

2일차 과제(C++ & Qt)

로봇 20기 인턴 최세영

목차

- hw1

```

cpp_test > include > head1.hpp > cal > pross()
1  using namespace std;
2  class cal
3  {
4  private:
5      int max,min,sum=0,count=0,*a=NULL;
6      float input=0;
7
8  public:
9      int check=0;
10     void make(){
11
12         cout<<"몇 개의 원소를 할당하겠습니까? : ";
13         cin >> input;
14
15         if((input>0&&input==(int)input)){
16             count=(int)input;a=new int[count];
17         }
18         else{cout<<"자연수를 입력하세요."<<endl;
19             check=0;}
20
21         for (int i = 0; i < count; i++){
22             input=0.1;
23             cout<<"정수형 데이터 입력 : ";
24             cin >> input;
25             if (input==(int)input){
26                 check=1;
27                 a[i]=(int)input;
28             }
29             else{
30                 cout<<"정수형을 입력하세요."<<endl;
31                 check=0;
32                 break;
33             }
34         }
35     }
36
37     void pross(){
38
39         if (check==1){
40             sum=0;
41             max=a[0];
42             min=a[0];
43
44             for (int i=0;i<count;i++){
45                 sum+=a[i];
46                 if (a[i]>max){max=a[i];}
47                 if (a[i]<min){min=a[i];}
48             }
49             cout<<"최대값: "<<max<<endl;
50             cout<<"최소값: "<<min<<endl;
51             cout<<"전체합: "<<sum<<endl;
52             cout<<"평균: "<<(float)sum/count<<endl;
53         }
54         delete [] a;
55     }

```

Hw1.hpp

- Cal 이라는 클래스 생성

-고유 변수로 int 형식의 max(최대값), min(최소값), sum(합), count(입력 개수), *a(동적 할당할 int형식 변수) 선언 후 실수 형식으로 input(입력값) 변수 선언

-공용으로 chech(입력 올바른지 확인) 변수와 make(원소 입력 및 저장), pross(계산 및 출력) 함수 선언 (make 함수)

-input으로 원소 개수 실수로 입력(문자가 들어오면 0으로 저장됨)

-실수 input이 양수이면서 int로 변환해도 값이 같고(정수)이면 count에 자연수 input 저장 후 a에 count 만큼 동적 할당

-아니면 오류문구 출력 후 check를 0으로 만들어 pross 함수 실행 막음

-count 수만큼 반복

-실수 input이 int로 변환해도 값이 같으면 input으로 정수를 받아 배열 a에 저장 (매 입력마다 input이 0.1로 초기화 되면서 문자가 들어오면 check가 0이 되도록 예외 처리함)

-아니면 오류문구 출력 후 check를 0으로 만들어 pross 함수 실행 막고 break로 반복 탈출

(pross 함수)

-check가 1일 때만 실행

-a의 첫 항을 max(최대), min(최소)로 저장 후 sum을 0으로 초기화

-a의 모든 원소들을 sum에 더하면서 max보다 큰지, min보다 작은지 비교하여 max와 min의 값 변경

-최대, 최소, 전체, 평균(출력을 float로 하도록 설정)

-a 동적 할당 해제(check==1 상관 없이)

```
cmakeLists.txt test hw_1.cpp x
cpp_test > src > hw_1.cpp > main(int, char **)
1  #include <iostream>
2  #include "../include/head1.hpp"
3  using namespace std;
4
5  int main(int argc, char ** argv){
6
7      cal start;
8      start.make();
9      start.pross();
10
11 }
```

-Include 폴더에서 head1.hpp 헤더파일(깃허브에는 hw1.hpp) 불러옴

-cal class로 된 start 선언

-make 함수 실행

-pross 함수 실행

(모든 작업은 class 내부 변수 및 함수가 다 함)

```

david@nomal:~/intern_ws/cpp_test/build$ ./test
몇 개의 원소를 할당하겠습니까? : asdf
자연수를 입력하세요 .
david@nomal:~/intern_ws/cpp_test/build$ ./test
몇 개의 원소를 할당하겠습니까? : 5
정수형 데이터 입력 : 2
정수형 데이터 입력 : 4
정수형 데이터 입력 : asd
정수형 데이터 입력 : 정수형을 입력하세요 .
david@nomal:~/intern_ws/cpp_test/build$ ./test
몇 개의 원소를 할당하겠습니까? : 5
정수형 데이터 입력 : 10
정수형 데이터 입력 : -3
정수형 데이터 입력 : 6
정수형 데이터 입력 : 23
정수형 데이터 입력 : -17
최대값 : 23
최소값 : -17
전체 합 : 19
평균 : 3.8
david@nomal:~/intern_ws/cpp_test/build$

```

Hw1.result
 -input에 글자를 넣어 ==0 상태라(자연수 X)
 -else로 빠져 오류문구 출력 및 check==0이기에 추가 출력 없음

-count 만큼 정수형 데이터 입력 받음 (자연수 O)

-글자를 넣어서 input==0.1 상태 (정수 X)

-매끄럽게 오류메세지 출력이 안되지만 break가 되어 예외 처리 수행

-제대로 입력하면 pross 실행되어 최대, 최소, 전체 합, 평균이 나옴