FACULTY OF COMPUTERS AND INFROMATION, CAIRO UNIVERSITY

CS213: Programming II
Year 2022-2023
First Semester

Assignment 1 – Version 2.0

Course Instructors:
Dr. Mohammad El-Ramly

Revision History

Version 1.0	By Dr Mohammed El-Ramly	10 August 2018	Main Doc
Version 2.0	By Dr Mohammed El-Ramly	1 Oct. 2022 – Revised for 2022/23	

CS213: Object Oriented Programming
Assignment 1 (6 marks + 1.5 + 1.5 bonus) – Version 2.0

Cairo University, Faculty of Artificial Intelligence and Information

Objectives

This assignment reviews CS112 (Programming I) concepts and skills and trains the students on the concepts of class, object, abstraction and encapsulation.

Instructions

1. **These instructions must be followed to get the full marks. يجب اتباع هذه التعليمات بكل دقة**
2. **Deadline is Tues 18th of October 2022 @ 11:59 pm. Weight is 6 marks + 3 bonus** marks.
3. Students will forms teams of three students **from the same group** whose IDs **do not end with the same digit.** For example, 2017023, 20170433 and 20170124 cannot be in one team because two of them have IDs ending with 3. الفريق من ٣ طلاب لا ينتهى رقم بطاقة الكلية لهم بنفس الرقم
4. Please submit **only work that you did yourself**. If you copy work from your friend or book or the net **you will fail the course**. تسليم حلول منقولة من أى مصدر يؤدى إلى الرسوب فى هذا المقرر
لا تغش الحل أو تنقله من أى مصدر و اسألنى فى أى شئ لا تفهمه لكن لا تنقل الحلول من النت أو من زملائك أو أى مكان

Task 0 (0 marks)

1. Review C++ syntax and refresh your C++ knowledge.
2. Review CS112 Programming I and its materials at http://www.acadox.com/class/64082
3. Read this quick tutorial https://www.codesdope.com/cpp-introduction/
4. Create a **private GitHub** repo for the project and **use it for development.**

Task 1 (3 marks) - Classes, objects, abstraction and operator overloading.

Students should divide the work as suggested below and then should integrate their code together and make sure it works properly.

Different variations of types **`int`** and **`float`** exist in C++ and other languages. They are = limited by minimum and maximum values depending on the number of bytes used to store the number. We need versions of these types with unlimited bounds. Java solves this problem by providing **`BigInteger`** and **`BigDecimal`** classes. In this problem it is required to develop a new C++ type (class) that can hold unlimited decimal integer values and performs arithmetic operations on them. You will develop in C++ a class, **`BigDecimalInt`** that supports writing statements with very long integer values like these (Your program must enable this code to run):

```
BigDecimalInt num1("12345678901234567890123456789O");
BigDecimalInt num2("+113456789011345678901134567890");
BigDecimalInt num3("-200000000000000000000000000000");
BigDecimalInt num4 = num2 + num1;
BigDecimalInt num5 = num2 - num1;
cout << "num1 = " << num1 << endl;
cout << "num2 = " << num2 << endl;
//Next statement will print 236913578023691357802369135780
  cout << "num2 + num1 = " << num4 << endl;
//Next statement will print -313456789011345678901134567890
cout << "num2 - num1 = " << num5 << endl;
```

CS213: Object Oriented Programming
Assignment 1 (6 marks + 1.5 + 1.5 bonus) – Version 2.0

Cairo University, Faculty of Artificial Intelligence and Information

1. Design the class **`BigDecimalInt`** that has the following public interface (set of operations available to use by developers using the class):

```
a. BigDecimalInt (string decStr);  // Initializes from string & rejects bad input
b. BigDecimalInt (int decInt);     // Initialize from integer
c. BigDecimalInt operator+ (BigDecimalInt anotherDec);  // member fn
d. BigDecimalInt operator- (BigDecimalInt anotherDec);  // member fn
e. bool operator< (BigDecimalInt anotherDec);           // member fn
f. bool operator> (BigDecimalInt anotherDec);           // member fn
g. bool operator==(BigDecimalInt anotherDec);           // member fn
h. BigDecimalInt operator= (BigDecimalInt anotherDec);  // member fn
i. int size();                                          // member fn
j. int sign();                                          // member fn
```

   You will also need to overwrite the **<<** operator as follows: (as friend or external function)

```
k. friend ostream& operator << (ostream& out, BigDecimalInt b)
```

