Start now and do not wait ابدأ الآن و لا تنتظر فلو تأخرت لن تجد وقتا كافيا

FACULTY OF COMPUTERS AND ARTIFICIAL INTELLIGENCE, CAIRO UNIVERSITY

CS112: Structured Programming
Winter 2021 - 2022

Second Semester

Assignment 1 – Version 1.0

Course Instructor:

Dr. Mohammed El-Ramly

Revision History

Version 1.0	By Dr Mohammed El-Ramly 20 Feb. 2022	Main Doc

CS111: Fundamentals of CS
Assignment 1 (7 marks + 3 bonus) – Version 1.0

Cairo University, Faculty of Computers and Artificial Intelligence

Objectives

This assignment trains students on speed typing, algorithms, flowcharts. program development, using libraries and teamwork.

Instructions

1. **It is very important to collect course work marks in order to learn well and get a good grade. من المهم للغاية حسن أداء أعمال السنة أولا لتتعلم جيدا و ثانيا لتنجح بسهولة و تحصل على تقدير مرتفع**
2. **These instructions must be followed to get the full marks. يجب اتباع هذه التعليمات بكل دقة**
3. **Deadline is Saturday 5 of March 2022 @ 11:59 pm.**
4. Weight of the assignment is **7 marks** + you can earn up to **3 bonus** marks.
5. Students will forms teams of three students **from the same group** whose IDs **do not end with the same digit.** For example, 2021023, 20210433 and 20210124 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
6. 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 تعالى و اسألنى فى أى شئ لا تفهمه
7. **Team will submit into acadox (in the proper task with correct TA name) the following:**
   - A zip file with a pdf document with their names and IDs and list of games and the application they made and the algorithms or flowcharts, screenshots and **video link** of typing club and typing plan for each student.
   - The report should have a cover page similar to this one.
   - The source code of the games and the application in separate folders with proper names. Same for bonus.
   - The algorithm or flowchart of each game or application that explains how it works.

   Cairo University
   Faculty of Computers and Artificial Intelligence

   CS112 – Structured Programming
   Second Semester 2021-2022
   Group XXX
   TA: Eng .....
   Assignment #1 Report
   First name, ID, group, email, mobile
   Second name, ID, group, email
   Third name, ID, group, email
8. **Team will submit in paper to the TA following:**
   - A printed version of the pdf document with (1) cover page (2) team names, IDs and list of games and application, (3) flowcharts or algorithms of each game and the application, (4) screen shots of tying club and (5) **link to video**.

Task 1 (1 mark) – Individual Task

1. This is to be done by each individual student.
2. Create an account on https://www.typingclub.com and login.
3. Do the **first 100 lessons** in the screen below and practice for at least 3 hours.
4. Make a plan of how you will increase your typing speed and what level you plan to reach (how many characters per minute)
5. If your level in typing is high, start from the max lesson you know and do **100 lessons** after that.

