Lab 11

**Task 1**

This program will take in an answer key and student test answers. It will output the values of the student’s IDs and their test scores as well.

The program will output:

* Student IDs
* Student test scores

Algorithm:

1. Include the header files <fstream>, <iomanip>, and <cstring>
2. Declare the function prototype definitions.
3. Declare the variables.
4. Declare pointer variables.
5. Create an ifstream object and open the input file.
6. Check to confirm there is an input file with a conditional statement.
7. Declare two dynamic char arrays (c-strings), initializing them to their respective pointer variables.
8. Use the .get() function to retrieve the answer key and store it in the first dynamic array using the .get() function.
9. Output the answer key.
10. Prepare the output by configuring a proper header for information.
11. Retrieve the first student ID from the input file and use a char character to retrieve the empty white space between the ID and test answers.
12. Store the first set of test answers in the second dynamic array.
13. Use a while loop to run the program, making the conditional check whether or not we are still accessing the input file.
14. Set the len variable equivalent to the length of the answerString array.
15. Using a for loop, clear out any trailing unanswered questions in the answerString dynamic array.
16. Ensure to set the final element of the answerString dynamic array to ‘\0’.
17. Calculate the score of the student’s test by using nested conditional statements in a for loop.
18. The first statement checks whether or not each element is “ “. If it is, then the score (set equal to 0) is not changed.
19. If the answer is wrong, then subtract one from score and if it is right, add two to score.
20. After the score is determined, output the information, formatted under their respective headers in the output.
21. Call the function ExamGrade() to determine the letter grade of the student.

* ExamGrade() returns a char and takes in an int as a parameter.
* Use a switch statement to determine the letter grade based on the students score which is calculate by:

static\_cast<int>(score / 40.0 \* 10)

1. Read the next set of information from the input file and the while loop will continue.
2. Once we’ve finally exited the while loop, ensure to close the input file.
3. Deallocate the dynamic arrays from memory.

**Task 2:**

This program calculates the winner of an election based on specific user input.

The program will output:

* The total votes of the election
* Each candidate and how many votes they received
* The winner of the election

Algorithm

1. Include the header files <string> and <iomanip>
2. Declare the function prototype definitions.
3. Declare and initialize the variables.
4. Declare and initialize the pointer variables.
5. Set pointer variables equivalent to dynamic arrays.
6. Use a for loop to iterate through and initialize each value of the arrays by prompting the user for input.
7. Ensure to also add each vote count to the total vote count at the end of every iteration.
8. Configure output to show two decimal places and format nice header sections in the output.
9. Use a for loop to iterate through and output each candidate's name and how many votes they received.
10. Each iteration also calls the function CalculatePercentage().

* CalculatePercentage returns a float and takes in two int parameters.
* The float returned is calculated with static\_cast<float>(int1 / int2 \* 100)

1. After the for loop, output the total votes of the election.
2. Output the winner of the election by calling the function CalculateWinner() in the index of the candidates array.

* CalculateWinner() returns an int and takes in an int array and an int parameter.
* Declare a temporary index variable local to the function.
* Loop through the array and determine which index of the array has the highest value with a conditional statement.
* Set that index of that value equal to the temporary index variable local to the function.
* Return the temporary index variable to wherever the function was called.

1. Deallocate both of the dynamic arrays from memory.