

# Programming Assignment #1 Specification

CPSC221, 2017W1

*Due: 23:00 Tuesday, October 03, 2017*

September 18, 2017

## Introduction

Aylo, an eccentric entrepreneur would like to open up a storefront for shishkebabs, called Speed Kabab. However, Aylo does not want to hire any extraneous employees for Speed Kabab, and would instead prefer the customers interact with a robot.

Your task is to write the software for the robot that handles inventory management. Your code must be made with C++ using Clang++ as a compiler. A skeleton implementation has been provided.

Terminal input and output will be done using standard IO, so `cin` and `cout` will be helpful. An automated grader will be used, so make sure your output and/or return values are formatted correctly.

Note that only so much time will be allotted for your program to run, so you may need to consider the efficiency of your implementation.

## Specification

### Food

Speed Kabab will stock shishkebabs made from five different ingredients:

beef, tomatoes, mushrooms, (green) pepper, and squash.

Each of these items is represented by the first single upper- or lower-case character of its name:

- beef is 'b' or 'B',
- tomato is 't' or 'T',
- mushroom is 'm' or 'M',
- pepper is 'p' or 'P', and
- squash is 's' or 'S'.

Shishkebabs are made by placing items from the ingredients list onto skewers that can contain up to 8 ingredients. Note that the kebabs can contain duplicate items (e.g. a kebab could consist of just five beef items), or a kebab could contain no items at all (just a skewer). Furthermore, while customers don't care about the order of the items on the kebab, the robot still has to keep track so that the items on the grill can be shown to the customer properly.

When the grill is running low on kebabs, the assistant chef, Lamao, will come by to add more kebabs onto the grill. However, the robot cannot tell the difference between Lamao and any other customer, so the interface Lamao uses to add items to the grill is the same as that of the customer.

## Grill

The grill is used to hold the supply of all kebabs which are laid out in a single row from left to right. Customers may buy any kebab on the grill one at a time.

Whenever an item is added to the grill, it will be placed to the right of all existing kebabs. Kebabs can be removed from any position in the grill; removing a kebab affects the counting order of all kebabs to its right although the kebabs do not need to be physically shifted.

## Customer Interaction

Customers (or Lamao) may ask the robot for one of six things:

To see the grill:

Aylo uses a tall counter to hide some poor food safety practices from the customers, so in order to let customers know what is currently available on the grill, the robot will attempt to display a visual representation of the grill on a screen. The details of this are specified below.

To buy a kebab at a specified position on the grill:

The customer can specify a number  $x$ , referring to the  $x$ th kebab from the left side of the grill. Note that the customer is not expected to be a computer scientist, so  $x$  will use 1-based indexing. If a kebab exists in position  $x$ , the robot will sell the kebab to the customer. If not, the robot will give the customer an error.

To buy a kebab with a list of required items:

The customer will type in a list of items they want on their kebab, in no particular order. If a kebab is present on the grill meeting the given criteria (possibly containing ingredients other than the minimum required items), the robot will sell that kebab, thereby removing it from the grill. In the case of more than one kebab meeting the criteria, the robot will sell the kebab furthest to the left. If no kebab is present on the grill meeting the given criteria, then the robot will be unable to satisfy this customer's request and the customer may make another request or wait for Lamao to put a new kebab on the grill.

To add a shishkebab with a list of items:

Lamao will type in a list of items in the same way that a customer specifies their wanted ingredients list. The robot builds a kebab containing the specified ingredients *in the order specified* and places it on the grill at the rightmost position.

To swap the positions of two kebabs on the grill:

To move a kebab to a hotter part of the grill, or to increase visibility of old kebabs, or simply as an attempt to make the robot work for its pay on a slow day, Lamao may order the robot to swap positions of two kebabs on the grill. Two 1-indexed positions are specified, and if the positions are valid then the kebabs will be swapped without affecting the positions of the other kebabs.

To sort the kebabs by (decreasing) order of meat items:

Aylo reduces his operating costs by using some nearly-expired meat in his kebabs, but luckily his customers do not seem to notice, and even prefer buying kebabs with more meat items.

As a win-win scenario for all, Aylo added a feature to his robot to rearrange any existing kebabs on the grill such that kebabs containing more beef items are placed towards the left side of the grill, and items with fewer beef items are placed towards the right side of the grill, so customers can see them more easily.

## Displaying the grill contents

The kebab contents and positions on the grill will be displayed as text on standard output. The numeric positions will be displayed from left to right, and below each position, the ingredients of each kebab will be displayed vertically. This has been done for you already. However, the formatting routine requires the kebab contents to be delivered in a vector of strings.

## Input

All input will be through standard input (i.e. the terminal). Inputs fall into three categories:

Query: An integer number corresponding to a choice on the robot's root menu.

Grill position: An integer number corresponding to a 1-indexed position on the grill.

Kebab ingredients: A string containing 0 or more characters. Any characters are allowable in this string, but kebab construction will use up to the first 8 valid ingredients only.

For example, the string "hBs3 Px/=mMt" will result in a kebab with the ingredients "BsPmMt", while the string "BBBBBBBBBBBBBBBB" will result in a kebab with the ingredients "BBBBBBBB".

## Hints

Class declarations have been provided for you. You can use the given files to complete the project. Please read the comments for each function carefully for information about required inputs, outputs, and behaviours.

LinkedList Nodes and (in most cases) Kebabs will be created in dynamic memory. Take special care to ensure that the items in dynamic memory are deleted at an appropriate time.

Please write the CSID of all contributing members in a file called `partners.txt`, exactly and only one CSID per line.

The classes used in this assignment make extensive use of the vector class, which operates like a resizable array and can be accessed using the `[ ]` operator. Items can be added to the end of the vector using the `push.back()` function, and the number of items stored in the vector can be checked using the `size()` function. Please see <http://www.cplusplus.com/reference/vector/vector/> for additional details about usage.

## Deliverables and submission

Your submission should consist of a ZIP archive containing the following files:

- `partners.txt`
- `kebab.cpp`

- `linkedlist.cpp`
- `grill.cpp`

Instructions for submission will be found on the PA1 page at the course website.