Labs

(3) Level A	(3) Level B	(2) Level C
Grade Average Version Two	Sentence Information	Worker Ant
Data Set	Test	Tic Tac Toe
Leap Frog	Selection Sort	Matching Game
Treasure Hunt	Maze	

Grade Average Version Two

Write a program that allows the user to enter the number of grades they received in a class and then enter each grade individually. The program will sum up all the grades and display the student's average and grade letter.

The grades will need to be stored into an array of doubles.

Rubric (counts as 1 other grade)

Points	Task
10	Gets number of grades
20	Array created to correct size
20	Gathers and stores the grades
20	Correct average
20	Correct grade letter
10	Formatting and neatness

Data Set

Write a program that will allow the user to manipulate a 1D. The array will store 10 integer values. The program will need to have a looping menu with the following options: reassign array, change value, print array, total, average, maximum, minimum and exit.

Menu Options:

- Reassign Array – Lets the user fill in values for all the elements
  - Do this with a for loop
- Change 1 element – Lets the user change 1 element
  - Gather an index and value
- Print Array
  - Do this with a for each loop
- Total
  - Do this with a for each loop
- Average
  - Prints the average value of the set
- Max
  - Prints the largest value of the set
- Min
  - Prints the smallest value of the set
- Exit
  - Closes the program

Rubric (counts as 1 other grade)

Points	Task
10	Menu
10	Reassign, done with for
10	Change single element
10	Print array, done with for each
10	Total, done with for each
10	Average
10	Max
10	Min
10	Exit
10	Formatting and neatness

Leap Frog

Write a program that will display the number of hops it would take a frog to complete a course containing open ground and rocks. If the course starts or ends with a rock the program will print out "Invalid course."

When a course is valid the frog will hop uses the following rules:

- The next space is empty
  - Frog hops 1 time and moves forward 1 space
- The next space has a rock
  - The frog hops 1 time, jumping all the rocks between it and the next space.

Inputting a course:

- Data is entered as a String (use nextLine)
  - The String will then need to be converted to a char array
- ' ' is used for an empty space
- 'R' is used for rock

The data will be read as a String and then converted to a char array for processing.

Rubric (counts as 1 other grade)

Points	Task
20	Gets course as a String
20	Convert course to char array
20	Correct number of hops for valid courses
20	Displays invalid course for invalid courses
20	Formatting and neatness

Treasure Hunt

Write a program where the user enters a String containing only the ‘N” letters and possibly a single ‘T’. When there is no ‘T’ the program will print “No Treasure. When there is a ‘T’ the program will tell the user the index of the ‘T’ (treasure).

The data will be read as a String and then converted to a char array for processing.

Rubric (counts as 1 other grade)

Points	Task
20	Gets data
20	Converts the data to a char array
20	Correct when there is no treasure
20	Correct when there is a treasure
20	Formatting and neatness

Sentence Information

Write a program that will allow the user to enter a String containing one or more words. The program will then tell the user how many words the sentence has, how many characters it has, how vowels it has and how many consonants it has.

Rubric (counts as 1 minor grade)

Points	Task
10	Gets a String
20	Correct word count
10	Correct character count
20	Correct vowel count
20	Correct consonant count
20	Formatting and neatness

Test

Write a program that allows a student to take a test. The test will consist of ten multiple choice questions.

The data will use 3 arrays of size 10:

- A String array that stores the questions
- A char array that stores the correct answers
- A char array that stores the student's answers

After the test has been taken the program will tell the user how many they got right, their score and their letter grade.

Rubric (counts as 1 minor grade)

Points	Task
10	Correctly loads the question array
10	Correctly loads the correct answer array
10	Displays questions and allows the students to answer them
15	Correctly stores the student's answers
15	Correct right
15	Correct score
15	Correct letter grade
10	Formatting and neatness

Selection Sort

Write a program that will create an array of size 10 and fill it with random numbers from 1 to 1000. The program will print the array once it has random numbers. Next, the program will sort the array using a selection sort. Finally the program will print the array after the sort.