2. Please note the following
   - Using data encapsulation. Decide on a suitable container for storing the content of the number. Study the different options and find the ***efficient*** and ***elegant*** one to use in terms of (1) memory used and (2) suitable attributes and (3) speed of developing the program. Possible options include string, dynamic array, vector, or other containers in STL. These are details that are not important to the user of your class. This is called ***encapsulation* or *data hiding*.** You will need to build + and – operations that work on the representation you chose. The user does not care about which algorithm you use as long as it works. This is called ***algorithm hiding.***
   - Signs might be added to the number at initialization, but only –ve sign is printed. You may like keep this information about the sign in a separate attribute.
     - +11111111111111111100000000000000000000011111111111 // valid
     - -00000000000000999999999999999911111111111111111999999 // valid
     - + 00099999 // invalid
     - +-999000000 // invalid
     - 99900011111 // valid
   - Both addition and subtraction should consider +v end –ve cases.
   - Implement the class **`BigDecimalInt`** and write 10 test cases (including –ve numbers) to test it and the member functions you included in it. Implement a program that runs the test cases and verifies the result.
   - Student with smallest ID **will implement items: a, b and c.** The one with next ID **will implement item d and e.** The one with the next ID **will implement items f to k**.
   - **See this https://www.geeksforgeeks.org/copy-constructor-vs-assignment-operator-in-c/**
   - **See https://ecomputernotes.com/cpp/classes-in-c/returning-object-from-function**

CS213: Object Oriented Programming
Assignment 1 (6 marks + 1.5 + 1.5 bonus) – Version 2.0

Cairo University, Faculty of Artificial Intelligence and Information

- Team should help each other and support in each.
- Team should integrate the work of Task 1 together and submit one working program.
- Team should test the entire program and make sure that it works correctly in full.
- All team members **must fully understand all parts of the program.**
- **Use separate compilation**

3. **Deliver**:
   a. **Write code should be in standard C++ not using third-party libraries.**
   b. Name your file **A1_P1_YourGroup_YourIDs.cpp** or **.zip** (if more than one file)
   c. In the report, write who did which part.

Task 2 (3 marks) – Individual problems from Sheet 1

Take the smallest ID % 11. Get 20210398 % 11 = 10. Then this team member will solve problems 10, 13%11 = 2 and 16%11 = 5. Next ID will solve problems 11, 3 and 6 and next one will solve 0, 4 and 7. These problems are in **Sheet 1 under Acadox.**

2- **Deliver:**

- Write code should be in **standard C++** not using third-party libraries.
- Name your file A1_SheetPb**XX**_YourID.cpp or .zip (if more than one file) (XX is pb num)
- In the report, write who did which problems.

Group Bonus 1: Task 3 (1.5 mark) – Starting a Software Business

Software is created to solve real world problems and help people better achieve their tasks. This can be a mobile app, a Web site, a desktop application, an embedded system, etc. For example,

1- Skype, Whatsapp, Facebook Messenger, line, etc., help people communicate easily.
2- Google Chrome, Internet Explorer, Firefox, etc., help people access web pages on a web server.
3- Gym management systems helps gym owners manage membership and gym services
4- Steps tracking mobile apps help you know how many steps you walked and stay healthy.
5- Auto cruising software help car driver drive the car automatically with minimal manual actions.
6- Google translate and similar software help people understand text in a different language.

In this task, develop an idea for a software system based on a real need for a real client. You will collect and document the requirement specifications of this software in details. To do this task:

1- Identify a real possible client (or a target group of people) in your personal circle; someone whose work or life will be better and who would be faster software helps him or her.
2- **The group will meet** the client at least twice and will **take a photo of the meeting**.
3- The group will ask the client in details about what he does (they do) and how a software system can help him or her be more efficient.

CS213: Object Oriented Programming
Assignment 1 (6 marks + 1.5 + 1.5 bonus) – Version 2.0

Cairo University, Faculty of Artificial Intelligence and Information

Deliver:

1- The group will write in ع or E (1) **Overview of the required software** (2) a **list of requirements** (all things that software should do) and (3) will draw the **main screen(s)** of the app using Word or pen or using programs like: https://balsamiq.com/ (Or search for Mockup apps for Android)
2- The group will submit one report of three pages at least, font 12, single space with their names and IDs, the project idea or title and the specifications of the project as described in 4 and photos with client.

١- ابحث فى دائرة معارفك عن عميل محتمل يمكن أن يتحسن عملهم بواسطة برنامج على الحاسب (أو مجموعة من الناس مثلا الشباب أو المسنين أو هواة الرياضة أو الراغبين فى التخسيس أو طلاب العلوم الإسلامية أو المحامين أو الباحثين عن عمل إلخ و يمكن أن يفيدهم عمل برنامج خاص لهم)
٢- ستلتقى مجموعتك بالعميل المحتمل (أو المجموعة) مرتين على الأقل و يجب **أخذ صورة لهذه اللقاءات.**
٣- مجموعتك ستسأل العميل عما يفعل و كيف يمكن للبرنامج المحتمل تسهيل مهمته و عمله أو تيسير حياته أو ملء حاجة لديه
٤- ستكتب المجموعة بأى لغة (١) مقدمة عامة عن البرنامج المطلوب ووظيفته و (٢) قائمة تفصيلية مطولة بالوظائف المطلوبة فى هذا البرنامج و ما ينبغى أن يعمله و (٣) عدة رسومات للشاشات الرئيسية للبرنامج مرسومة باليد أو أى برنامج مثل https://balsamiq.com/ أو غيره.
٥- ستقدم المجموعة تقريرا من ٣ صفحات على الاقل بصيغة بى دى أف PDF مكتوبة بخط مقاس ١٢ و مسافة واحدة بين السطور و عليه اسماء الطلاب و اسم المشروع و التفاصيل المذكورة فى ٤ و صور لقاء العميل.

