Assignment 2

Numerical Computing

Rudolf C. Kischer

260956107

Floating point in C, Overflow and Underflow, numerical cancellation

Q3. $1 + \frac{1}{4} + \frac{1}{9} + \frac{1}{16} + ...$

converges to $\frac{\pi^2}{6}$

Question:

a) Write a C program to sum the series, terminating if the sum is unchanged by the addition of a term. How good is the result using single precision?

Answer:

Summation Function:

```c
float sum_square_reciprocals(int n){
    float sum = 0.0f;
    for(int i=1; i<=n; i++) {
        sum += 1.0f / pow(i, 2.0f);
    }
    return sum;
}
```

Approximation Function:

```c
float approach_congergent_value() {
    float convergent_value = pow(M_PI,2) / 6.0f;
    float prev_diff = MAXFLOAT;
    int i = 0;

    while (true) {
      float sum = sum_square_reciprocals(i);
      printf("i = %d, sum = %.30f\n", i, sum);
      //difference between convergent value and sum
      float diff = fabs(convergent_value - sum);
      if (diff >= prev_diff) {
        break;
      }
      prev_diff = diff;
      i++;
    }
    return sum_square_reciprocals(i-1);
}
```

Input:

```
int main() {

    float sum = approach_congergent_value();

    return 0;
}
```

Output:

```
i = 0, sum = 0.00000000000000000000000e+00
i = 1, sum = 1.00000000000000000000000e+00
i = 2, sum = 1.25000000000000000000000e+00
...
i = 4095, sum = 1.64472520351409912109375000e+00
i = 4096, sum = 1.64472532272338867187500000e+00
i = 4097, sum = 1.64472532272338867187500000e+00

True convergent_value = 1.64493405818939208984375000000
```

Explanation: The appoximation takes many iterations, to reach a value which does not change. We can see that the value, although approaching the convergent value, only is accurate to 3 decimal places. This is a poor result in the context of numerical computing.

Question: b) Give an explanation for the poor result. Suppose we want to get higher accuracy based on the above series still using single precision. How do you solve this problem?

Answer: - Although this series is said to converge, it converges at the limit as n goes to infinity. However, the series converges slowly, because the terms become very small fast as because the denominator increases by the square of the index. This means that we need to add many terms to get a good approximation.

- Because the terms become increasingly small, they eventually reach a point where each new term is no longer representable in the range of single precision floating point numbers, When it reaches this point, we can no longer add to the sum, because the new term is rounded down to 0.
- There are a couple cop out solutions. One of them being directly computing the convergent value. The series can be rearranged to the convergent value, and then we can compute PI use many other methods, which might not have this problem.
- Another solution to get higher accuracy is to use double precision. This will allow us to add more terms to the sum, before we reach the limit of representable numbers.
- Both of these solutions are not ideal, because they do not directly solve the problem.