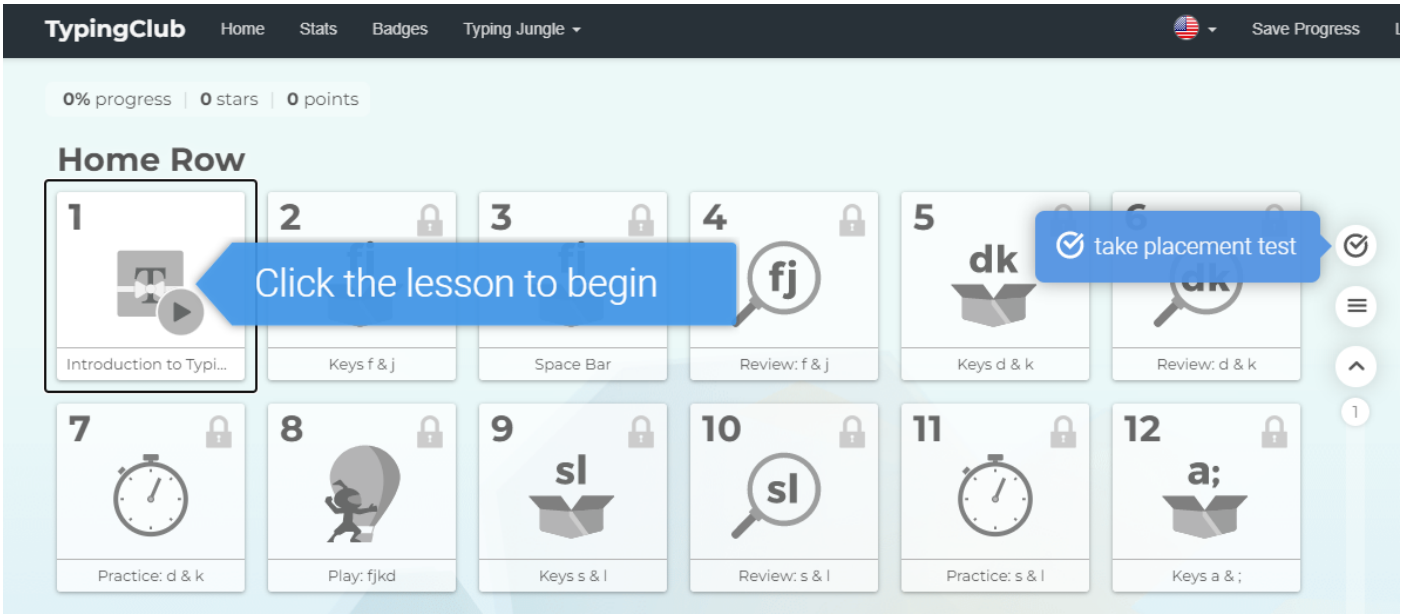
CS111: Fundamentals of CS

Assignment 1 (7 marks + 3 bonus) – Version 1.0

Cairo University, Faculty of Computers and Artificial Intelligence

What to deliver in the report?

1. Print screens that show your account name and statistics of how long you used the program.
2. **Your target level for speed typing and your plan to reach it.**



Task 2 (2.5 marks) – Individual Task

Each team member will solve one of these problems, by designing (algorithm or flowchart) and programming in Python a two-player game according to the last number of his ID number. For example if his ID number is 20170080, then he solves problem **0** and if his number is 20170236, he solves problem **6**. If the student solves the **wrong game,** he gets **ZERO**.

Divide the program into functions and write a main function that calls and executes the other functions. Review the X-O and Nim games developed in CS111 and available in acadox.

Remember that the generic sequence of board / number games is as follows:

1- Decide on data structure(s) to store game status

2- Display initial game status

3- While (True)

   a. First player plays
   b. Update game status (internally and displayed to users)
   c. If someone wins, declare winner and end game
   d. else if game is over, declare draw and end game

   e. Second player plays
   f. Update game status (internally and displayed to users)
   g. If someone wins, declare winner and end game
   h. else if game is over, declare draw and end game

CS111: Fundamentals of CS

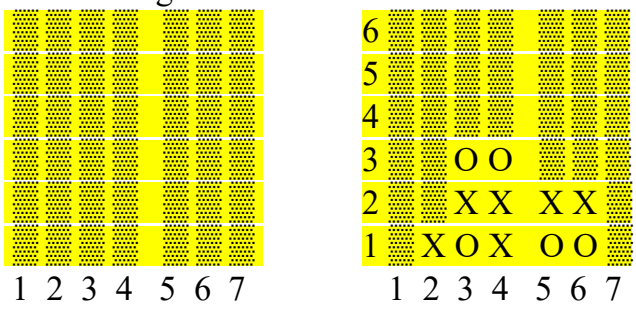
Assignment 1 (7 marks + 3 bonus) – Version 1.0

Cairo University, Faculty of Computers and Artificial Intelligence

List of Games

0. **SOS** game. Before play begins, a square grid of at 4x4 or 5x5 squares in size is drawn. Players take turns to add either an "S" or an "O" to any square, with no requirement to use the same letter each turn. The object of the game is for each player to attempt to create the straight sequence S-O-S among connected squares (either diagonally, horizontally, or vertically), and to create as many such sequences as they can. If a player succeeds in creating an SOS, that player immediately takes another turn, and continues to do so until no SOS can be created on their turn. Otherwise turns alternate between players after each move. https://en.wikipedia.org/wiki/SOS_(game)

1. **Connect- 4 game**. A board of 7 columns x 6 rows is displayed that is held up in physical game as in figure below with row 1 is the bottom row:



   Players choose a symbol, either X or O. In their turn, they drop the symbol from top of the board (number 6) until it settles in the bottom empty cell. See video: https://www.youtube.com/watch?v=ylZBRUJi3UQ The first player to connect 4 symbols horizontally, vertically or diagonally wins.

2. **Tic-Tac-Toe with numbers**. A board of 3 x 3 is displayed and player 1 takes odd numbers 1, 3, 5, 7, 9 and player 2 takes even numbers 0, 2, 4, 6, 8. Players take turns to write their numbers. Odd numbers start. Use each number only once. The first person to complete a line that adds up to 15 is the winner. The line can have both odd and even numbers.

3. **Number scrabble** is played with the list of numbers between 1 and 9. Each player takes turns picking a number from the list. Once a number has been picked, it cannot be picked again. If a player has picked three numbers that add up to 15, that player wins the game. However, if all the numbers are used and no player gets exactly 15, the game is a draw.

4. **Two squares game.** This game is played on a board of 4 x 4 squares. Two players take turns; each of them takes turn to place a rectangle (that covers two squares) on the board, covering any 2 squares. Only one rectangle can be on a square. A square cannot be covered twice. The last player who can place a card on the board is the winner.