Individual Bonus 2: Task 4 (1 mark) – Using AI for Programming

Machine learning, combined with the huge size of data available online and the huge processing power that big companies own, created a chance to build powerful intelligent models for innovating text, images and even code. If you want to see the power of AI for innovation, try these two sites.

https://thispersondoesnotexist.com/ (very time you refresh, it generates a new image of a person that does not exist. Image is generated from learning that the model made.)

https://app.inferkit.com/demo (Write any useful text, e.g. try this "Your name is a father / mother of three children ". Keep the cursor on the same line, tick **Start at beginning** box) and then click **Generate Text** a few times and it will write a new story for you.

AI-based systems are also expanding in the area of programming, beyond imagination. Examples include:

https://openai.com/blog/openai-codex (OpenAI tool for writing code by describing what it does)

https://github.com/features/copilot (GitHub AI-based pair programmer)

https://alphacode.deepmind.com/ (DeepMind AI tool for competitive problem solving)

In this task, you will:

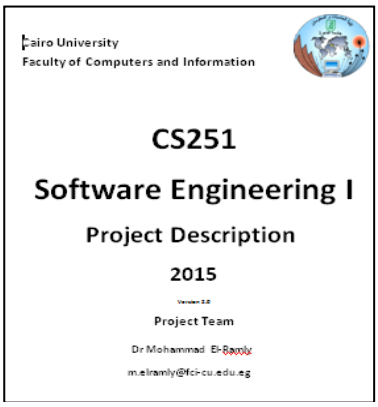
1- Implement one of the problems you solved in this assignment using one of the AI-based program development tools. You can try Codex or Copilot (register with university email).
2- For Copilot, join GitHub Copilot with your student account for free (for non-students, it is paid)
3- Install and use Copilot.
4- Develop the solution and **record a short video of your experience.**

CS213: Object Oriented Programming
Assignment 1 (6 marks + 1.5 + 1.5 bonus) – Version 2.0

Cairo University, Faculty of Artificial Intelligence and Information

5- If you use Codex, do the same steps.
6- **Deliver:**
   - The **code** generated with AI assistance with a **readme** file explaining what you did
   - Explain your experience and **(1) if it worked well and (2) your evaluation for this tech.**
   - The link to the **video** explaining the experience you had. **Come show the prof the results.**

Submission Instructions


Cairo University
Faculty of Computers and Information
CS251
Software Engineering I
Project Description
2015
Project Team
Dr Mohammad El-Ramly
m.elramly@fci-cu.edu.eg

Team will submit into acadox the following:

1. A zip file with the following components.
2. A pdf report with the following items.
   - The document should have a cover page like this.
   - The source code of each program in a separate folder.
   - A screen shot for every GitHub account.
3. Team will create a project in **GitHub** to upload code there.
4. Each team member will work individually on his part. **But the team must provide ONE integrated and working program and report.**
5. Team members are expected to help each other but not do work of others.
6. **All team members must understand the details** of all programs and be able to explain it or even modify it if needed.
7. Team members are responsible of testing all the programs and making sure they work.
8. TA can ask any team member about any of the programs developed and its code**.**
9. **Ask your TA** about the discussion time of your work.

Marking Criterion

1. 2.0 for developing your part in the group problem (Task 1) correctly.
2. 1.0 for integrating the code and developing one working program.
3. 3.0 for developing a solution for the individual problems (Task 2).
4. -1 for not understanding the solutions of individual problems of others
5. 1.5 **Group Bonus** for collecting high quality information about the app and the report
6. 1.5 **Individual Bonus** for successfully using the tools and making a video / report

Marking Instructions for TA

4. Code that does not run does not exist.
5. Be merciful, kind and polite with the kids.
6. Appreciate their effort and use marking as learning opportunity to educate them.
7. Ask every student in all code, even not only on what s/he did.
8. Start by asking him or her: What did you do? Then delete one function from his code and ask him to write it again. Ask him or her do you fully understand the code of you team members, and if he does, delete a simple function and ask him to write it.
9. Be generous in marks.
10. Always smile even if you are frustrated.