Student Name:李佳燕 Assignment: asm01

Student ID: 0416039

Student Email: girlannie123@gmail.com

**Claim: I worked on my own. ( YES/ NO ) Must tick one.**

**If this is not your own work, your score is zero.**

DO NOT CHEAT in programming. You should learn on your own.

**Introduction (at least 150 words) [15%]**

Word Count: 163 (must filled, or zero point)

The Object of this homework is to use the floating point unit to compute the number pi. First of all, we have to ask the user to enter a number, which is represented the number of terms in the formula / 4 1 – 1/3 + 1/5 – 1/7 + …. Then we should write the program to compute the answer of pi according to the number of terms and the formula. Finally, print out the answer on the screen. If the number the user enter is bigger then 100,000, then set the number to 100,000. If the number the user enter is zero, then exit the program.

The program is always computing until the user enter zero. That is, whenever the user enter the number, the program will run. And when user enter zero, the program will exit finally. The point is that how to compute pi according to the number of the terms and the formula, and I’ll explain my idea below.

**Program description (at least 200 words) [20%]**

Word Count: 209 (must filled, or zero point)

First, give the initial value to the floating variable( pi, current to zero, sign to zero and previous to minus one). Then print the string on the screen to ask user to enter the number of terms. Comparing the number to know that if the number is bigger 100,000 or equal zero or not. If the number is not equal zero and lower 100,000, give the number of terms to ecx. If the number is bigger then 100,000, give the max number( 100,000 ) to the ecx. If the number is equal zero, exit the program.

Next, go into a for loop to compute pi. First, plus two to the previous and give the answer to current in the loop, which we can know what is the next denominator we have to compute. Then do the division between current and sign. After adding the answer of division and pi we put the answer into pi, changing the sign of variable sign, then loop again until ecx = 0 and we exit the loop. After finish computing, multiply pi by four and print the answer on the screen. Then print the string to ask user to enter the number of terms, which we can compute the new answer of pi.

**Methodology (at least 300 words) [30%]**

Word Count: 328 (must filled, or zero point)

First of all, assembling language is different from C or C++, so it is different to give the initial value. So how to put the initial value into the floating variable? Here is the solution. Declare a variable which content the initial value can solve the problem. Let the variable we just declare store into stack, then pop it to the variable we want to initial. In this way, we can solve the problem.

Second, how to determined whether the number of terms is equal, bigger then 100,000 or not? Use the function cmp to solve the problem! And call another function if the number of term is greater than 100,000 or equal to zero.

Third, how to plus two to the previous and store the answer into current? The solution is very easy. Declare a variable which content the number two. Then let the variable and the value of previous store into stack, adding them and store the answer into ST(0). Then we can give the value to current. Of course, we also have to give the value to previous so we can compute the next number which have to plus two.

Then, it’s time to calculate the answer pi. Clean the stack and put the value of current and the variable which named sign into stack. Then do the division, adding the value pi and store the answer into pi. And we will get the current answer to pi. Of course, don’t forget to change the sign of the variable sign so that we can meet the formula / 4 1 – 1/3 + 1/5 – 1/7 + …

When ecx become zero, we exit the loop. And we can prepare the answer to pi on the screen after we finish multiply pi by 4. Declare a variable and give the value of four. Then put the variable and pi into stack and multiply them. After storing the value into pi, then we can print the answer.

