The Auction

Instructions

This exercise will help you practice **Separation of Concerns** by implementing a small Python app.

Read the question carefully. A lot of you lost marks in the past quiz for not reading properly.

The app must be bug-free, efficient and respect the separation of concerns. **This exercise is worth points in the scope of assignment 2.**

You may work in teams of 2-3. If you do so make sure your **names and student ids are present in comment at the top of every file you submi**t.

Problem

A small town is organizing an auction to finance building an arena for community sports. The auction has caught a lot of media attention and so the organizer believe they will need to computerize their inventory.

They are asking you to write a simple program which will prompt the user for commands to handle the inventory.

The program should be able to do the following:

- Add an item to the inventory
- Remove an item from the inventory
- Print the inventory
- Exit and save the inventory to a text file

The program should not be case sensitive with regard to item names.

Moreover, for both computer and human efficiency, the inventory should be kept sorted at all time.

Luckily for you, one of the organizer is a developper and, despite not having time to implement the app, they drew a flowchart of the program they want you to implement. You can find this flowchart on the last page of this document.

The program must offer all the functionalities mentioned. Furthermore, the input and output formats should be as presented on the next page.

Welcome message

On the start of the program, the following message should be printed.

```
== Welcome to the Auction! ==
```

Add

Here is an example of what the user should see when adding an item.

```
Please enter a command (“add”, “remove”, “print” or “exit”).
>>> add
Name the item you want to add.
>>> pair of shoes
You added Pair Of Shoes to the inventory.
```

Remove

Here is an example of what the user should see when removing an item.

```
Please enter a command (“add”, “remove”, “print” or “exit”).
>>> remove
Name the item you want to remove.
>>> pair of shoes
You removed Pair Of Shoes from the inventory.
```

If the item was not in the list, the above is still displayed, but nothing is removed from the list.

Print

Here is an example of what the user should see when printing the inventory.

```
Please enter a command (“add”, “remove”, “print” or “exit”).
>>> print
Here are the items in the inventory:
1) Pair Of Shoes
2) Brand New Bike
3) Golden Car
```

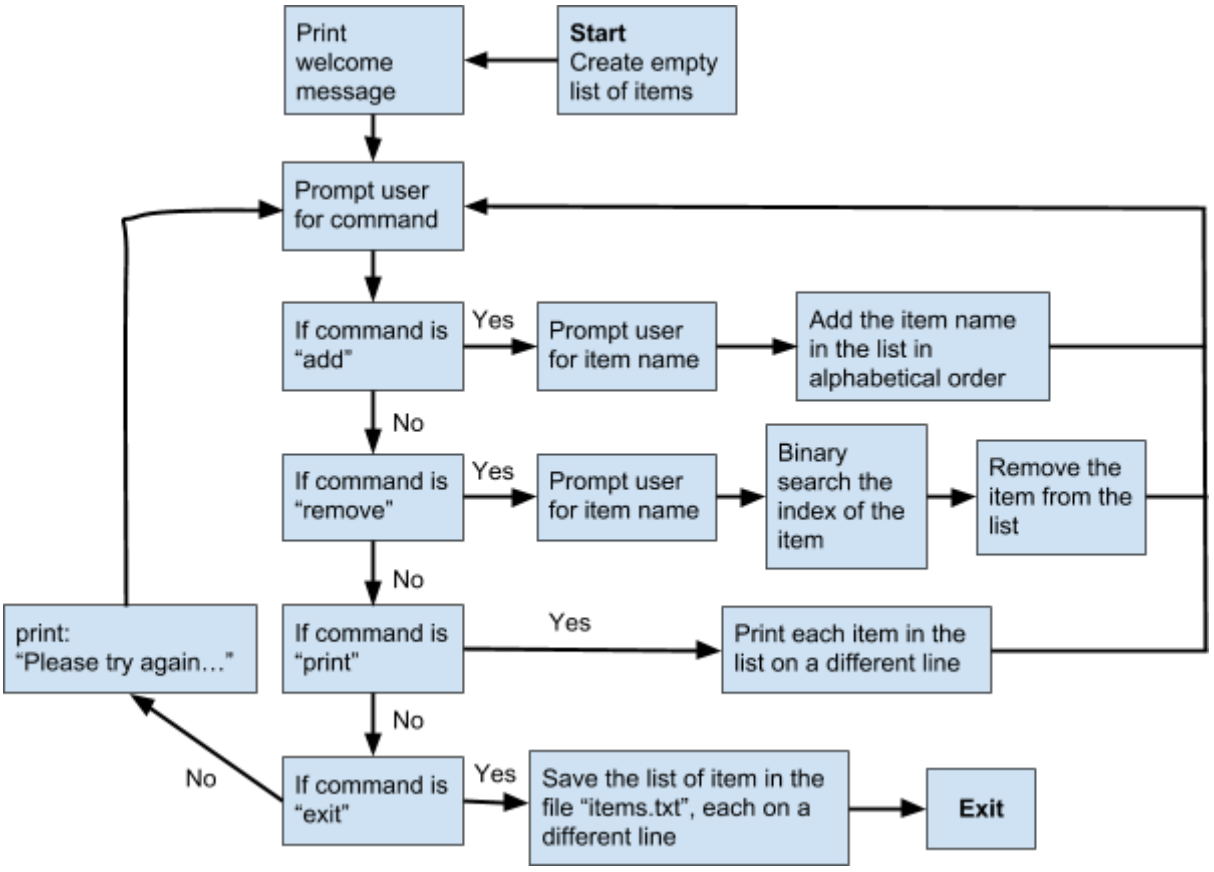
Exit

Here is an example of what the user should see when exiting the program.

```
Please enter a command (“add”, “remove”, “print” or “exit”).
>>> exit
Your inventory will be saved to “items.txt”. Goodbye!
```

The content of the file should be the same as the output from the print command.

Flowchart


Print welcome message
Start
Create empty list of items
Prompt user for command
If command is “add”
Yes
Prompt user for item name
Add the item name in the list in alphabetical order
No
If command is “remove”
Yes
Prompt user for item name
Binary search the index of the item
Remove the item from the list
No
print: “Please try again…”
If command is “print”
Yes
Print each item in the list on a different line
No
No
If command is “exit”
Yes
Save the list of item in the file “items.txt”, each on a different line
Exit