VG101 — Introduction to Computer and Programming

Assignment 7

Manuel — UM-JI (Fall 2018)

- MATLAB: write each exercise in a different file
- C/C++: use the provided assignment template
- Include simple comments in the code
- If applicable, split the code over several functions
- Extensively test your code and impove it
- Write a single README file per assignment
- Archive the files (`*.{zip|tar}`) and upload on Canvas

*Exercises preceded by a * are mandatory. Any student not completing all of them, or submitting a work that cannot compile or be interpreted will automatically be deducted 1 mark on the final course grade.*

JOJ Online Judge

Exercises 1 to 4 can be tested on JOJ Online Judge.

Important reminders regarding the Online Judge (OJ):

- For each exercise save all the files, without any folder structure, into a `.tar` archive;
- Strictly stick to the input and output formats provided in the specifications;
- The OJ only checks the correctness of the code not its quality;
- For feedbacks on the quality, submit the code as part of the assignment and include the OJ score as well as the failed cases in the README file;

Ex. 1 — *Simple I/O in C++*

Write a short C++ program which prompts the user for the current year, the user's current age, and another year. The program should calculate the age that the user was or will be in the second year entered.

Specifications.

- Input: three lines, the current year on the first, the current age on the second, and the other year on the third;
- Output: one line showing the calculated age;

Ex. 2 — *Basic C++ programming*

Write a short C++ program which displays a menu for the user to choose between converting US dollars into Chinese RMB and Chinese RMB into US dollars. The user inputs the amount and the precision, then the program display the corresponding result.

Note: use 1.00 USD = 6.20350 CNY as the exchange rate

Specifications.

- Precision means "how many digit are expected in the final result", e.g. 123.456 has precision 6
- Input: three lines, the first one being 1 for USD or 2 for CNY, the amount on the second one, and the precision on the third one;
- Output: one line showing the result of the conversion;

* **Ex. 3 —** *File I/O in C++*

Rewrite exercise 4 from assignment 6 using C++ file I/O style.

Specifications.

- The binary, input, and output files are expected to be in the same directory
- Do not use absolute paths
- Do not prompt the user
- JOJ extra rules:
  - Add `#include "assignment.h"`
  - Do not include the `main()` function
  - The function prototype should be `void ex3()`

Partial sample output (ex. 3)

```
$ ./h7 -ex3
1 2 3
2 3 4
5 6 7

9 8 7
4 3 2
5 7 2
```

* **Ex. 4 —** *Basic programming*

Let the sequence $(u_n)_{n\geq 0}$ of integers be defined by

$$\begin{cases} u_0 = a \\ u_{i+1} = \begin{cases} \frac{1}{2}u_i, & \text{if } u_i \text{ is even} \\ 3u_i + 1, & \text{if } u_i \text{ is odd} \end{cases} \end{cases}$$

1. Write a function which prompts the user for a and determines N such that $u_N = 1$.
2. Write a function which prompts the user for a value M, and returns A, the value of $2 \leq a \leq M$, such that N is maximized.

Specifications

- Input: two lines, the first one being 1 or 2 for the function from question 1 or 2, respectively, and an integer corresponding to a or M on the second one;
- Output: one line showing the result;

* **Ex. 5 —** *From C to C++*

Write the C++ counterpart class of the following C structure and functions.

From C to C++

```c
#include <stdio.h>

static const char GRADES[] = {'F','F','F','F','F','F','D','C','B','A','A'};

typedef struct _Grade {
  char ltr;
  int prct;
} Grade;

void GradePrct(Grade *grade, int prct) {
  grade->prct = prct;
  grade->ltr = GRADES[prct / 10];
}

void GradeLtr(Grade *grade, char ltr) {
  grade->ltr = ltr;
  grade->prct = 100 - (ltr - 'A') * 10 - 5;
}

void printGrade(Grade *grade) {
  printf("Grade: %d -> %c\n", grade->prct, grade->ltr);
}

int main() {
  Grade g;
  int prct;

  printf("Input two space seprated grades (1st in %%, 2nd in letter): ");
  scanf("%d", &prct);
  scanf("\n");

  GradePrct(&g, prct);
  printGrade(&g);

  GradeLtr(&g, getchar());
  printGrade(&g);

  return 0;
}
```