

객체지향프로그래밍

HW02

C211123 이준선

2023년 3월 16일

1 HW02

문자열을 입력하면 알파벳 26자를 인덱스로 하여 알파벳 수를 계산하여 다음과 같이 출력하는 프로그램 작성하시오. 단, cin.getline(), isalpha(), tolower() 함수를 사용한다. (출력 생략)

2 과제 코드

```
1 #include <iostream>
2 #include <string>
3 #include <unordered_map>
4 #include <algorithm>
5
6 class StringManager
7 {
8 public:
9     StringManager() = default;
10    ~StringManager() = default;
11
12    void printMenu() const {
13        std::cout << "영문 텍스트를 입력하세요. 히스토그램을
그립니다." << std::endl;
14        std::cout << "텍스트의 끝은 " << endChar << "
입니다. " << maximumCount << "개까지 가능합니다." << std::endl;
15    }
16
17    void getString() {
18        std::string input;
19        std::getline(std::cin, input, endChar);
20        this->input = input.substr(0, maximumCount);
21
22        for (const auto c : this->input) {
23            if (std::isalpha(c)) {
24                alphabetCount++;
25            }
26        }
27    }
28
29    auto createHistogram() {
30        for (const auto c : input) {
```

```

32         // if c is not alphabet, continue
33         if (!std::isalpha(c)) continue;
34         histogram[c]++;
35     }
36 }
37
38 const auto& getHistogram() const {
39     return histogram;
40 }
41
42 int getAlphabetCount() const {
43     return alphabetCount;
44 }
45
46 void printHistogram() {
47     for (char c = 'a'; c <= 'z'; c++) {
48         _drawBar(c);
49     }
50 }
51
52 void printStatus() const {
53     std::cout << "총 알파벳
수 " << getAlphabetCount() << std::endl;
54 }
55
56 void toLower() {
57     for (auto& c : input) {
58         c = std::tolower(c);
59     }
60 }
61
62 private:
63     std::string input;
64     int maximumCount = 10000;
65     char endChar = ',';
66     int alphabetCount = 0;
67     std::unordered_map<char, int> histogram;
68
69     void _drawBar(const char c) {
70         std::cout << c << " (" << histogram[c] << ")\t: ";
71         for (std::size_t i = 0; i < histogram[c]; i++) {
72             std::cout << "*";
73         }
74         std::cout << std::endl;
75     }
76 };
77
78 int main()
79 {
80     StringManager manager;
81     manager.printMenu();
82     manager.getString();
83     manager.toLower();

```

```

84     manager.createHistogram();
85     manager.printStatus();
86     manager.printHistogram();
87
88     return 0;
89 }

