


LOTTO PROGRAM 만들기

곽상우 노진우 허선용 박상우



개요



알고리즘
선정과정

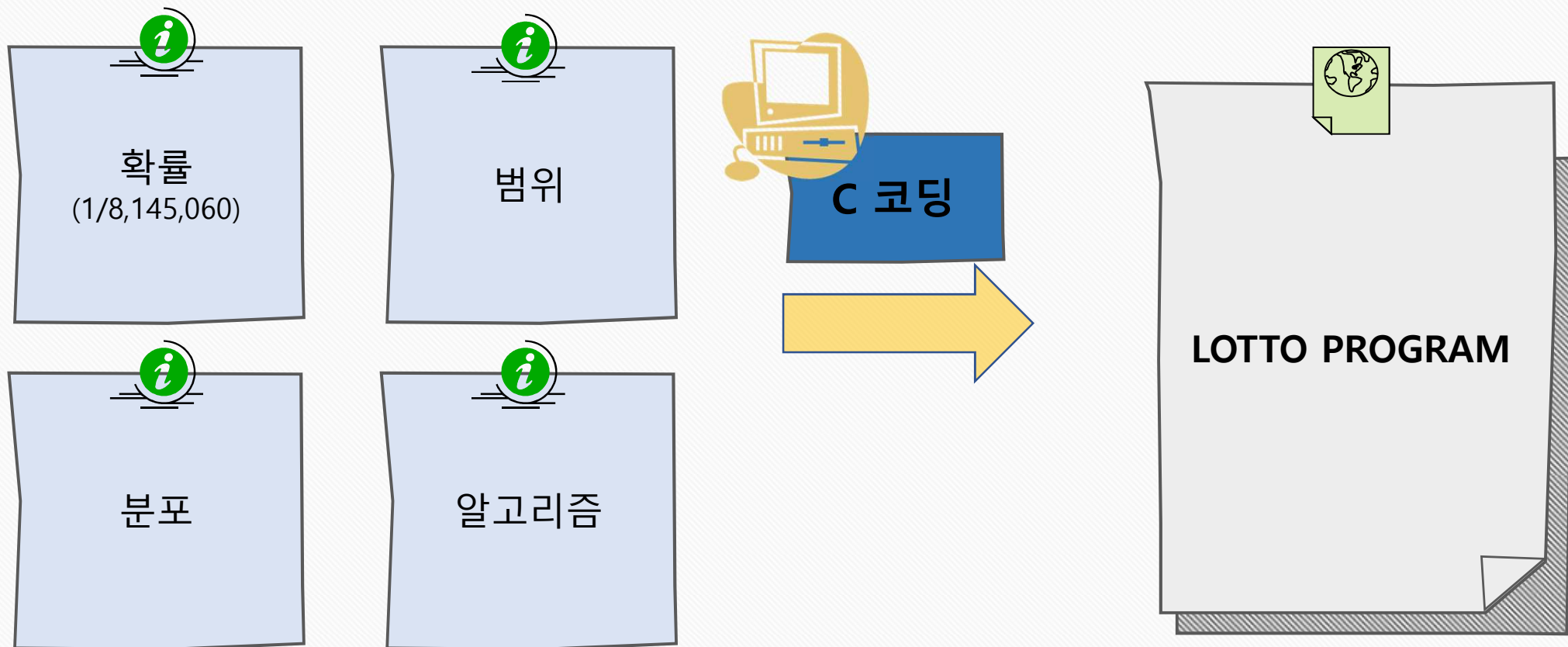


제작과정



결과





알고리즘을 찾아라.