   Example:

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Player 1: 15, 16

X	2	3	4
X	6	7	8
9	10	11	12
13	14	15	16

Player 1: 1, 5

Player 2: 8, 12

X	2	3	4
X	6	X	8
9	10	X	12
13	14	15	16

Player 2: 7, 11

Player 1: 13, 14

X	2	3	4
X	X	X	8
9	X	X	12
13	14	15	16

Player 1: 6, 10

Player 1 wins the game

X	2	X	X
X	X	X	8
9	X	X	12
13	14	15	16

Player 2: 3, 4

CS111: Fundamentals of CS

Assignment 1 (7 marks + 3 bonus) – Version 1.0

Cairo University, Faculty of Computers and Artificial Intelligence

X	2	X	X
X	X	X	8
9	X	X	12
13	14	X	X

X	2	X	X
X	X	X	X
9	X	X	X
13	14	X	X

X	2	X	X
X	X	X	X
9	X	X	X
X	X	X	X

5. **One line memory game.** In this game, 10 characters are chosen. Each character is repeated twice. The characters are put in random order. Each player picks two numbers between 1 and 20. The two characters in these positions are shown and the rest are covered with their position number. If the two characters match, the player wins a point and these two characters are covered with * from now on. The **screen is cleared** and the remaining character positions are displayed for the next player. When all characters are covered with *, the game ends and the player with the biggest score wins.
   Example:
   Welcome: 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0
   Player#1 – score 0: 11, 13
   Welcome: 1 2 3 4 5 6 7 8 9 0 A 2 C 4 5 6 7 8 9 0
   Screen is cleared.

   Welcome: 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0
   Player#2– score 0: 6, 11
   Welcome: 1 2 3 4 5 A 7 8 9 0 A 2 3 4 5 6 7 8 9 0
   Screen is cleared.

   Welcome: 1 2 3 4 5 * 7 8 9 0 * 2 3 4 5 6 7 8 9 0
   Player#1– score 0: 1, 8
   Welcome: C 2 3 4 5 * 7 B 9 0 * 2 3 4 5 6 7 8 9 0
   Screen is cleared.

   Welcome: 1 2 3 4 5 * 7 8 9 0 * 2 3 4 5 6 7 8 9 0
   Player#2– score 1: 1, 13
   Welcome: C 2 3 4 5 * 7 8 9 0 * 2 C 4 5 6 7 8 9 0
   Screen is cleared.

   Welcome: * 2 3 4 5 * 7 8 9 0 * 2 * 4 5 6 7 8 9 0
   Player#1– score 0: 2, 16
   Welcome: * F 3 4 5 * 7 8 9 0 * 2 * 4 5 F 7 8 9 0
   Screen is cleared.

   Welcome: * * 3 4 5 * 7 8 9 0 * 2 * 4 5 * 7 8 9 0
   Player#2– score 2: 19, 20
   Welcome: * * 3 4 5 * 7 8 9 0 * 2 * 4 5 * 7 8 D E
   Screen is cleared.
   ……. *Repeat until one of them wins*

6. **100 game.** Two players start from 0 and alternatively add a number from 1 to 10 to the sum. The player who reaches 100 wins.
7. **Subtract a square.** This is a two-player mathematical game of strategy. It is played by two people with a pile of coins (or other tokens) between them. The players take turns removing

coins from the pile, always removing a non-zero square number of coins (1, 4, 9, 16, …). The player who removes the last coin wins.
Example of this game is at: http://delphiforfun.org/Programs/SubtractingSquares.htm

8. **Fibonacci Nim** is played by two players, who alternate removing coins or other counters from a pile of coins. On the first move, a player is not allowed to take all of the coins, and on each subsequent move, the number of coins removed can be any number that is at most twice the previous move. The player who takes the last coin wins. The pile of coins to starts the game with is chosen arbitrarily.

9. **Kayles game**. This game begins with an arbitrary number of tokens in a single row. Two players alternate turns. During a turn a player may remove either one or two adjacent tokens. Note that if tokens are removed from the middle of the row, then their removal creates a gap that divides the row to two parts. The player who removes the last token wins.
Example:
`1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0`
Player 1: 2, 3
`1 - - 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0`
Player 2: 17, 18
`1 - - 4 5 6 7 8 9 0 1 2 3 4 5 6 - - 9 0`
Player 1: 5
`1 - - 4 - 6 7 8 9 0 1 2 3 4 5 6 - - 9 0`
Player 2: 8, 9
`1 - - 4 - 6 7 - - 0 1 2 3 4 5 6 - - 9 0`
Player 1: 12, 13
`1 - - 4 - 6 7 - - 0 1 - - 4 5 6 - - 9 0`
Player 2: 15
`1 - - 4 - 6 7 - - 0 1 - - 4 - 6 - - 9 0`
….. till one players takes the last piece and wins.