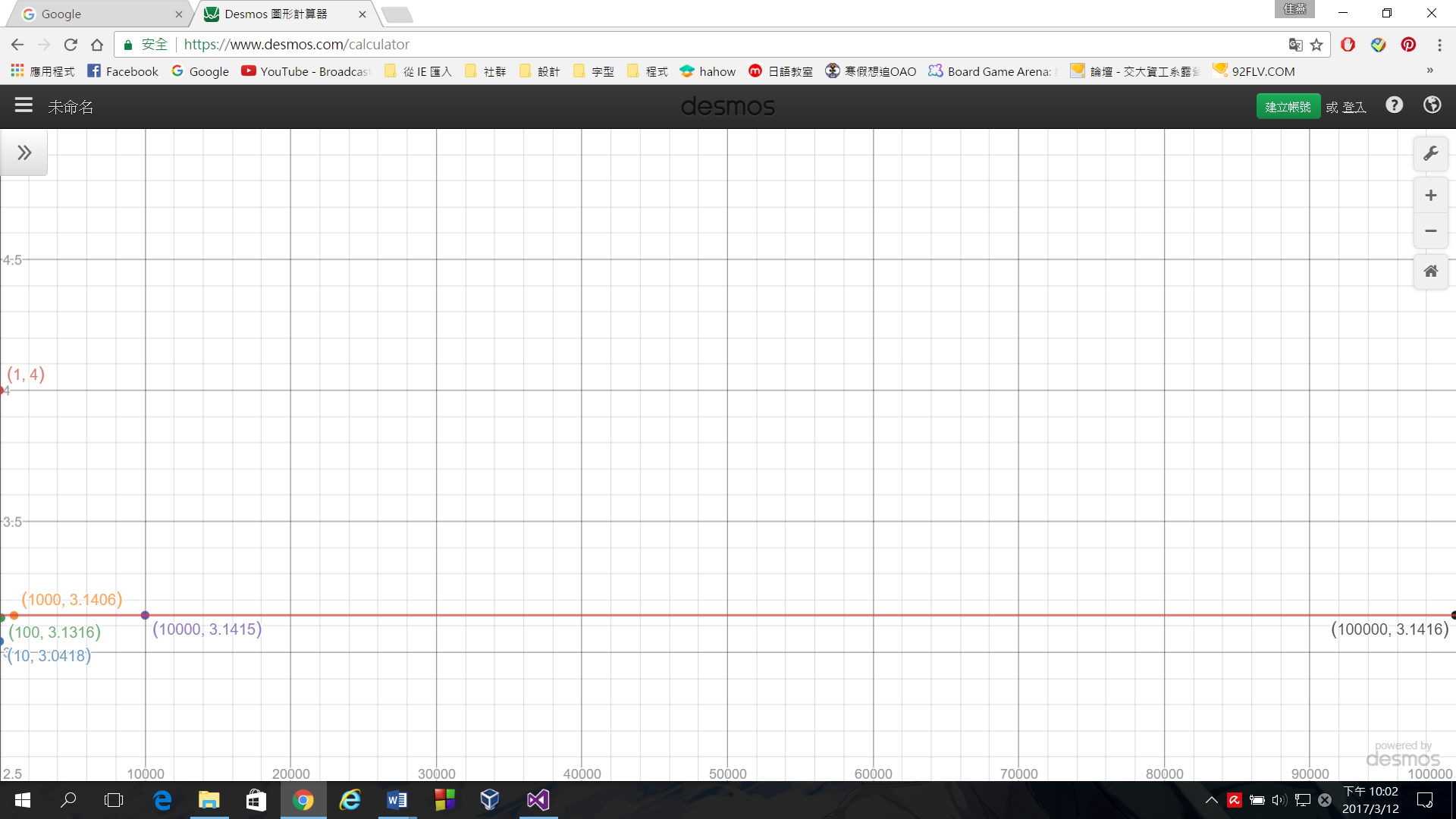
**Experiments (at least 200 words) [20%]**

Word Count: 200 (must filled, or zero point)

The table that contain number of terms, compute value of pi, and actual value of pi

|  |  |  |
| --- | --- | --- |
| Number of Terms | Computed Value of Pi | Actual Value of Pi |
| 1 | 4.0 | 3.14159265359 |
| 10 | 3.0418396 | 3.14159265359 |
| 100 | 3.1315929 | 3.14159265359 |
| 1000 | 3.1405926 | 3.14159265359 |
| 10000 | 3.1414926 | 3.14159265359 |
| 100000 | 3.1415826 | 3.14159265359 |
| 1000000 | 3.1415826 | 3.14159265359 |

The graph that different number of terms and the estimated value of pi:



It is obviously that when the number of terms become bigger, the estimated value of pi is closer to the actual value of pi. In the graph, the red line is represented the actual value of pi. When the number of terms is one, it is far to the line. However, when the number of terms becomes bigger, it is easy to see that they are much closer than previous. But it is not true that the value is always bigger than pi. When the number of terms is 10, the estimated value of pi is smaller than actual value of pi. So not any number of terms to calculate pi is bigger than pi. The only thing we can make sure just that if the number become bigger, the estimated value is closer to actual number of pi.

**Conclusion (at least 150 words) [15%]**

Word Count: 150 (must filled, or zero point)

Actually, this assignment is my first assembling program writing only by myself. Before this assignment, I just wrote the program when I was in the class, typing what teacher type on the screen. So to me, it is a challenge to writing the assignment. Of course, typing in the class is very helpful. Without typing in the class, I can not writing the assignment such quickly. I believe that if I didn’t write the program when I was in the class, I would do this assignment more than one day. When I finished the assignment, I have a sense of achievement. I have a feeling that I learn something new! And I am sure that I am more familiar to the assembling language. I know that I still have a lot of thing to do. So I will try to do my best to learn as much as I can.