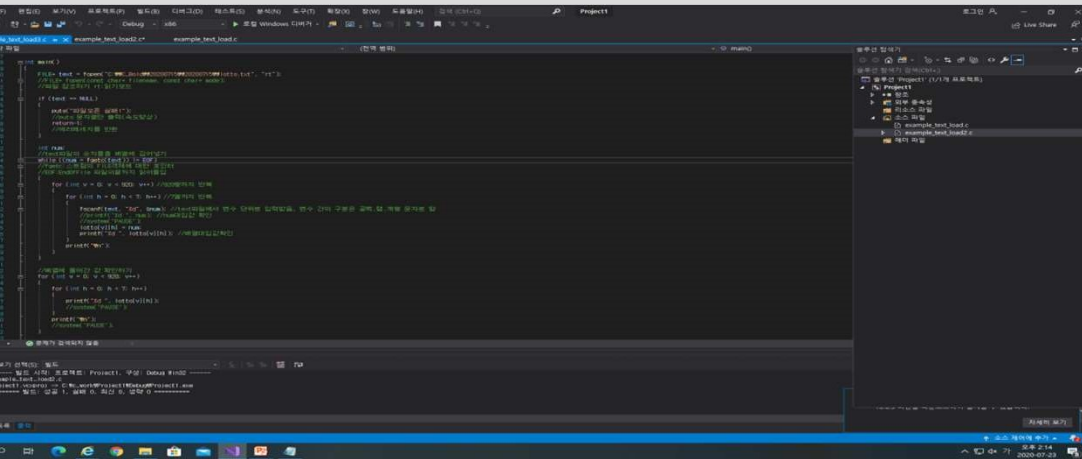
제작과정

OTTO PROGRAM



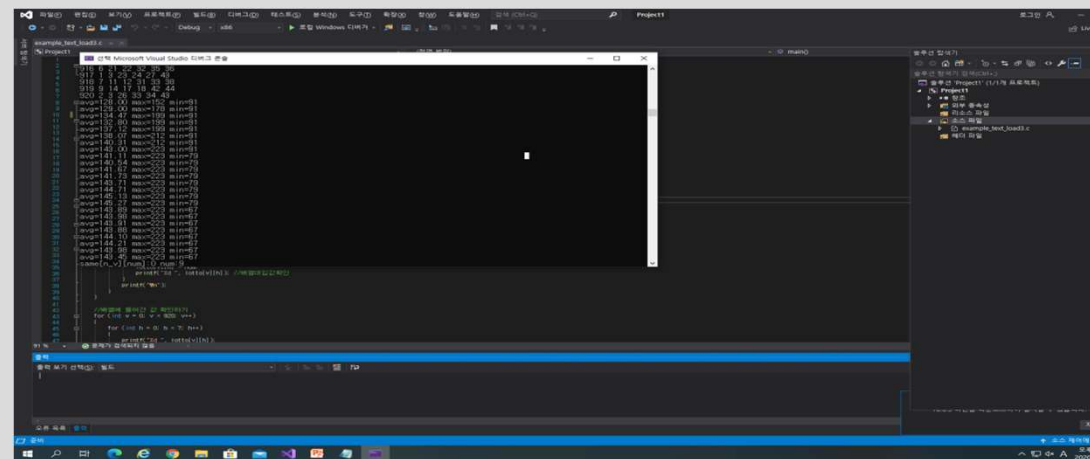
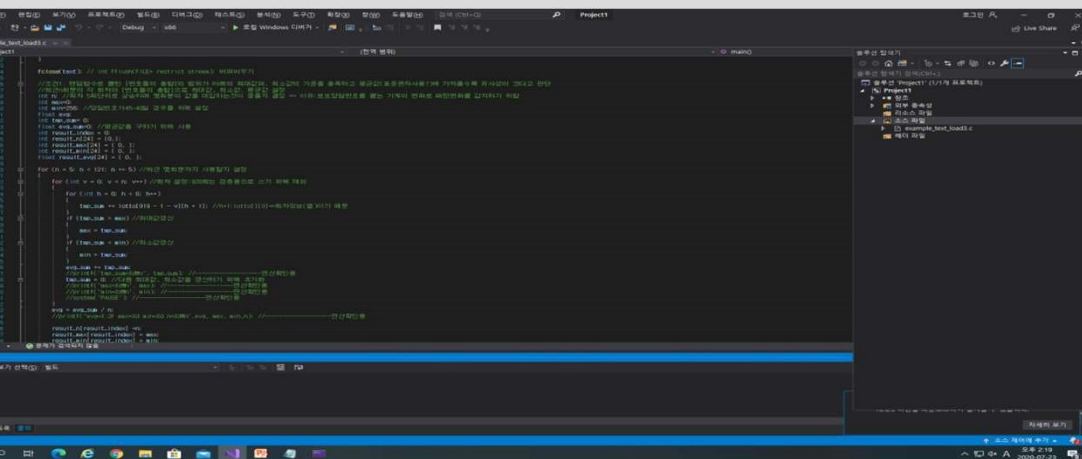
DATA 호출