Task 3 (2.5 marks) – Group Task

In this task, the team will work together to develop a small Python application that is similar to real world applications that do the same task.

Task done will depend on the sum of team IDs % 2. For example, if students with IDs 20210234 + 20210318 + 20210400 are in one team, then sum % 2 = 60630952 % 2 = 0. They do App 0.

Application 0 – Population Application with `openpyxl`

In this application, you will use **`openpyxl`** https://zetcode.com/python/openpyxl/ to create an application to show statistics about different countries. The statistics about each country will be in a file named after this country as show in the following examples, with each row having two values: the name of a state / province, governance and its population. It is possible also to store each country in a sheet inside the same Excel workbook.

CS111: Fundamentals of CS

Assignment 1 (7 marks + 3 bonus) – Version 1.0

Cairo University, Faculty of Computers and Artificial Intelligence

USA.xlsx

	A	B
1	Alabama	5,024,279
2	Alaska	733,391
3	Arizona	7,151,502
4	….	…..
5	….	…..

Canada.xlsx

	A	B
1	Ontario	14,677,900
2	Quebec	8,494,500
3	BC	5,103,500
4	…..	…..
5	…..	…..

Egypt.xlsx

	A	B
1	Cairo	10,030,000
2	Giza	9,200,000
3	Sharkia	7,640,000
4	…..	…..
5	…..	…..

The application gives the user a menu of choices:

1- Enter a new country to load its file (or display a message if not available)
2- Display the population of each state / province / governorate and total population of the country
3- Display the state / province / governorate with the highest population and the one with the lowest population.
4- Exit

Application 1 – PDF Separate and Merge Application with `PyPDF2`

In this application, you will use **`PyPDF2`** https://zetcode.com/python/openpyxl/ to create an application to split and merge pdf files. The application will allow the user to merge two pdf files together and create a third file or extract individual pages from a pdf file.

It will allow give the user these choices:

1- Merge two files # Takes names of two valid pdf files and name or merged file
2- Extract a page from file # Takes filename and page num X & outputs filename-X.pdf
3- Split file into separate pages # Takes filename and outputs filename-1.pdf, filename-2.pdf, …
4- Exit

Task 4 (1 mark) – Collaboration, code quality

1. While every team member will develop his game individually, team members are expected to help each other and explain difficult concepts to each other.
2. All team members must understand the details of the games of other team members and be able to explain it or even modify it if needed.
3. Code must have enough comments and be well documented.
4. **Team members are responsible of testing the game of each one and making sure it works correctly.**
5. **Team will produce a video explaining how ALL their programs work. Video will have FCAI logo and student nameas at the start and will be uploaded in Youtube or a Google Drive and link will be added to report.**
6. TA can ask any team member about any of the games developed and its code. **1 mark** is given for good answers and for video. Failure to show that they understand other games, will give **0.**

CS111: Fundamentals of CS
Assignment 1 (7 marks + 3 bonus) – Version 1.0

Cairo University, Faculty of Computers and Artificial Intelligence

Bonus

Student / team can choose from this list of bonus tasks (individual and group) and **he can get a max of 3 bonus marks.**

Bonus1: Task 5 (1 mark) – Individual - C++ Version of Your Game

1- Create a C++ version of your game that works correctly exactly like the Python version.
2- You must follow good C++ coding style.

Bonus2: Task 6 (1 mark) – Group - Graphical User Interface

In this task, team creates a GUI for the application of Task 3 using a suitable library. Create menus, text boxes, lists, etc. as needed in order to make available and run the functions

Bonus4: Task 7 (2 marks) – Individual - Graphics Game

Each team member can do one extra game from the list above **with graphics** according to **the second least digit in** his ID. If his ID is 2021034**5**, he does game number 4 and if his ID is 2021039**5**, he does game number 9, etc. There is a library to help develop games with Python, called https://www.pygame.org/. There is a book about using it to make games https://inventwithpython.com/pygame/. **ZERO** mark if you do the wrong game.

Marking Criterion

1. 1 (individual) for evidence of finishing 100 classes in TypingClub.com
2. 2 (individual) for finishing the required game and it works properly
3. 0.5 (individual) for documenting the game algorithms and how it works in the report
4. 2 (group) for finishing the required application and it works properly
5. 0.5 (group) for documenting the application algorithms and how it works in the report
6. 0.5 (group) for evidence of group work and understanding cod of others
7. 0.5 (group) for **working video** explaining all programs done and has names and FCI logo

Bonus

8. 1 (individual) for finishing the required game in C++ and it works without errors.
9. 1 (group) for making a nice GUI for the application.
10. 2 (individual) for building a working graphics game according to the second rightmost digit in ID