Selection Sort Steps

- **Find the location of smallest value from 0 to the end of the list**
- **Swap the values at 0 and the location of the smallest value**
- **Repeat the first two steps for all the index beyond the first**
    - **Example:**
        - **Find the location of smallest value from 1 to the end of the list**
        - **Swap the values at 1 and the location of the smallest value**

You will get not credit if you use Arrays.sort().

Rubric (counts as 1 minor grade)

Points	Task
10	Randomly filled array
10	1st print
60	Sort
10	2nd print
10	Formatting and neatness

Maze

Create a program that allows the user to navigate a maze until they reach the end. The user will have the following options: move up, move down, move left and move Right.

Each time before a user moves the program will display the maze, marking the player's location with an 'X'. When a player tries to make an invalid move the program will print "Invalid move." When the player reaches the end of the maze the program will display how many moves it took him/her to solve it.

Maze Symbols:

- 'S' – Starting Point
- 'E' – Ending Point
- 'W' – Wall
- '-' – Path

The Following is an example of what a maze may look like:

```
W  W  W  W  W  -
S  -  -  -  W  -
W  -  W  -  W  -
W  -  W  -  -  -
E  -  W  -  W  -
W  W  W  W  W  -
```

Rubric (counts as 1 minor grade)

Points	Task
10	Loading the Maze
10	Display of the maze
10	"Invalid Move" when moving onto a wall
10	"Invalid Move" when moving outside the array bounds
20	End when player reaches E
20	Correct move count
20	Formatting and neatness

Worker Ant

Write a program that models an ant gathering food. The ant in our program will live in an ant farm.

The ground above the ant farm will be model with 3 character values.

- 'H' – Home
- 'F' – Ground with food
- ' ' – ground with no food

When the program starts the user will enter what the ground above the ant farm looks likes. **The user is expected to enter only 1 H.**

After data has been entered the program will tell the user how many steps it would take the ant to gather all the food and return home.

Ant gathering algorithm:

1. The ant will gather all the food on the left side of his home
   - When the ant reaches food it will pick it up and carry it back to the home
   - Each time food was successfully gathered from a non-0 location it will set back out going left in search of more food.
   - When the ant reaches spot zero it will gather any food there and head home.
2. The ant will gather all the food on the right of its home.
   a. When the ant reaches food it will pick it up and carry it back to the home
   b. Each time food was successfully gathered from a non-ending location it will set back out going right in search of more food.
   c. When the ant reaches the last location it will gather any food there and head home.

Rubric (counts as 2 minor grades)

Points	Task
10	Gets data
70	Correct number of steps
20	Formatting and neatness

Tic Tac Toe

Write a program that will allow two players to play a single game of Tic-Tac-Toe. The first player will play as ‘X’ and the second player will play as ‘O’. When the game ends the program will display the player that won or CAT.

Data will be stored in a 2D array of chars. If a player puts in a move that is already taken, they will be prompted for a valid move.

Extra Credit A

Modify the program so at the start of the program the user can select one player or two players. The computer will move randomly.

Extra Credit B

Make the computer block when there are 3 in a row.

Rubric (counts as 2 minor grades)

Points	Task
20	Only takes valid moves
20	Alternates turns
20	Displays the board in-between moves
20	Correct results when there is a winner
20	Correct on a CAT game
10	Extra Credit A
10	Extra Credit B

Matching Game

Write a program that will allow the user to play a simple matching game. The game board will be 4 by 4. The program will use a 4 by 4 two dimensional array of integers and a 4 by 4 boolean to track the game data.

The integer array will store pairs of single digit numbers and the boolean array will store true after items have been matched correctly.

Each turn the user will pick two different locations. When the values in the locations match, the boolean array at those locations will be changed to true.

When displaying the game board, locations that are marked true will have their numbers displayed.

Extra Credit

Modify the program to randomly generate the board, like my sample.

Rubric (counts as 2 minor grades)

Points	Task
10	Correctly loads the integer array with 8 pairs
10	Correctly loads the Boolean with false
30	Matches stay visible after matched
30	Game end when all pairs have been matched
20	Formatting and neatness