1. Add code below so that the vector \texttt{fib} contains all the Fibonacci numbers up to and including the first such number that is greater than one million. You should not use any variables other than \texttt{fib}, nor make any calls to the \texttt{length} function. Instead, build \texttt{fib} by vector growing, and make use of the \texttt{end} keyword.

\begin{verbatim}
fib = [0, 1];
% Add the necessary code here
\end{verbatim}

2. A certain fast food restaurant in Collegetown has the following menu and prices:

<table>
<thead>
<tr>
<th>Main Dish:</th>
<th>1. Veggie Burger</th>
<th>$4.00</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2. Hamburger</td>
<td>$3.50</td>
</tr>
<tr>
<td>Beverage:</td>
<td>7. Water</td>
<td>$1.00</td>
</tr>
<tr>
<td></td>
<td>8. Soda</td>
<td>$1.20</td>
</tr>
<tr>
<td>Super Meal:</td>
<td>Main dish + salad + beverage Price of main dish + $2.00</td>
<td></td>
</tr>
<tr>
<td>(must include all items; no changes allowed)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A customer with a coupon receives a 10\% discount. A customer can have at most one main dish and at most one beverage.

Complete the program below to calculate and print the price of an order. Assume that the cashier will not allow any changes to the Super Meal and will answer the prompts correctly.

\begin{verbatim}
meal = input('Super Meal combo? (1 if yes, 0 if no) ');
dish = input('Main dish number? (1 or 2; 0 if none) ');
drink = input('Beverage number? (7 or 8; 0 if none) ');
coupon= input('Customer is using a coupon? (1 if yes, 0 if no) ');
\end{verbatim}
3. For each of the following sub-problems, complete the program below so that it produces the desired result. You should not modify the programs in any way, only fill in the blanks that are provided. In every case you will need to use for loops with step other than one.

(a) The following program reads an integer \(k\), and outputs all the multiples of \(k\) up to 1000.

```matlab
k = input('Please enter a positive integer smaller than 1000: ');
for j = ______________________
    fprintf('%d ', j);
end
fprintf('
');
```

(b) The following program reads in a real number \(x\) and an integer \(N\), and computes the sum \(\sum_{k=0}^{N} \frac{(-1)^k x^{2k}}{(2k)!}\) to the first \(N\) terms. (This sum converges to \(\cos(x)\) as \(N \to \infty\).)

```matlab
x = input('Please input a real number between 0 and \(\pi/2\): ');
N = input('Please input a positive integer: ');
sum = 0;
for j = _______________________
    sum = sum + (-1)^(j/2) * x^j / factorial(j);
end
fprintf('The sum of the first %d terms is %12.8f
', N, sum);
```

(c) The following does the same thing as in part (b), but this time we are not allowed to use exponentiation and the \texttt{factorial} function, and must compute these explicitly.

```matlab
x = input('Please input a real number between 0 and \(\pi/2\): ');
N = input('Please input a positive integer: ');
sum = 1; % Explicitly assign the first term (when j=0)
sign = 1; % The sign of a term, either 1 or -1
jfact = 1; % Current value of j!
xtoj = 1; % Current value of \(x^j\)
for j = _________________________
    sign = ________________________________;
jfact = ________________________________;
xtoj = ________________________________;
    sum = ____________________________________________;
end
fprintf('The sum of the first %d terms is %12.8f
', N, sum);
```