Project 3: Production Planning

Due Friday 26 September at 11pm

Background

As an operations engineer for Dull Computer, Inc., you are responsible for managing the factory schedule and responding to customer inquiries about order fulfillment. Specifically, when a customer enters a proposed order, your job is to figure out whether it can be completed within the requested time, or if not, provide options for a partial or delayed delivery.

Your Assignment

Your company is quite small, capable of assembling at most 100 computers (PCs) per day. You have all the parts on hand to build a computer, except for the microprocessor chips which come in weekly shipments from the manufacturer.

You sell computers in two different speed configurations: **fast** and **blazing**. Every seven days, Dull receives a shipment of 500 fast chips that are used to make fast computers and 300 blazing chips that are used to make blazing computers. The shipments start on day 0, that is, the factory has 500 fast chips and 300 blazing chips available for production on the first day. Unfortunately, the factory needs to be retooled to switch between making fast and blazing computers (this must be done on the last day of the week at midnight), so each week the company can either produce fast computers or blazing computers, but not both.

A customer order specifies a quantity of PCs requested of each speed, and a due date, specified in terms of number of days from the present (day 0). Your response should indicate what portion of the order can be completed in time, and the *makespan*, defined as the number of days it would take to complete the entire order.

Your production plan must operate as follows. You decide which speed of PCs to make first, and then keep manufacturing that speed until either (1) you run out of chips for that speed, or (2) you have completed the order quantity for that speed. At that point, if there is an outstanding order quantity for the other speed, you switch over production at the next opportunity (i.e., for the following week). If the other speed is already done, then you stick with the current speed until you have manufactured all PCs requested in the order.

Recall there are three key constraints you need to satisfy.

1. In a given week, you may make PCs of only one speed. You can decide which speed to start with, but after that the production must follow the specified plan structure.
2. Your weekly production is limited by the number of microprocessor chips of the associated speed in inventory.
3. Your daily production is limited to 100 PCs per day. The factory operates seven days per week.

Also recall that every week, a new shipment of microprocessor chips arrives, and is added to inventory in time for that week's production (i.e., starting day 8, 15,…). In any given week, your production of PCs may be limited by factory capacity or chip inventory.

Your specific task is to write a program that extracts (without prompt) from the user (i.e., the customer) three numbers, describing an *order*:

1. Quantity of fast PCs requested (a positive whole number).
2. Quantity of blazing PCs requested (a positive whole number).
3. Due date, in number of days (a positive whole number) from the present.

Your program should then calculate two production plans, one where the factory starts producing fast PCs, and the other starting with blazing PC production. You report on each plan, and then indicate the makespan of the better plan (that is, the one that completes the entire order in the shortest time possible, whether or not meeting the due date is feasible). If the plans producing fast PCs first or blazing PCs first have the same makespan, then produce fast first.

For each plan, your program must calculate the following values:

1. The respective numbers of fast and blazing PCs that can be delivered by the due date.
2. The overall makespan, in days, that it would take to make all the PCs requested (fast and blazing).
3. The number of production switchovers required.

Your output must mimic the pattern of the following example (exact text and punctuation). Suppose the customer requests 805 fast PCs and 222 blazing PCs, due on day 6. Then you would output:

```
Start with Fast production.
Deliver on time 500 fast PCs, 0 blazing PCs.
2 switches.
Start with Blazing production.
Deliver on time 0 fast PCs, 222 blazing PCs.
1 switches.
Shortest makespan (Blazing) is 16 days.
```

As another example, suppose the customer requests 2500 fast PCs and 1500 blazing PCs due on day 50. Then you would output:

```
Start with Fast production.
Deliver on time 2500 fast PCs, 1500 blazing PCs.
3 switches.
Start with Blazing production.
Deliver on time 2500 fast PCs, 1500 blazing PCs.
3 switches.
Shortest makespan (Fast) is 44 days.
```

One more example. For a request of 0 fast and 999 blazing, due on day 21:

```
Start with Fast production.
Deliver on time 0 fast PCs, 900 blazing PCs.
1 switches.
Start with Blazing production.
Deliver on time 0 fast PCs, 900 blazing PCs.
0 switches.
Shortest makespan (Fast) is 22 days.
```

Tips and Coding Requirements

As always, it is highly advisable to work out the logic of your program in advance, before reducing your solution to C++ code. This project is much more complex than our previous assignments, so it is important to break up your program into multiple functions that address separable tasks in the assignment. In order to emphasize the use of procedural abstraction, your solution is **required** to use—in a nontrivial and appropriate way—at least three functions (not counting `main`), and your `main` itself may contain at most 12 lines of code.

Submission Requirements

Instructions for submitting to the auto-grader will be posted separately.

Grading

10 points for submitting something that compiles.

10 points for employing proper style, including the use of functions as mentioned above. We will also check for adherence with the course C++ style guidelines (to be posted).

35 points for correctness on a variety of test inputs.