

Unit6-05: Lists

Daily Assignment

Part A: Knowledge & Understanding

From the online book [Computer Based Problem Solving](#), read [Lists](#). Refer to today's page and answer the following questions in **bold**.

1. What is the main difference between a list and an array?

2. What is the name of the function to insert an element at the beginning of a list?

In Python:

In C++:

3. What is the name of the function to insert an element at the end of a list?

In Python:

In C++:

4. Insert a screenshot of an example of using lists in Python.

5. Insert a screenshot of an example of using lists in C++.

Part B: Design

Create a program called "Lists" that asks the user repeatedly to enter a mark between 0 and 100.

- The program uses a loop to enter each mark into a list of integers.
- When the user enters -1, the loop stops.
- It then calls a function called "calc_average()" in Python and "calcAverage()" in C++ that accepts a list of integers and returns the average of all the marks entered.
 - Do NOT use a hard-coded number (i.e. 10) to calculate the average.
 - In Python, use:
 - `len(list_of_marks)`

- In C++, use:
 - `listOfMarks.size()`
- To declare a list, use the following:
 - In Python:
 - `list_of_marks[]`
 - In C++:
 - `#include <list>`
 - `std::list<std::int> listOfMarks;`
- Display the average in `main()`.
 - Since your data is random, it is not possible to create test cases in advance.
 - Instead, insert 3 test cases of your program output. Then mathematically check to make sure your output is correct.

Ensure the `main()` function catches any "bad" user input so that only "valid" input is passed to the function.

*Note: The top-down design should be for the **entire program**. There should be a **flowchart for each function**.*

6. Use draw.io to create the **top-down design** for the Lists program. Insert your screenshot below.

7. Write the **pseudocode** for the Lists program.

8. Insert **at least 3 test cases**. Add more for each scenario if necessary. Make sure to include different examples showing that your program will not crash.

Your Test Case:
Your Program Output with the Above Input for Python:
Your Program Output with the Above Input for C++
Your Test Case:
Your Program Output with the Above Input for Python:
Your Program Output with the Above Input for C++
Your Test Case:
Your Program Output with the Above Input for Python:

Your Program Output with the Above Input for C++

Part C: Coding

9. Log into [Github.com](https://github.com). Inside your ICS3U organization, create a new repository called “**Unit6-05-Python**”. Make sure to add a **README** file and a **.gitignore** for **Python**. Insert the link to your new Github repository below.

10. Clone your Github repo to Codespaces.
- In Codespaces, click on the **Unit6/Unit6-05** directory. (If it is not there, create it.)
 - Right click on the folder you created above and select Open Terminal Here.
 - Type the following in the terminal window to clone your Github repository:
 - `git clone <URL of your Github repository above>`
 - Insert a screenshot of your updated directory structure proving that it worked.

11. Inside your **Unit6-05-Python** folder in Codespaces:

- Create the program above.
- Save the file as: “**lists.py**”.
- Run your program proving that it works.

- a. Run: **black ./*.py**

Insert a screenshot after you ran it proving there are no style errors.

- b. Copy and paste your code below. Use [Code Blocks](#) to colour your code if necessary. *Make sure to include header and body comments!*

- c. Insert a screenshot of your program running **after the test cases above** ↑ (in program output for Python) proving that it is working. *(Do not put anything here.)*

12. Commit your files back to Github:

- *Make sure you are in the folder where your file is located.*
- **git add -A**
- **git commit -m “Commit message”** ← The “commit message” should describe the changes you made in the code.
- **git push origin main**
- Insert a screenshot of your bash terminal after you have completed this.

Extra

13. Log into [Github.com](https://github.com). Inside your ICS3U organization, create a new repository called “**Unit6-05 CPP**”. Make sure to add a **README** file and a **.gitignore** for **C++**. Insert the link to your new Github repository below.

14. Clone your Github repo to Codespaces.
- In Codespaces, click on the **Unit6/Unit6-05** directory. (If it is not there, create it.)
 - Right click on the folder you created above and select Open Terminal Here.
 - Type the following in the terminal window to clone your Github repository:
 - `git clone <URL of your Github repository above>`
 - Insert a screenshot of your updated directory structure proving that it worked.

15. Inside your **Unit6-05-CPP** folder in Codespaces:
- Recreate the program above, but in C++.
 - Save the file as: “**lists.cpp**”.
 - Run your program proving that it works.
- a. Run: **cpplint** on your program. Fix any style errors and run cpplint until there are no style errors. Insert a screenshot after you ran it proving there are no style errors.

- b. Copy and paste your code below. Use [Code Blocks](#) to colour your code as shown in the video. *Make sure to include header and body comments!*

- c. Insert a screenshot of your program running **after the test cases above** ↑ (in program output for C++) proving that it is working. (*Do not put anything here.*)

16. Commit your files back to Github. Insert a screenshot of your terminal after you have completed this.