*CSCI-250* *Project 1*

# *Bin-O-Bucks*

**Requirements:** Write a MIPS assembly language program to help ”Robo Bank Teller” dispense cash using the smallest number of Bin-O-Bucks. Robots love binary numbers so much that their monetary denominations are based on powers of two. Bill denominations include $1024, $512, $256, $128, $64, $32, $16, $8, $4, $2 and $1.

The program will begin by printing a welcome message and prompting the customer to enter a dollar amount. You may assume that the customer will enter an integer value greater than or equal to 1 and less than 2048 (you do not need to check for that).

Use a variety of MIPS arithmetic instructions to compute the number of each type of bill denomination for changing the given amount of money into the fewest number of bills. Your solution must utilize the

1. bitwise logical shift left (SLL),
2. bitwise shift right (SRL or SRA),
3. multiplication (MUL),
4. dual-action division (DIV $numerator, $denominator) that yields the quotient and remainder, and
5. subtract (SUB) MIPS instructions.

Your solution for computing each individual denomination need only use one or two of these listed instructions. However, by the time the program has found all denominations, your program will have demonstrated use of all of these different types of arithmetic instructions. Find a place to use them all.

The printed output begins by reporting the entered dollar amount. Then display the number of bills for each denomination printed and a descriptive label identifying each denomination one per line. You may vary the wording of the messages if you wish to be more creative.

Sample Program Run

```
Greetings! I’m your robot bank teller.
How many power-of-two dollars do you want?
2000
Here are your $2000 Bin-O-Bucks:
1 1024 dollar bill
1 512 dollar bill
1 256 dollar bill
1 128 dollar bill
1 64 dollar bill
0 32 dollar bill
1 16 dollar bill
0 8 dollar bill
0 4 dollar bill
0 2 dollar bill
```

```
0 1 dollar bill
```

**Comments:** Include comments explaining what's going on in your code. Comments should indicate how each register is being used. Include your pledged statement in a header comment at the top of each program file.

**Deliverables:** Upload the source file binOBucks.asm to Blackboard.