호출 DATA 확인



조건1값 산출

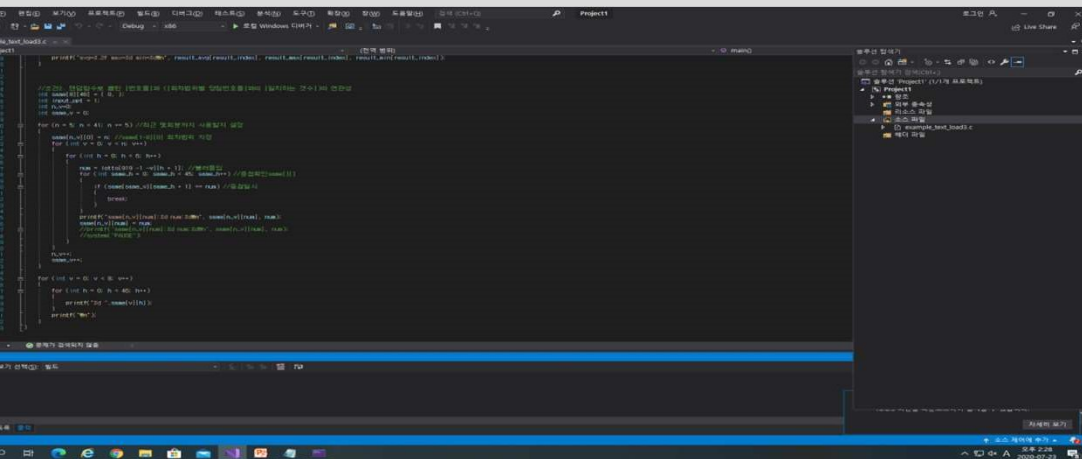
조건값 확인



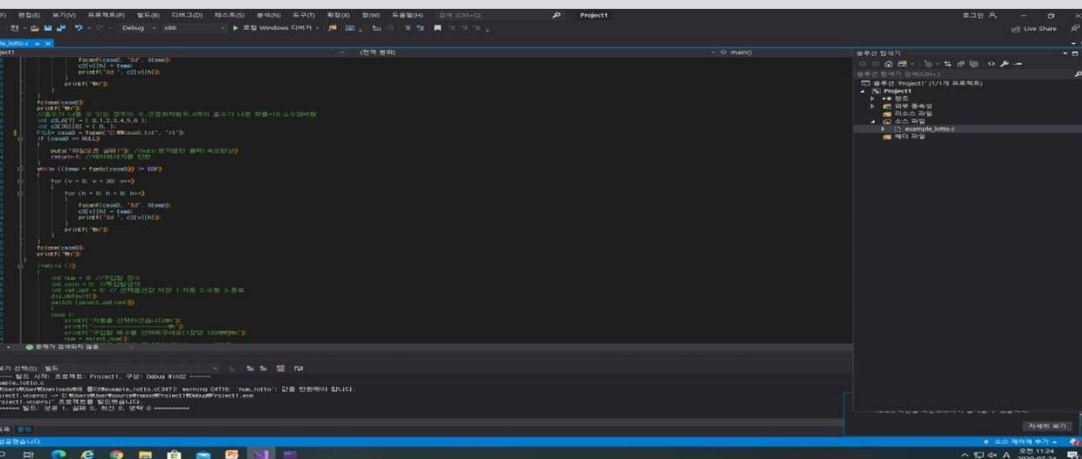
제작과정

OTTO PROGRAM

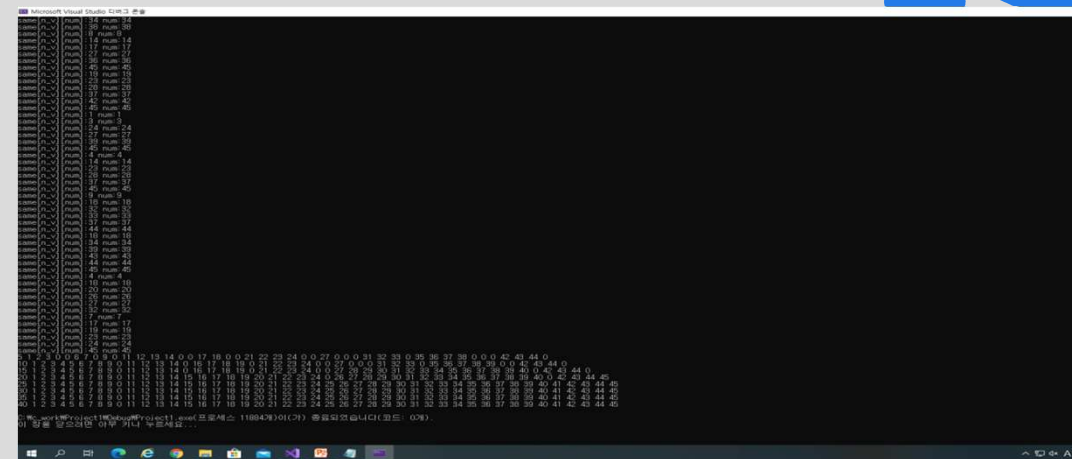
조건2값 산출



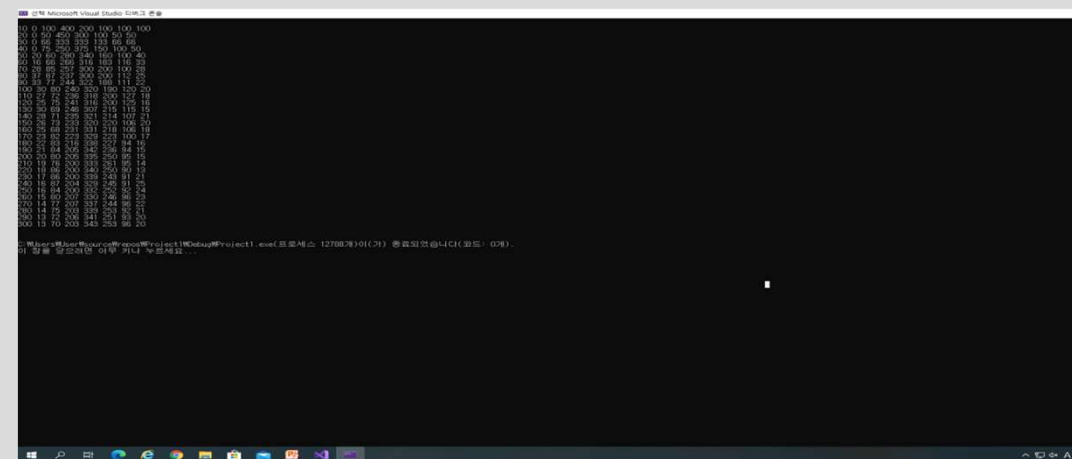
조건3값 산출



조건값 확인



조건값 확인



제작과정

로또 프로그램



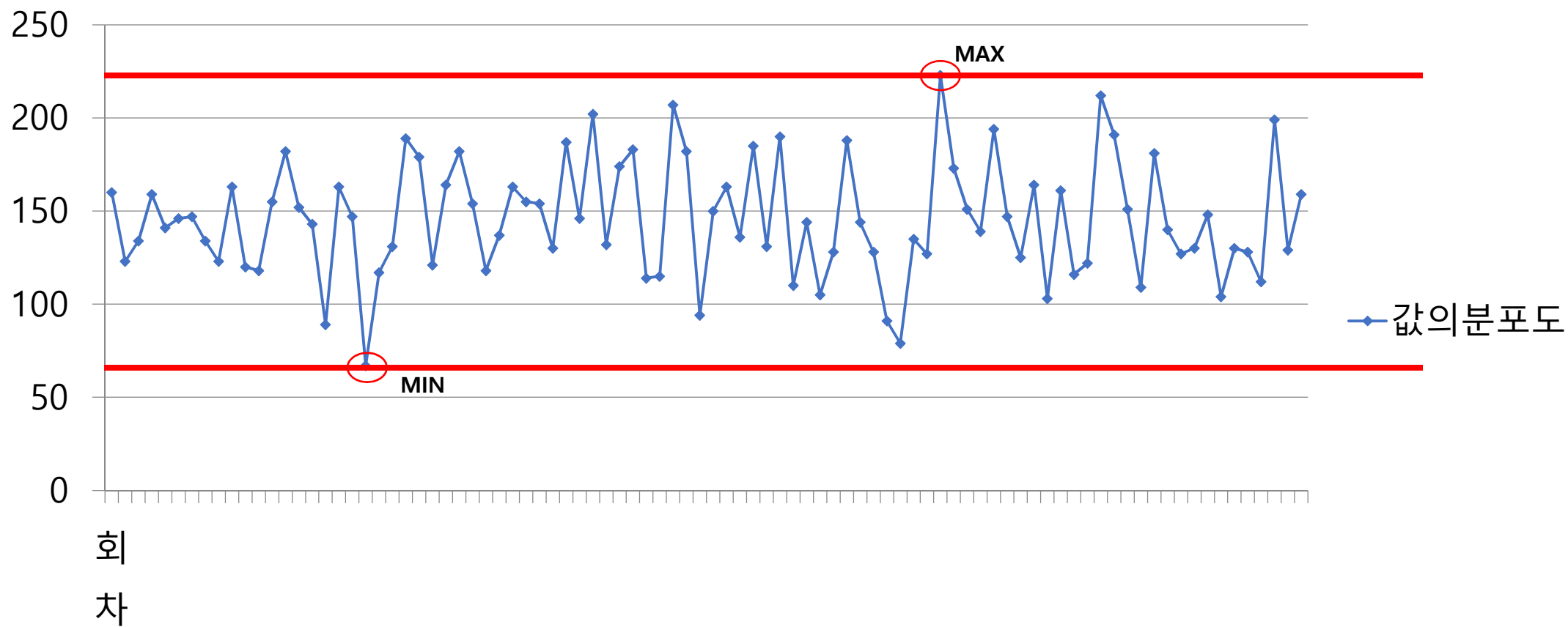
로또번호 생성

확인

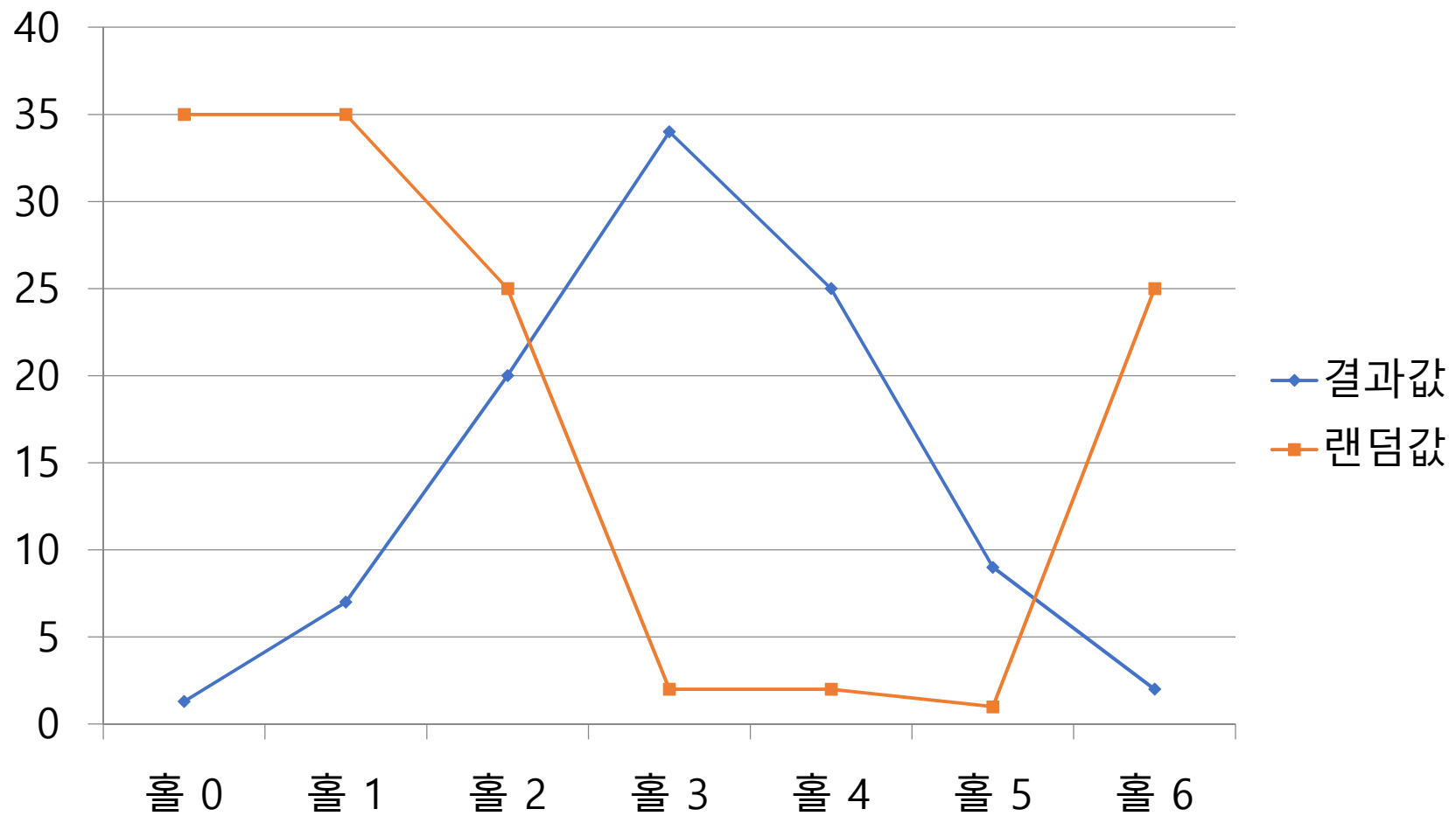
```
1 // 로또 번호 생성
2 #include <iostream>
3 #include <vector>
4 #include <algorithm>
