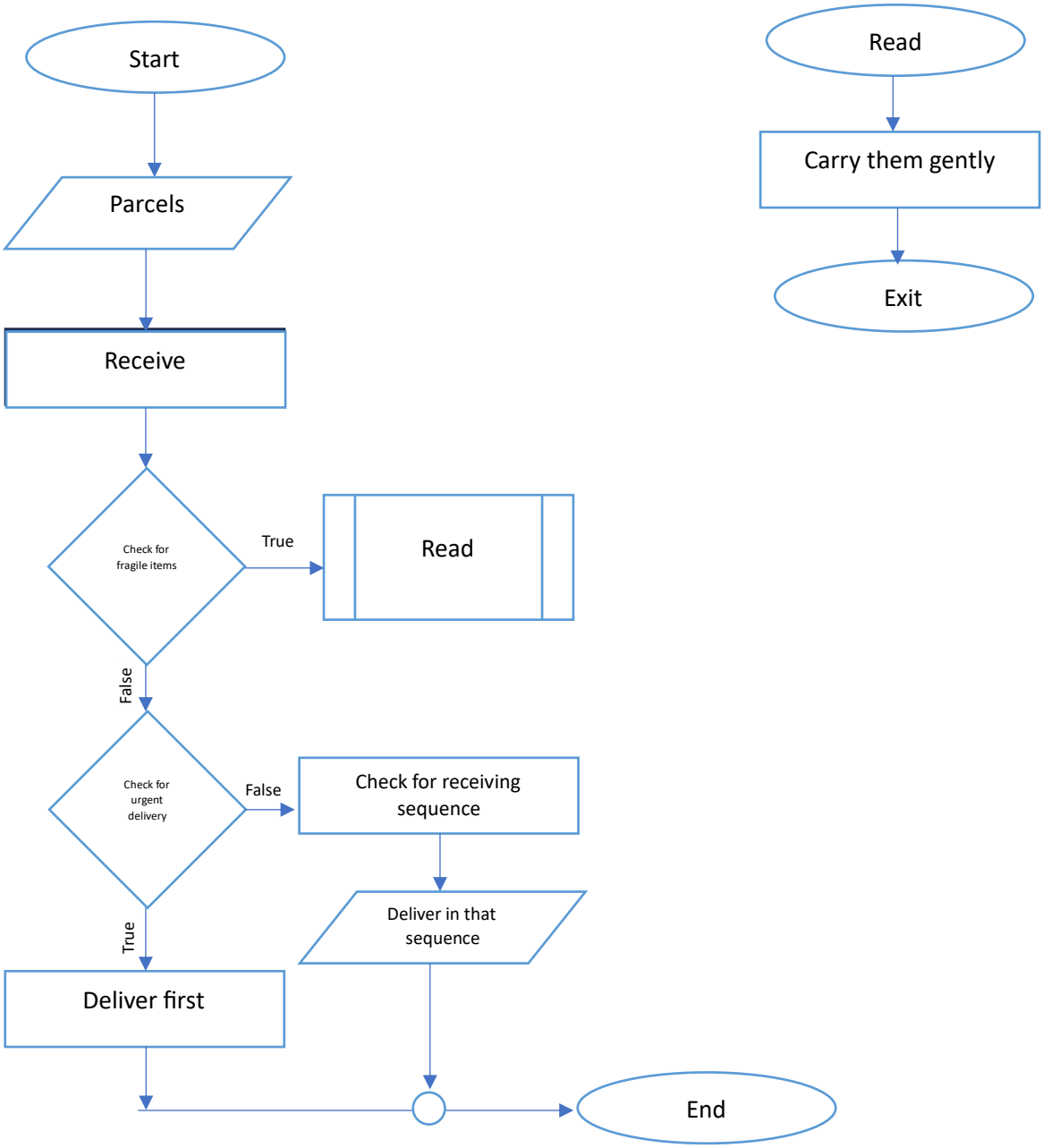
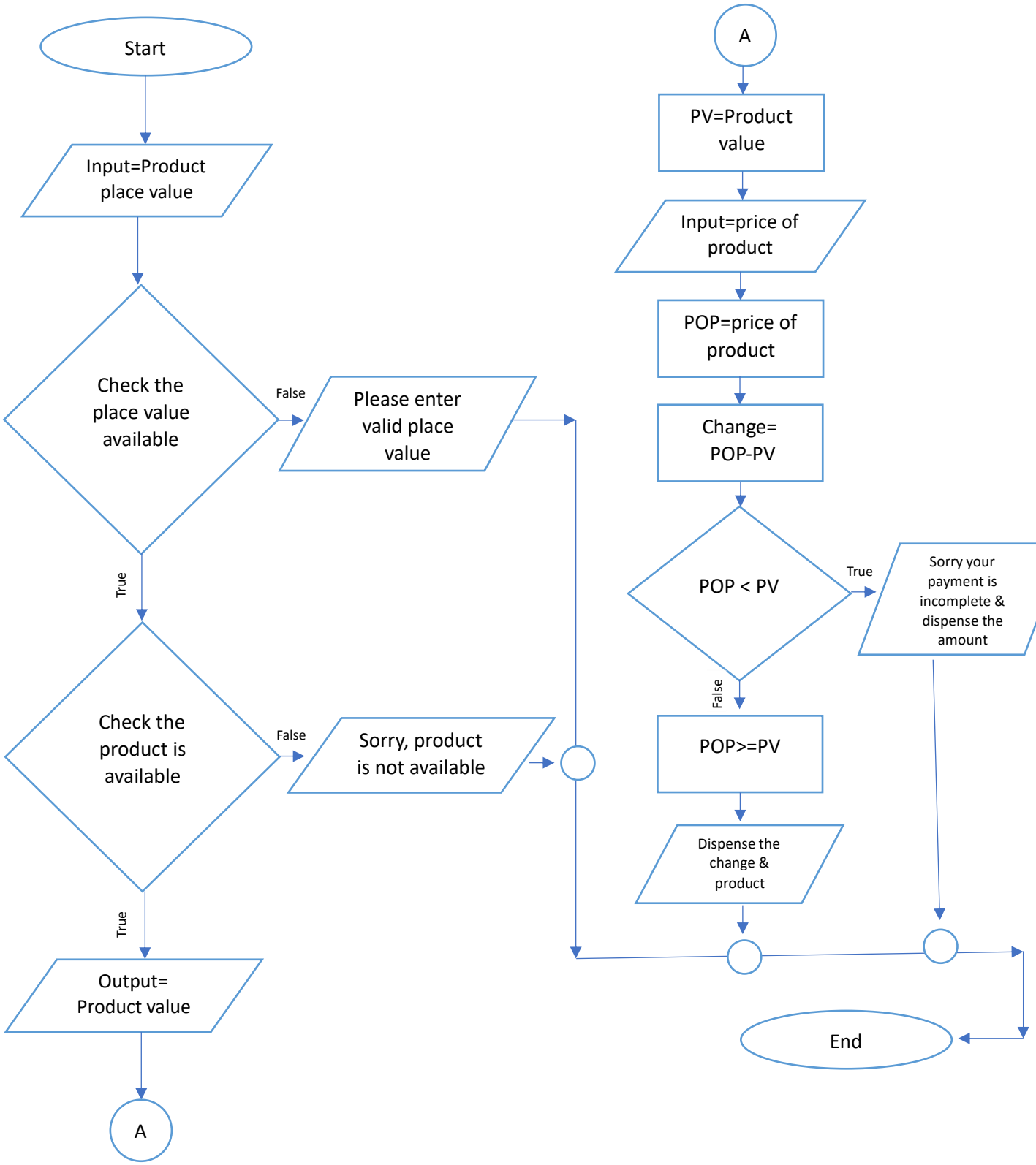
FLOWCHARTS (CONT.)

