Puzzle Coding

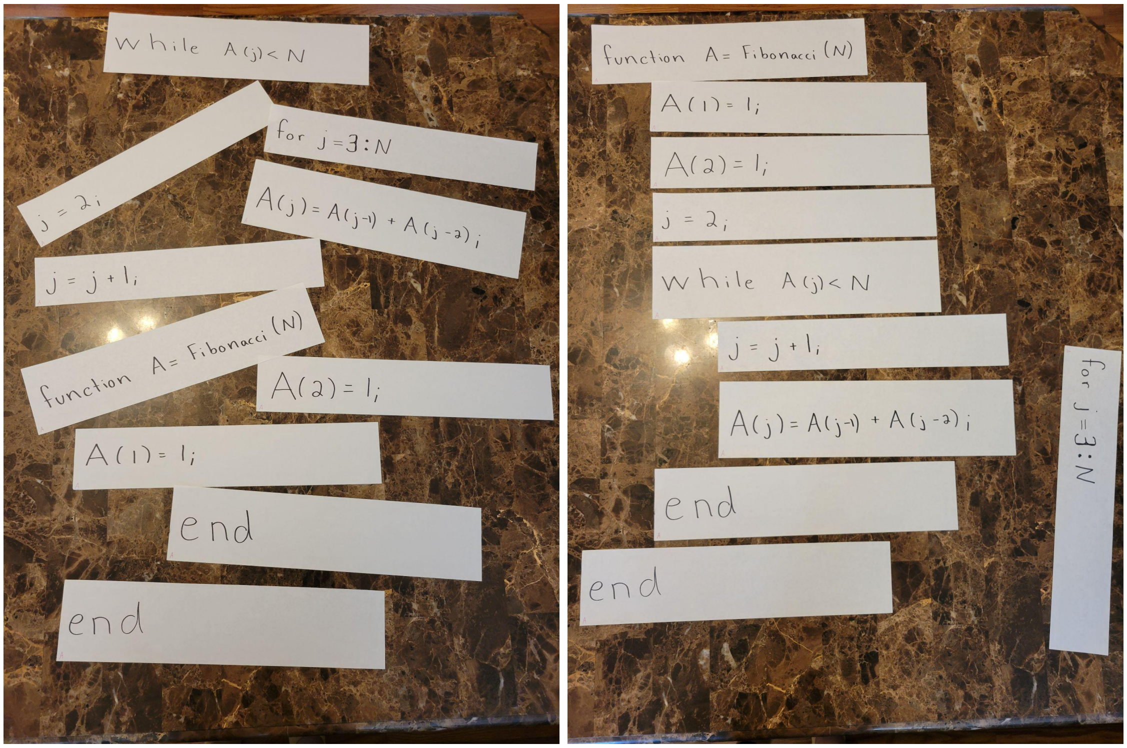
Learning Objectives

- Review usage of for loops while loops, and if statements.
- Review of programming logic and logical operators.
- Review proper coding techniques including indentation and documentation.
- Practice testing and debugging techniques.
- Determine test cases.

Materials

- Dry Erase Markers
- 2 Whiteboards per Group
- Sticky Notes per Group
- Code Strips (Strips of Paper with Lines of Code Written on Them) for Each Group

Example Images



```
while A(j)< N
j = 2;
for j=3:N
A(j)= A(j-1) + A(j-2);
j = j+1;
function A= Fibonacci (N)
A(2)= 1;
A(1)= 1;
end
end
```

```
function A= Fibonacci (N)
  A(1)= 1;
  A(2)= 1;
  j = 2;
  while A(j)< N
    j = j+1;
    A(j)= A(j-1) + A(j-2);
  end
end
for j=3:N
```

Figure 1: This figure illustrates the steps of this activity. The image on the left shows the strips of code in random order. The goal is for students to rearrange the strips of code to solve a problem. The solution to coding scenario 1 is shown in the image on the right.

Puzzle Coding

Directions

1. Teams of students use the code strips to create a function or script that solves each of the problems (See Figure 1, left).
2. When arranging the code strips, indent the strips appropriately. This means indent any lines of code that are within a function, loop, or if statement. (See Figure 1, right).
3. Use sticky notes to add comments to the lines of code.
4. Determine appropriate test cases to check your functions.
5. Desk check your code. Use one of the provided whiteboards as your command window and one to use as their workspace. “Run” your program by:
    - ○ Writing appropriate command(s) into the command window.
    - ○ Reading through your function line by line.
    - ○ Track the current value of all defined variables in the workspace whiteboard.
6. Use MATLAB to verify your answer is correct.

Coding Scenarios

1. Write a function to find all of the Fibonacci numbers that are less than *N*. Recall that the Fibonacci numbers are found by computing the sum of the two previous numbers in the sequence.
2. Write a function to compute the first *N* Fibonacci numbers. Recall that the Fibonacci numbers are found by computing the sum of the two previous numbers in the sequence.
3. Write a function that takes a numerical test score and determines the grade earned according to the standard ten-point grading scale.
4. Write a function to determine if a number is prime.

Code Strips for Scenarios 1 and 2

```
function  A=Fibonacci(N)
for j=3:N
while  A(j)j=2;
j=j+1;
A(j)=A(j-1)+A(j-2);
A(1)=1;
A(2)=1;
end
end
```

Puzzle Coding

Code Strips for Scenario 3

```
function  grade=computeGrade(score)
if  score>=90
elseif score>=80
elseif score>=70
elseif score>=60
else
else
if (score>=0 && score<=100)
if (score>=0 || score<=100)
grade='Illegal Input';
grade='A';
grade='B';
grade='C';
grade='D';
grade='F';
end
end
end
```

Code Strips for Scenario 4

```
function out=isPrime(n)
out = false;
out = false;
out = true;
out = true;
for k=2:n-1
if n==0 || n==1
if rem(n,k)==0
else
break
end
end
end
end
```

Puzzle Coding

Post-Activity Discussion

1. What skills have you learned from engaging in this task that you may be able to use to help you in your future programming assignments?
2. What is the importance of determining good test cases? How did you determine your test cases?
3. What kind of errors did you make while working through this task? How did you fix your errors?
4. What's the purpose of the workspace, editor, and command window in MATLAB? What is the equivalent object in this activity?
5. Explain the importance of indentation when writing a program.

Extension

If you have students that are able to quickly work through are or are highly engaged with this activity, you could ask them to create their own coding scenario by writing sample code on strips of paper for another group and have them arrange the strips in order.

References

[1] Barnes, J., & Libertini, J. M. (2018). *Tactile Learning Activities in Mathematics: A Recipe Book for the Undergraduate Classroom* (Vol. 54). American Mathematical Soc.