

202018021

Fu, Ziyu

Programming I Special Task 2

2021/07/07

Task 1

1. Objective

To make a program that calculates a definite integral using Riemann sum.

2. Strategy of solving

- Create a function that calculates a trapezoidal area
- Divide the integrated area into proper trapezoids and calculate the value

3. Program code

```
#include <stdio.h>
#include <math.h>
double originalf(double x){
    return x*x*x;
}

double trapezoidArea(double base1, double base2, double height){
    return 0.5*(base1+base2)*height;
}

double integrate(double a, double b, int interval){
    int flipped = 1;
    if(a>b){
        double temp = a;
        a = b;
        b = temp;
        flipped = -1;
    }else if (a==b){
        return 0;
    }

    double step = (b-a)/interval;
    double x1 = a;
    double x2 = a+step;
    double y1, y2, integral = 0;
    while(x2<=b){
        y1 = originalf(x1);
        y2 = originalf(x2);
        integral += trapezoidArea(y1, y2, step);
        x1 = x2;
        x2 += step;
    }
    return flipped * integral;
}
```

```
int main() {  
    printf("result: %f\n", integrate(0,2,10));  
    return 0;  
}
```

4. Results and discussions

Console output:

```
result: 4.040000
```

The program outputs as expected.

Professor's comment hints at finding the value by trapezoidal estimation, so the integral will be estimated by using the trapezoidal rule with 10 intervals.

There are 2 special cases that might occur (although neither can happen in this question):

- (1) If the lower bound is higher than upper bound
- (2) If the integration is from a number to itself (area = 0)

The fix is a little basic but it works reliably. However, again, the task statement specified the exact calculation that needs to be carried out, so considerations on the special cases aren't exactly necessary.