

Extra Credit Exam
Duration: **50 minutes**
Answer *all* questions.

1. Consider the fragment of code below. Trace through the code and answer the questions below. Show all working.

```
1.      int arr [5];  
2.      int k;  
3.  
4.      for (k = 0; k < 5; k = k + 1)  
5.          arr[k] = k + 1;  
6.  
7.      int i = 0;  
8.      int j = 4;  
9.      int temp = 0;  
10.  
11.     while (i <= j){  
12.         temp = arr[i];  
13.         arr[i] = arr[j];  
14.         arr[j] = temp;  
15.         i = i + 1;  
16.         j = j - 1;  
17.     }
```

- a) Using a trace table with headings *k* and *arr[k]*, show the contents of the array, *arr* after lines 1 - 5 have executed. [5]
- b) Using a trace table with headings *i*, *j*, *temp* and *arr*, show the *changed* contents of *arr* after lines 7 - 17 have executed. Use the values in *arr* from a). [5]

Total Marks: 10

2. a. Consider the sequence of numbers: -4 3 10 17 24 31 38 45. Write a fragment of code that uses a *while* loop to generate the numbers in the sequence, one at a time. As each number is generated, it is printed and a running sum is kept. At the end, print the sum of the numbers in the sequence. The numbers are not to be hard-coded. [5]
- b. Write a fragment of code which requests two weights in kilograms and grams and prints the sum of the weights. For example, if the weights are 3kg 500g and 4kg 700g your program should print 8kg 200g. [5]

Total Marks: 10

3. In a certain medical system, a diabetic patient takes his/her blood glucose measurements (integer values) once per day. The measurements are stored in a file *measurements.txt*. The amount of measurements is unknown beforehand but a negative integer is used to indicate the end of the data in the file. The lowest and highest valid measurements are 25 and 600 respectively.

Write a program which reads all the measurements from the file and finds and displays:

- The number of valid glucose measurements.
- The highest, lowest and average glucose measurement.
- The amount of times the patient's glucose measurement was below normal and the amount of times it was above normal. A normal measurement is between 80 and 140 inclusive.

[10]

Total Marks: 10

4. A leap year is a year which has 366 days. The extra day is the 29th of February. There is a leap year every four years.

Assume that a function *isLeapYear* exists. The function accepts a year as a parameter (an integer value) and returns *true* if the year is a leap year and *false* otherwise. You **do not** have to write this function.

- Using the *isLeapYear* function, write a function *daysInMonth* which accepts a year and a month as parameters (integer values) and returns the amount of days in the given month. If the month is invalid (e.g., 13), *daysInMonth* should return -1.

[5]

- Write a function *isValidDate* which accepts a year, a month and a day as parameters (positive integer values) representing a date and returns *true* if the date is valid and *false* otherwise. For example : 2018 5 12 represents a valid date (12th May 2018) but 2018 11 31 is not a valid date (31st November 2018)

You may find the following rhyme useful:

[5]

*Thirty days hath September, April, June, and November; all the rest have thirty-one,
except for February alone.*

Total Marks: 10

**End of Examination
(Total Marks: 40)**