```

Listing 1: main.cpp

3 결과

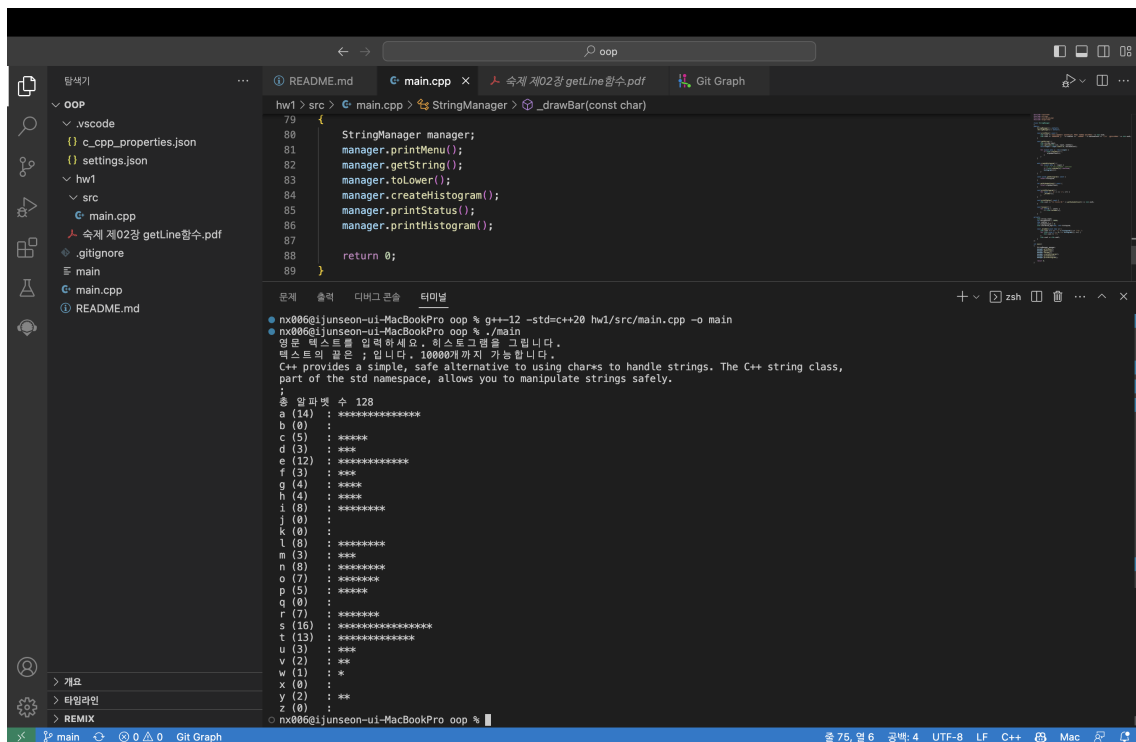


Figure 1: 결과 사진

4 분석

문자열을 입력받고, 그 문자열 속에 들어있는 알파벳들 각각의 숫자를 출력하는 문제입니다.

문자열을 입력받을 때는 getline() 함수를 이용해야 한다는 조건이 존재합니다. 또한 대문자와 소문자를 구분하지 않는데, 이때 std::tolower() 함수를 이용하라고 힌트를 주고 있습니다.

문자열의 최대 입력 갯수는 10,000개까지 가능합니다.

먼저 알파벳에 대응하는 숫자를 출력해야 합니다. 이는 char → int 쌍의 해시맵을 이용할 수 있습니다. 이때 알파벳을 순서대로 출력해야 하는데, 어차피 개수가 0인 알파벳을 이용해야 하므로 굳이 삽입 및 탐색에 O(N)의 시간이 걸리는 map을 이용할 필요는 없습니다. unordered_map을 이용하면 됩니다.

한편 오직 알파벳만 카운팅해야 합니다. 이는 isalpha 함수를 이용해야 할 것을 문제에서 힌트를 주고 있습니다.

4.1 코드 설명

```
1 void getString() {
2     std::string input;
3     std::getline(std::cin, input, endChar);
4     this->input = input.substr(0, maximumCount);
5     for (const auto c : this->input) {
6         if (std::isalpha(c)) {
7             alphabetCount++;
8         }
9     }
10 }
11
12 void toLower() {
13     for (auto& c : input) {
14         c = std::tolower(c);
15     }
16 }
```

Listing 2: main.cpp

문자열을 입력받는 함수입니다. `std::getline` 함수를 이용하고 있습니다. `endChar`는 ; 입니다. 이후에 만약 입력받은 문자열의 크기가 `maximumCount`인 10,000을 넘어가는 것을 방지하기 위해, `std::string::substr` 메소드를 이용해 0부터 `maximumCount - 1`까지 범위를 잘라서, 클래스 멤버 변수에 저장합니다.

이후에 `input`을 선형 탐색하면서, 만약 `c`가 알파벳이면 `alphabetCount`의 개수를 증가시킵니다.

주의해야 할 점은 문제에서의 조건이 대소문자를 구분하지 않는 것이기에, `input`의 모든 알파벳을 소문자로 만들어야 합니다. `std::tolower`을 이용하여 소문자로 바꿀 수 있고, 원본을 바꿔야 하기 때문에 참조를 써야 하는데 주의해야 합니다.

총 $O(n)$ 의 시간이 걸립니다.

```
1 auto createHistogram() {
2     for (const auto c : input) {
3         // if c is not alphabet, continue
4         if (!std::isalpha(c)) continue;
5         histogram[c]++;
6     }
7 }
```

Listing 3: main.cpp

해시맵인 히스토그램을 생성하는 함수입니다. `input`에 대해서 순회하며, 만약 알파벳이 아니라면 `histogram`을 생성하지 않고, 알파벳일 때 히스토그램을 생성합니다. 해당 알파벳을 키로 하여 `value`인 크기를 1씩 증가시킵니다.

```
1 void _drawBar(const char c) {
2     std::cout << c << " (" << histogram[c] << ")\t: ";
3     for (std::size_t i = 0; i < histogram[c]; i++) {
4         std::cout << "*";
5     }
6     std::cout << std::endl;
7 }
```

Listing 4: main.cpp

히스토그램 바를 그리는 함수입니다.