



파이썬 - HW2

임베디드스쿨1기

Lv1과정

2020. 08. 04

김인겸

1. c언어로 미분해보기

1) c언어 적분 복습

```
1 #include <stdio.h>
2 #include <math.h>
3
4 int main(void)
5 {
6     //0~3까지 y=x^2을 정적분해보자
7     double dx = 0.0001f; //구간 0~3을 1/10000 등분.
8     double cur_x = 0.0f; //x값의 변화량
9     double area = 0.0f; //면적
10    double y; //x값의 변화량에 따른 y값
11    int a = 0;
12    int b = 3;
13    int i;
14
15
16    int loop_num = round((b-a) / dx); //반복횟수
17
18    printf("%d\n", loop_num);
19
20    for(i = 1 ; i <= loop_num ; i++)
21    {
22        cur_x += dx;
23        y = cur_x * cur_x;
24        area += y * dx;
25    }
26
27    printf("0부터3까지 x^2의 정적분 값은 %lf 입니다\n", area);
28
29    return 0;
30 }
```

30000
0부터3까지 x^2의 정적분 값은 9.000449 입니다

1. c언어로 미분해보기

2) c언어로 미분해보기

```
1 #include <stdio.h>
2
3 // y=x^2을 임의의 값x에서 미분한 값을 알아보자
4 // 미분은 평균변화율의 극한이다
5 // x+dx에서 x에 근접할 때의 순간변화율을 알아보자.
6
7 double differential(double x);    //함수를 미리 선언.
8
9 int main(void)
10 {
11     double value;                // 최종값
12     double x_input;              // 입력받을 x값
13
14     printf("함수 y=x^2에서 알고 싶은 순간변화율의 x값을 입력하시오 : ");
15     scanf("%lf", &x_input);
16
17     value = differential(x_input); //함수 활용
18
19     printf("y=x^2의 %lf에서의 미분값은 %lf 입니다\n", x_input, value);
20
21     return 0;
22 }
```

(밑으로)

1. c언어로 미분해보기

```
23
24 double differential(double x)           // 미분값 구하는 함수를 정의
25 {
26     double dx = 1;                     // dx는 0의 양의방향으로 수렴할 것이다
27     double y2 = 0.0;                   // x + dx에서의 함수값
28     double y1;                          // x에서의 함수값
29     double gradient = 0.0;              // 기울기
30     double loop_num = 10000000;        // 반복횟수를 천만번으로 해보자.
31
32     int i;
33     y1 = x * x; //y1값은 고정
34
35     for(i = 1 ; i <= loop_num ; i++)
36     {
37         y2 = (x + dx) * (x + dx);       // x+dx에서의 y값
38         gradient = (y2 - y1) / ((x + dx) - x) ; // x와 x+dx에서의 평균변화율값
39         dx -= 1.0 / loop_num;           // dx값이 천만분의 일 만큼 계속해서 줄어든다
40     }                                   // 마지막 반복문이 실행될 때 gradient값은
41                                       // x에서의 순간변화율 값에 근접한다.
42
43     return gradient;
44 }
```

```
(base) ingyeomkim@ingyeomkim-Inspiron-7590:~/embedded/c_program
함수 y=x^2에서 알고 싶은 순간변화율의 x값을 입력하시오 : 2.3
y=x^2의 2.300000에서의 미분값은 4.600000 입니다
(base) ingyeomkim@ingyeomkim-Inspiron-7590:~/embedded/c_program
함수 y=x^2에서 알고 싶은 순간변화율의 x값을 입력하시오 : -5.11
y=x^2의 -5.110000에서의 미분값은 -10.220000 입니다
(base) ingyeomkim@ingyeomkim-Inspiron-7590:~/embedded/c_program
```


2. 파이썬으로 적분 해보기

```
# x=0부터 x=3까지 y=x^2의 정적분 값을 구해보자

a = 0          # x는 0부터
b = 3          # x는 3까지
dx = 0.0001    # x값의 증분(만분의 일)
loop = (b-a) / dx  # 반복횟수(3만번)
cur_x = 0      # x가 dx만큼 변할 때 현재의 x값
area = 0       # 면적

for i in range(1, 30000):
    cur_x += dx
    y = cur_x * cur_x
    area += y * dx

print("정적분 값은 {0} 입니다".format(area))
```

정적분 값은 8.99955000500408 입니다

3. 파이썬으로 미분 해보기

$y=x^2$ 의 $x=3$ 에서의 미분값을 구해보자

dx = 0.0001

x 값의 변화량

x1 = 3.0

$x=3$

x2 = x1 + dx

y2 = x2 * x2

$x=3+dx$ 에서의 y 값

y1 = 3.0 * 3.0

$x=3$ 에서의 y 값

gradient = (y2 - y1) / (x2 - x1)

$x=3$ 과 $x=3+dx$ 의 평균변화율

dx 값이 작아질수록 순간변화율 값에 근접한다.

print("미분값은 {} 입니다".format(gradient))

미분값은 6.000099999999392 입니다

4. 25p에서 발생한 현상 조사

for~else문 :

- 1) for~else문은 break와 같이 사용된다.
- 2) for문 안에서 break가 사용되면 else문을 출력하지 않는다.
- 3) for문 안에서 break가 사용되지 않으면 else문을 출력한다.

4. 25p에서 발생한 현상 조사

예제 1) i값이 6이 되었을 때
break에 의해 for문은 종료되고
break가 사용되었기 때문에
else문은 출력하지 않는다

```
In [5]: for i in range(1,11):  
        if i > 5:  
            break  
        print("Item{0}".format(i))  
    else:  
        print("Exit whitout break")  
    print("Always printed")
```

```
Item1  
Item2  
Item3  
Item4  
Item5  
Always printed
```

예제 2) for문에서 break가 없으므로
for문의 반복이 끝난 후
break가 사용되지 않았기 때문에
else를 출력한다.

```
In [9]: for i in range(1,11):  
        if i % 2 == 0:  
            continue  
        print("Item{0}".format(i))  
    else:  
        print("Exit whitout break")  
    print("Always printed")
```

```
Item1  
Item3  
Item5  
Item7  
Item9  
Exit whitout break  
Always printed
```


4. 25p에서 발생한 현상 조사

예제 3) for문 안에서 $i > 10$ 인 경우가 존재하지 않기 때문에 break가 사용되지 않았다.
break가 사용되지 않았기 때문에 else문을 출력한다.

```
In [12]: for i in range(1,11):  
          if i > 10:  
              break  
          print("Item{0}".format(i))  
        else:  
            print("Exit whitout break")
```

```
Item1  
Item2  
Item3  
Item4  
Item5  
Item6  
Item7  
Item8  
Item9  
Item10  
Exit whitout break
```

for~else문은 for문안에서 break가 사용됐는지 사용 되지 않았는지 알 수 있게 해준다.