=4, b=1, c=2, d=1, e=0, f=3, g=2, h=3, and i=5

.text

.globl main

main:

# assigns 4 to $t1, which represents a in the "determinant" equation.

li $t1,4

# assigns 1 to $t2, which represents b in the "determinant" equation.

li $t2,1

# assigns 2 to $t3, which represents c in the "determinant" equation.

li $t3,2

# assigns 1 to $t4, which represents d in the "determinant" equation.

li $t4,1

# assigns 0 to $t5, which represents e in the "determinant" equation.

li $t5,0

# assigns 3 to $t6, which represents f in the "determinant" equation.

li $t6,3

# assigns 2 to $t7, which represents g in the "determinant" equation.

li $t7,2

# assigns 3 to $t8, which represents h in the "determinant" equation.

li $t8,3

# assigns 5 to $t9, which represents i in the "determinant" equation.

li $t9,5

# Calculates aei and inputs it into $s0

mul $s0,$t1,$t5

mul $s0,$s0,$t9

# Calculates bfg and inputs it into $s1

mul $s1,$t2,$t6

mul $s1,$s1,$t7

# Calculates cdh and inputs it into $s2

mul $s2,$t3,$t4

mul $s2,$s2,$t8

# Calculates afh and inputs it into $s3

mul $s3,$t1,$t6

mul $s3,$s3,$t8

# Calculates bdi and inputs it into $s4

mul $s4,$t2,$t4

mul $s4,$s4,$t9

# Calculates ceg and inputs it into $s5

mul $s5,$t3,$t5

mul $s5,$s5,$t7

# Performs aei + bfg and stores it in $t0

add $t0,$s0,$s1

# Performs aei + bfg + cdh and stores it in $t0

add $t0,$t0,$s2

# Performs aei + bfg + cdh - afh and stores it in $t0

sub $t0,$t0,$s3

# Performs aei + bfg + cdh - afh - bdi and stores it in $t0

sub $t0,$t0,$s4

# Performs aei + bfg + cdh - afh - bdi - ceg and stores it in $t0

sub $t0,$t0,$s5

# prints "The calculated determinant is: "

la $a0, prompt

li $v0, 4

syscall

# prints the output of the operation ($t0)

move $a0, $t0

li $v0, 1

syscall

# exits the program

li $v0, 10

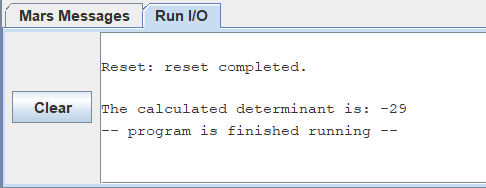
syscall

.data

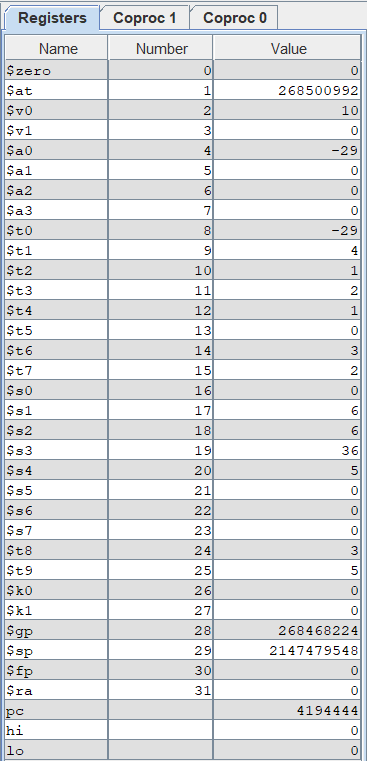
prompt:

.asciiz "The calculated determinant is: "

Sample Run (Console):



Sample Run (Registers area), only in decimal:



Result:

(aei + bfg + cdh - afh - bdi – ceg) where a=4, b=1, c=2, d=1, e=0, f=3, g=2, h=3, and i=5

This equation yields -29 and as can be seen above, the appropriate result is displayed in the command window and in the registers area. Also note the result was put in register $t0

Brief Comments:

The program runs correctly, according to the specifications