Some Problems to Solve

1. **You are working in a logistics company responsible for delivering packages. Design a flowchart to manage the process of receiving, sorting, and delivering packages. Include decision structures for handling fragile items and urgent deliveries.**



2. Imagine you are automating the process of a vending machine. Create a flowchart that includes decision points for user input, selecting products, accepting payment, and dispensing the correct item. Include error-handling for invalid inputs and insufficient funds.



PSEUDOCODE

Some problems to solve.

1. **Write pseudocode to find the smallest number among three given variables. Implement a decision-making structure to compare the variables.**

```
START

// Input/
INPUT number1
INPUT number2
INPUT number3

// Conditional Statements
IF
    number1 < number2 < number3
THEN
   PRINT " number1 is smallest number"
IF
number2 < number3 < number1
THEN
   PRINT "number2 is smallest"
ELSE
PRINT "number3 is smallest"

END
```

2. **Develop pseudocode for a basic calculator that performs multiplication and division. The pseudocode should prompt the user for two numbers and an operator, then display the result of the operation.**

```
START

// Input
INPUT number1
INPUT number2

// Output
Select the operation
   1) Multiplication
   2) Division

// Conditional Statements
IF
     User select Multiplication

THEN
   PRINT number1*number2

ELSE
PRINT number1/number2

END
```

ALGORITHM

Some Problems to Solve:

1. **Write an algorithm to determine whether a number is a prime number. The algorithm should iterate through possible divisors and determine if the number has any divisors other than 1 and itself.**
   1) Ask user to enter the **number**
   2) Set **num** to **number**
   3) Set **i** to 2
   4) If **num** is equal to 2 or 3
   5) Display it's a prime number
   6) Else
   7) While(num is greater than 3)
   8) Do(Set **i** to **i** + 1 then divide num by **i** until **num**)
   9) If **num** is divisible by **i** then Display it's not a prime number
   10) Else Display it's a prime number

2. **Create an algorithm that asks the user for a day number (1-365) and outputs the corresponding day of the week, assuming that January 1st is a Monday.**
   1) Ask user to enter the number between(1-365)
   2) Divide number by 7
   3) If remainder is 0 than print (Monday)
   4) If remainder is 1 than print (Tuesday)
   5) If remainder is 2 than print (Wednesday)
   6) If remainder is 3 than print (Thursday)
   7) If remainder is 4 than print (Friday)
   8) If remainder is 5 than print (Saturday)
   9) else print (Sunday)