CSC 340.03

Kayvaun Khoshkhou 920357344

Assignment 1

Due 09/01/2020 11:55 PM

Part A: Communication

1. iLearn

Screenshot of asmt 1 discussion forum response:

2. Emailing
    a. I confirm that I will use my SFSU email address (@mail.sfsu.edu) when contacting my grader and my course instructor so that my emails will not be filtered.
    b. When contacting my instructor I will start the email subject with format: "CSC340.04 TAC" so that I will get an answer timely.
    c. If I do not get an answer within 24 hours, I will check if I sent the email properly. In either case I will kindly resend my email message.

3. Grader
    a. https://ilearn.sfsu.edu/ay2021/mod/forum/view.php?id=122349
    b. Peijun Huang
    c. Phuang5@mail.sfsu.edu

4. Guidelines for All Assignments and Assignment Report Template
    a. YES. This is to confirm that I have carefully read, understood, and agreed to the Guidelines for ALL Assignments above and the Assignment Report Template. I will strictly follow the instructions.
5. YES. This is to confirm that I have carefully read, understood, and agreed to the Course Policy on Student Conduct and Academic Honesty which was distributed to me with the course syllabus and whose digital copy was shared with me on the File Manager. I am acutely aware that the policy includes, but is not limited to, the San Francisco State University's Code of Student Conduct (at https://conduct.sfsu.edu/standards), the Computer Science Department's Student Policies (at https://cs.sfsu.edu/student-policies), and the Honor Code of this course (at http://csc220.ducta.net/00-README-StudentConduct_AcademicHonesty.pdf). I will strictly follow all the rules.

Part B: File Manager

Screenshot for transferable knowledge:

CSC340/JAVA-TransferableKnowledge:

IDX	FILE NAME	TYPE	SIZE	LAST MODIFIED	PERMS
1	00-README.txt	File	0.00 MB	02:18 08-14-20	604
2	Assertion.pdf	File	0.29 MB	02:18 08-14-20	604
3	Big-O.pdf	File	0.83 MB	02:18 08-14-20	604
4	BigO-GrowthRates.xlsx	File	0.76 MB	02:18 08-14-20	604
5	CSC220_DataStructures/	DIR	0.00 MB	02:18 08-14-20	705
6	ClassDesignGuidelines.pdf	File	0.27 MB	02:18 08-14-20	604
7	CommandLine-NetBeans.pdf	File	0.43 MB	02:18 08-14-20	604
8	Compare.pdf	File	0.87 MB	02:18 08-14-20	604
9	Exceptions.pdf	File	0.45 MB	02:00 08-14-20	604
10	Generics.pdf	File	0.58 MB	02:00 08-14-20	604
11	GoogleGuava.pdf	File	0.71 MB	02:00 08-14-20	604
12	Hashing.pdf	File	0.53 MB	02:00 08-14-20	604
13	ProgrammingTipsCSC220.pdf	File	0.46 MB	02:00 08-14-20	604
14	Shallow_and_Deep_Copies.pdf	File	0.66 MB	02:00 08-14-20	604
15	SortingSelectionInsertionShell.pdf	File	1.03 MB	02:00 08-14-20	604
16	enum.pdf	File	0.36 MB	02:00 08-14-20	604
17	main.pdf	File	0.30 MB	02:00 08-14-20	604

CSC340 iLearn: Announcements and Forums

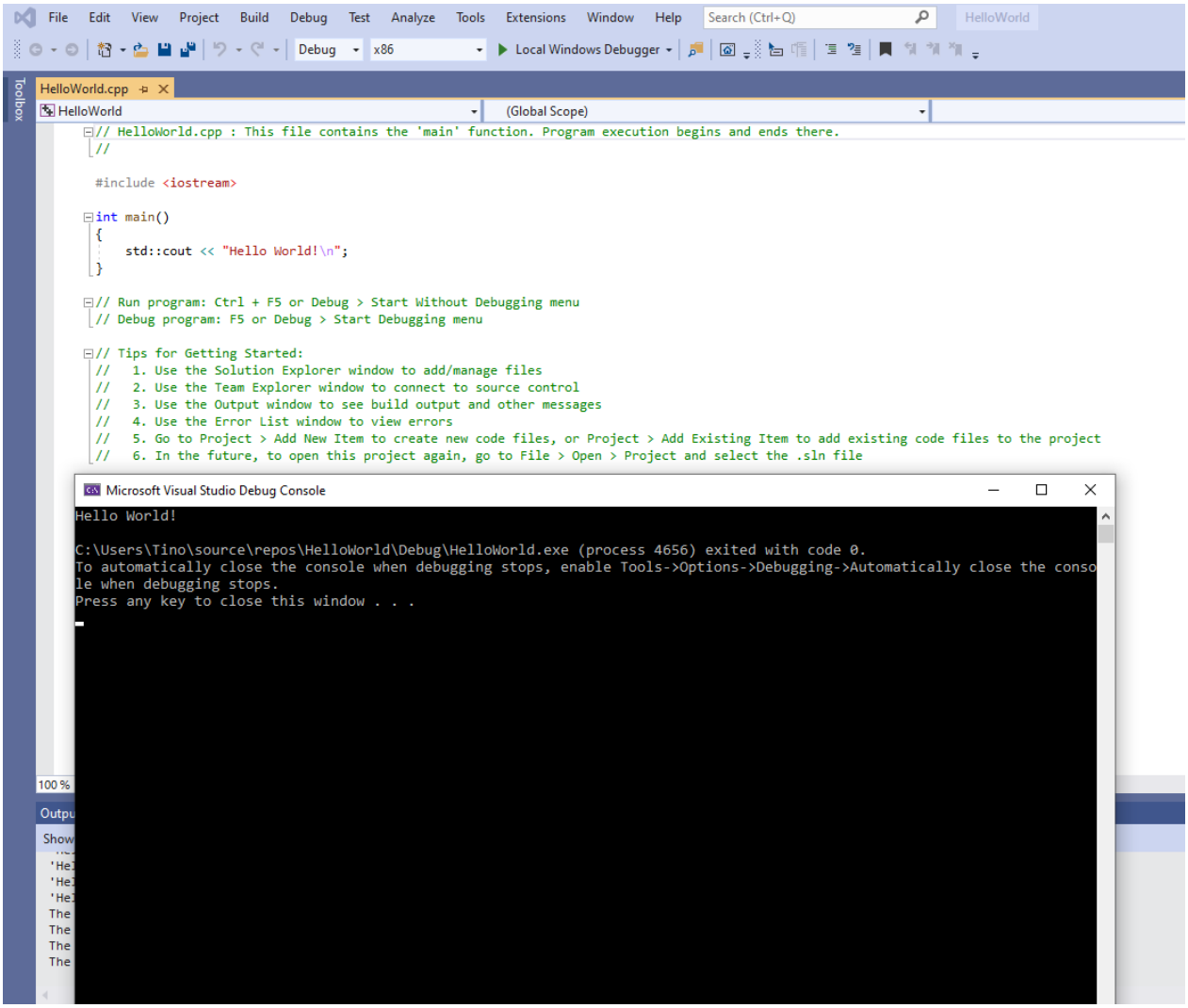
Screenshot for CPP Reference:

CSC340/CPP-Reference:

IDX	FILE NAME	TYPE	SIZE	LAST MODIFIED	PERMS
1	00-README.txt	File	0.00 MB	02:00 08-14-20	604
2	01-WelcomePrograms/	DIR	0.00 MB	02:00 08-14-20	705
3	02-SimplePrograms/	DIR	0.00 MB	02:00 08-14-20	705
4	03-Selections/	DIR	0.00 MB	02:00 08-14-20	705
5	04-MathCharsStrings/	DIR	0.00 MB	01:59 08-14-20	705
6	05-Loops/	DIR	0.00 MB	01:59 08-14-20	705
7	06-Functions/	DIR	0.00 MB	01:59 08-14-20	705
8	07-1DArrayCStrings/	DIR	0.00 MB	01:59 08-14-20	705
9	08-MultiDArray/	DIR	0.00 MB	01:59 08-14-20	705
10	09-ObjectsClasses/	DIR	0.00 MB	01:59 08-14-20	705
11	10-OOPThinking/	DIR	0.00 MB	01:59 08-14-20	705
12	11-PointersDynamicMemoryManagement/	DIR	0.00 MB	01:58 08-14-20	705
13	12-TemplatesVectorsStacks/	DIR	0.00 MB	01:58 08-14-20	705
14	13-FileInputOutput/	DIR	0.00 MB	01:58 08-14-20	705
15	14-OperatorOverloading/	DIR	0.00 MB	01:58 08-14-20	705
16	15-InheritancePolymorphism/	DIR	0.00 MB	01:58 08-14-20	705
17	16-ExceptionHandling/	DIR	0.00 MB	01:58 08-14-20	705
18	17-Recursion/	DIR	0.00 MB	01:57 08-14-20	705

CSC340 iLearn: Announcements and Forums

Part C: C++ IDE Installation



Part D: JAVA Refreshment

MyFavoriteApp.java

```
import java.util.Scanner;

public class MyFavoriteApp {

    public static void main(String[] args) {

        Scanner reader = new Scanner(System.in);
        System.out.print("Enter two numbers: ");

        double first = reader.nextDouble();
        double second = reader.nextDouble();

        System.out.print("Choose an operator symbol (+, -, *, /): ");
        char operator = reader.next().charAt(0);

        double result;

        switch(operator)
        {
            case '+':
                result = first + second;
                break;
            case '-':
                result = first - second;
                break;
            case '*':
                result = first * second;
```

MyFavoriteApp.java

```
        double result;

        switch(operator)
        {
            case '+':
                result = first + second;
                break;
            case '-':
                result = first - second;
                break;
            case '*':
                result = first * second;
                break;
            case '/':
                result = first / second;
                break;
            default:
                System.out.printf("Please use an appropriate operating symbol.");
                return;
        }
        System.out.printf("%.1f %c %.1f = %.1f\n", first, operator, second, result);
    }
}
```

Find: Previous Next Select

MyFavoriteApp > main > switch (operator) > case '-' >

Output

MyFavoriteApp (jar) | MyFavoriteApp (run)

```
ant -f C:\\Users\\Tino\\Desktop\\MyFavoriteApp -Dnb.internal.action.name=run run
init:
Deleting: C:\Users\Tino\Desktop\MyFavoriteApp\build\built-jar.properties
deps-jar:
Updating property file: C:\Users\Tino\Desktop\MyFavoriteApp\build\built-jar.properties
compile:
run:
Enter two numbers: 1 1
Choose an operator symbol (+, -, *, /): +
1.0 + 1.0 = 2.0
BUILD SUCCESSFUL (total time: 5 seconds)
```

c) In order to improve my program I would incorporate a check for when there is no need for a double. I could do this by using System.out.printf() or System.out.format(). This would allow me to control the number of decimal spaces visible in the output. I would also expand on the number of operators to work with for example I would include modulus, or even exponents or square root.