

# CS 271 and 462

## PA 2 - Programming Assignment 2

Starting with this assignment, if you submit a program with a syntax error, you will receive a grade of zero for that program.

When you finish this assignment, you will submit two programs:

- 1) cargo.c
- 2) pattern.c

### Overview of This Assignment

In this assignment, you will create 2 C programs using the Kate editor. You'll upload the programs to a CS Linux host, then compile and run them on Linux.

Once you have finished and tested the programs, you will upload them to the PA 1 assignment in Canvas.

### C Programming Skills That You Are Practicing in This Assignment

1. Using while loops
2. Using for loops
3. Creating a text file for input
4. Running a C program with redirected input

### Before You Begin

Read the lecture notes from week 2 and review the sample programs that were done in class.

Read the "Style and Documentation Requirements".

### New Linux Stuff

If you want the input to come from a file instead of the keyboard (applies to ALL input), you can run the program using what is called "input redirection".

- 1) Create the program. Use scanf as you normally would.
- 2) Compile as usual.
- 3) Run this way

```
./myprogram < inputfile
```

ALL input will be taken from the file named "inputfile".

## Reminder: Backup Your Work

### The First Program: `cargo.c`

This program will first input the total capacity weight of a cargo ship that is being loaded. Then the program will input the weights of [shipping containers](#) being loaded onto a ship. You must keep track of the total weight that has been loaded, the number of containers that have been loaded. Also, you must make sure that the total weight loaded does not exceed the weight capacity of the ship.

Note: weight units are tons and all weight data is floating point data

Requirements:

- Follow documentation and style guidelines for the course (see PA 1).
- Input the ship's total capacity weight.
- Input the shipping container weights one at a time.
- If the input weight is negative or zero, ignore it. Do not use the weight in calculations and do not print an error message.
- If the input weight will cause the total to exceed the ship's weight capacity, do not use the weight in calculations. Display the message "The ship is full."
- You must use a while loop.
- DO NOT use the break or continue statements.
- When the ship is full, display the
  - 1) ship's weight capacity
  - 2) number of containers loaded
  - 3) total weight loadedRemember to include meaningful text with all output.

Testing Suggestions:

Create a data file named `containers.txt` (or whatever you want to call it). When you run your program, use input redirection.

```
./cargo < containers.txt
```

Example Input File

```
47.0
8.3
9.0
-3.0
14.0
7.0
11.0
5.9
```

Note that there must always be more containers (more numbers in the input) than the ship can hold.

Example Output

```
Ship's Weight Capacity: 47.0 tons
4 containers were loaded
Total weight loaded: 38.3 tons
```

## The Second Program: `pattern.c`

- Input an integer from the user. This is the number of lines in the pattern.
- If the number is greater than 2, display a pattern of asterisks and spaces using the number of lines from the input. See examples below.

Requirements:

- Follow documentation and style guidelines for the course (see PA 1).
- Use for loops (not while loops or do-while loops)
- DO NOT use the break or continue statement.

Example output for 3 lines.

```
*
* *
*
```

Example output for 4 lines.

```
*
* *
* *
*
```

Example output for 5 lines.

```
  *
 * *
*   *
 * *
  *
```

Example output for 6 lines.

```
    *
   * *
  *   *
 *     *
*     *
 *   *
  *
```

All of the programs you submit in this class must:

- 1) Have header comments. Follow the example shown above.
- 2) Have at few inline comments.
- 3) Use consistent indentation and spacing.
- 4) Have meaningful variable names.

The grading rubric is in Canvas. Note: Grading will include a check to make sure that you did not use break or continue in your programs.

**Submit two files: `cargo.c` and `pattern.c`**

Make sure that you select the correct files and that both programs are uploaded before you submit.