Data Analysis and Hours of Sleep

**Algebra**

**Unit 3 Line Master 5a**

**Part 1: Determining the minimum number of hours slept
in one week**

Use this link to access the code for determining the least number
in a data set.

<https://scratch.mit.edu/projects/702461611/editor/>

If you have a Scratch login, save the project in your Scratch account by selecting **Remix** at the top of the screen.
A login is not required to work with the code, but you will not be
able to save your changes without it.

This code is in the code editing window:



This application prompts the user for the number of hours
they slept each night for 7 nights.

It will then determine the least number and indicate that this
is the minimum hours of sleep they had on any night that week.

 Data Analysis and Hours of Sleep
 (cont’d)

**Algebra**

**Unit 3 Line Master 5b**

While finding the least of 7 numbers is not challenging to do
without a computer, you will be altering this code to determine additional information.

Execute the code by clicking on the **green flag** .

Enter these numbers, one at a time, to test the application:

8

7

6

9

8

7

8

Did the application output the least number from the list?

Here is pseudocode for this application:

*Start*

 **numNights** = 7

 **hoursOfSleep** = 0

 **minimumHours** = 24

 *repeat* **numNights**

 output “Number of Hours:”

 store user input as **hoursOfSleep**

*if* **hoursOfSleep** < **minimumHours** *then*

 **minimumHours** = **hoursOfSleep**

 *end if*

 *end repeat*

*end program*

 Data Analysis and Hours of Sleep
 (cont’d)

**Algebra**

**Unit 3 Line Master 5c**

The variables used are listed and described below.

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Purpose** | **Initial Value** |
| numNights | Contains the number of nights for which the hours of sleep will be analyzed. This can be changed depending how many nights of data the user wishes to explore. | 7 |
| hoursOfSleep | Contains the hours of sleep entered by the user when the code is executed. As this data is collected for each night, the application checks if the newest number is less than the previous number entered. | 0 |
| minimumHours | This holds the least number of hours entered by the user. As the code is executed, its value will be replaced when a lesser number is entered.  | 24 |

 Data Analysis and Hours of Sleep
 (cont’d)

**Algebra**

**Unit 3 Line Master 5d**

Let’s take a closer look at this code to understand how it works:

** 

* After the first subprogram called **resetVariables** is executed, the code inside the repeat is executed.
The sprite displays a prompt for the user to enter the number of hours slept on the first night.
* After the user enters this number, the application checks if it’s less than the current value of **minimumHours**, which is set to 24.
If it is less than 24, the **minimumHours** variable holds the current value of **hoursOfSleep**.
* The second time through the loop, the code will again prompt for the number of hours slept and will again check if the number entered by the user is less than the current value of **minimumHours**.
* The code will continue to do this, until all 7 nights of sleep data have been entered. When the application has finished, the variable **minimumHours** will hold the least number of hours and it will be displayed on the stage.

 Data Analysis and Hours of Sleep
 (cont’d)

**Algebra**

**Unit 3 Line Master 5e**

**Part 2: Determining the maximum number of hours slept
in one week**

Alter the code to include determining the maximum number of hours slept in one week.
Try adding this feature to the code now.
If you’re stuck, see the pseudocode at the end of this Master.

If you’re still unsure of what to do, read Master 6 for guided steps
you can follow.

Run the code using the same set of numbers you entered previously.
Is the maximum displayed?
If not, check your code for an error.

**Part 3: Determining the total number of hours slept
in one week**

Alter the code to include determining the total number of hours slept—that is, the sum of all the hours entered for the 7 nights.
If you’re stuck, see the pseudocode at the end of this Master.

If you’re still unsure of what to do, read Master 6 for guided steps
you can follow.

Run the code using the same set of numbers you entered previously.
Is the total displayed?
If not, check your code for an error.

 Data Analysis and Hours of Sleep
 (cont’d)

**Algebra**

**Unit 3 Line Master 5f**

**Part 4: Determining the mean number of hours slept
in one week**

Alter the code to include calculating the mean number of hours
slept in one week.
If you’re stuck, see the pseudocode at the end of this Master.

If you’re still unsure of what to do, read Master 6 for guided steps
you can follow.

Run the code using the same set of numbers you entered previously.
Is the mean displayed? Use a calculator to check.
If not, check your code for an error.

**Extensions:**

Alter the application to:

* Allow the user to indicate for how many nights they would like
to enter data.
* Change the Stage to improve how the output is displayed.
* Provide feedback to the user based on the number of hours slept. For example, if the mean is less than 8 h, provide tips on how to
try to get more sleep or share information about the effects of not getting enough sleep. Conditional statements are useful for this.
* Create a list called userData that stores and displays the
user’s input.
* Incorporate additional subprograms in your application.
* Incorporate data about a different topic instead of hours of sleep (Master 7 provides support for this).

 Data Analysis and Hours of Sleep
 (cont’d)

**Algebra**

**Unit 3 Line Master 5g**

**Hints:**

Here is the pseudocode for Part 2 to determine the maximum number of hours slept:

*Start*

 **numNights** = 7

 **hoursOfSleep** = 0

 **minimumHours** = 24

**maximumHours** = 0

 *repeat* **numNights**

 output “Number of Hours:”

 store user input as **hoursOfSleep**

 *if* **hoursOfSleep** < **minimumHours** *then*

 **minimumHours** = **hoursOfSleep**

 *end if*

*if* **hoursOfSleep** > **maximumHours** *then*

 **maximumHours** = **hoursOfSleep**

 *end if*

 *end repeat*

***end program***

 Data Analysis and Hours of Sleep
 (cont’d)

**Algebra**

**Unit 3 Line Master 5h**

Here is the pseudocode for Part 3 to determine the total number
of hours slept:

*Start*

 **numNights** = 7

 **hoursOfSleep** = 0

 **minimumHours** = 24

**maximumHours** = 0
**totalHours** = 0

 *repeat* **numNights**

 output “Number of Hours:”

 store user input as **hoursOfSleep**

 **totalHours** = **totalHours** + **hoursOfSleep**

 *if* **hoursOfSleep** < **minimumHours** *then*

 **minimumHours** = **hoursOfSleep**

 *end if*

*if* **hoursOfSleep** > **maximumHours** *then*

 **maximumHours** = **hoursOfSleep**

 *end if*

 *end repeat*

***end program***

 Data Analysis and Hours of Sleep
 (cont’d)

**Algebra**

**Unit 3 Line Master 5i**

Here is the pseudocode for Part 4 to determine the mean number
of hours slept:

*Start*

 **numNights** = 7

 **hoursOfSleep** = 0

 **minimumHours** = 24

**maximumHours** = 0
**totalHours** = 0
**meanHours** = 0

 *repeat* **numNights**

 output “Number of Hours:”

 store user input as **hoursOfSleep**

 **totalHours** = **totalHours** + **hoursOfSleep**

 *if* **hoursOfSleep** < **minimumHours** *then*

 **minimumHours** = **hoursOfSleep**

 *end if*

*if* **hoursOfSleep** > **maximumHours** *then*

 **maximumHours** = **hoursOfSleep**

 *end if*

 *end repeat*

**meanHours** = **totalHours**/**numNights**

***end program***