This final is open book, open notes, open computer. You may not utilize any means of communicating with anyone else, via computer or otherwise.

**PLEASE DOUBLE CHECK YOU HAVE SUBMITTED ONLY THE .CPP FILE.**

Absolutely no points will be given if the .cpp file is not submitted. Please complete the programs below and upload to Moodle.

For the written section, please write your answers in the space provided

# Program: 40 Points

**Part 1 – Accept user input into an array - 15 Points**

Write a program that asks the user to enter up to 50 temperatures (assume the user is entering the temperatures in Celsius). Temperatures entered by the user should only be accepted if the value is between -273.15 and 1000 degrees C. The user can indicate they are finished by typing -1.

Sample Input / Output

Please enter the temperatures in Celsius.

Enter a Temperature (-1 to stop): **0**

User input

Enter a Temperature (-1 to stop): **10.2**

Enter a Temperature (-1 to stop): **25.7**

Enter a Temperature (-1 to stop): **-1**

**Part 2 – Write temperature conversion functions – 10 Points**

In Parts 3 you will need to print the value of each temperature the user entered, converted to both Fahrenheit and Kelvin. For this part, write **two** functions. Each function should accept as input a temperature in Celsius. One function (convertToK) should return the corresponding degrees Kelvin. The other function should return the corresponding degrees Fahrenheit (convertToF).

To convert from Celsius to Fahrenheit: F = (9/5)C + 32

To convert from Celsius to Kelvin: K = C + 273.15

**Part 3 – Write the converted temperatures to the console – 15 Points**

Print a table to the **screen** with each row containing a temperature (taken from the user) in Celsius, then Fahrenheit, and finally Kelvin (which should be obtained by calling the functions in part 2). Use output formatting to make the output appear identical to the table below.

C F K

-------------------------

0.00 32.00 273.15

10.20 50.36 283.35

25.70 78.26 298.85

**Part 2: Written Portion. Write your answers in the space provided.**

2) Which of the following statements correctly initializes an array to have the number 5 in all of its elements? **(4 points). Circle all that apply.**

a: int n[3] = 5;

b: int n = {5, 5, 5, 5};

c: int n[] = (5, 5, 5);

d: int n[5] = {5, 5, 5, 5, 5};

3) Suppose you have an array “n” that has already been filled with elements (and the variable count correctly holds the number of elements). Which of the following loops will **NOT** correctly print out each element in the array? **(4 points). Circle all that apply**

a: for ( int i = 1; i <= count; i++ ) {

cout << n[i-1] << endl;

}

b: for ( int i = 0; i < count; i++ ) {

cout << n[i] << endl;

}

c: for ( int i = 0; i <= count; i++) {

cout << n[i] << endl;

}

d: for ( int i = 0; i <= count-1; i++) {

cout << n[i] << endl;

}

4) Which of the following correctly print out each character in a c-string called str?  **(4 points) Circle all that apply**

a) for ( int i = 0; str[i] != ‘\0’; i++) {

cout << str[i];

}

b) int i = 0;

while (str[i] != ‘\0’) {

cout << str[i++];

}

c) int i = 0;

do{

cout << str[i++];

} while(str[i] != ‘\0’);

d) int i = 0;

while (str[++i] != ‘\0’)

cout << str[i];

}

**5)** Expressions: Evaluate **each** of the following expressions. (**10 Points)**

a) 4 + 2 > 5 % 4

- true

b) 5 < 6 || 6 > 4 || 4 > 5

- true

c) 8 - 3 \* 3 > 0

- false

d) (51 / 12 – 4) \* 2 == 0

- false

e) 5 < 9 && (true != false)

**-** true

**6)** Convert the following while loops to an equivalent for loop **(8 Points)**

a) int x = 7;

while ( x > 0 ) {

x--;

cout << x << endl;

}

////////////////////////

for (int i = 7; i > 0; i--) {

cout << i << endl;

}

////////////////////////

b) int x = 0;

int y = 10;

while ( x < y ) {

x+=3;

cout << x << endl;

}

///////////////////

for (int i = 3; i < 13; i += 3) {

cout << i << endl;

}

////////////////////////

**7)** Write a complete function named *compliment* that accepts a Boolean array, passed by reference, along with a count of how many Booleans are in the function. The function should change each true value in the array to false, and each false value in the array to true. There should be no return value. **(10 Points)**

/////////////////////////////////////////////////

bool compliment(&boolArray[], int count) {

for (int i = 0; i < count; i++) {

if (boolArray[i] == true) {

boolArray[i] = false;

}

else if (boolArray[i] == false) {

boolArray[i] = true;

}

}

}

////////////////////////////////////////////////////////////

8) Write a complete function that returns the minimum number of the three provided as input parameters. It should be used as follows: **(10 Points)**

int main() {

cout << findMin(4, 2, 6) << endl;

cout << findMin(9, 12, 11) << endl;

cout << findMin(5, 3, 2) << endl;

cout << findMin(4, 4, 4) << endl;

cout << findMin(3, 4, 4) << endl;

cout << findMin(6, 4, 6) << endl;

}

The output of this program would be:

2

9

2

4

3

4

////////////////////////////////

int findMin(int a, int b, int c) {

if (a >= b && b >= c) {

return c;

}

else if (a >= b && b <= c) {

return b;

}

else if (a <= b && b >= c) {

return a;

}

}

////////////////////////

**9)** Create a single input validation loop that requires the user to enter a single number divisible by 3, greater than 3, and less than 100**. For example, 27 would be acceptable (27/3 = 9, and it's greater than 3 and less than 100. 300 is unacceptable, 40 is unacceptable, -9 is unacceptable) (10 Points)**

////////////////////////////

int input;

bool valid = false;

do {

cin >> input;

if (input % 3 == 0) {

if (input > 3 && input < 100) {

return valid;

}

}

} while (!valid);

////////////////////////////