

For this project, you will write an MIPS assembly language program to display a graphic of a given size. Your program should query the user for the size of the graphic and the character used to build the graphic. It should then produce an ASCII image with straight lines for a border and use the provided character to display a diamond inside the box. The size of the box should be the size that was provided by the user. Given the shape of the diamond, you do not have to handle the case where the size is an even number and you should request a new size if that occurs. Likewise, a negative number is an invalid size. Your program should generate graphics until the user enters 0 for the size.

Your program should include appropriate comments indicating what the code should be doing and what registers are being used for. After displaying the results, your program should exit cleanly. Please include your name and CLID in the program headers and include your CLID in the file names. Your programs should be turned in through Moodle before class starts on the due date. You should test your programs using the SPIM simulator to ensure their functionality before submitting them.

Example output:

```
Enter the size of the graphic or 0 to quit: 11
Enter a character to print: *
```

```
-----
|      *      |
|     ***     |
|    *****  |
|   *         |
|  *          |
| *           |
|*            |
| *           |
|  *          |
|   *         |
|    *****  |
|   *         |
|  *          |
| *           |
|*            |
| *           |
|  *          |
|   *         |
|    *****  |
|     ***     |
|      *      |
|-----|
```

```
Enter the size of the graphic or 0 to quit: 8
Only odd numbers are allowed.
```

```
Enter the size of the graphic or 0 to quit: 7
Enter a character to print: 0
```

```
-----
|  0  |
| 000 |
|00000|
| 000 |
|  0  |
|-----|
```

```
Enter the size of the graphic or 0 to quit: -2
The size must be a positive number.
```

| Enter the size of the graphic or 0 to quit: 0 |

Objectives:

1. To introduce and practice building control structures in the MIPS assembly language.
2. To introduce and practice building loops in the MIPS assembly language.