5 #include <random>
6 #include <fstream>
7
8 using namespace std;
9
10 // 로또 번호 생성 (1~45까지의 랜덤한 6개 숫자)
11 vector<int> generateLottoNumbers() {
12     vector<int> lottoNumbers;
13     for (int i = 0; i < 6; i++) {
14         int num = rand() % 45 + 1;
15         while (find(lottoNumbers.begin(), lottoNumbers.end(), num) != lottoNumbers.end()) {
16             num = rand() % 45 + 1;
17         }
18         lottoNumbers.push_back(num);
19     }
20     sort(lottoNumbers.begin(), lottoNumbers.end());
21     return lottoNumbers;
22 }
23
24 // 로또 번호 확인 (1~45까지의 랜덤한 6개 숫자)
25 bool checkLottoNumbers(const vector<int> &lottoNumbers, const vector<int> &winningNumbers) {
26     int count = 0;
27     for (int i = 0; i < 6; i++) {
28         if (find(winningNumbers.begin(), winningNumbers.end(), lottoNumbers[i]) != winningNumbers.end()) {
29             count++;
30         }
31     }
32     return count == 6;
33 }
34
35 int main() {
36     vector<int> lottoNumbers = generateLottoNumbers();
37     vector<int> winningNumbers = {1, 2, 3, 4, 5, 6};
38     bool isWinning = checkLottoNumbers(lottoNumbers, winningNumbers);
39     if (isWinning) {
40         cout << "로또 번호 생성 성공" << endl;
41     } else {
42         cout << "로또 번호 생성 실패" << endl;
43     }
44 }
```

```
1 // 로또 번호 확인
2 #include <iostream>
3 #include <vector>
4 #include <algorithm>
5 #include <random>
6 #include <fstream>
7
8 using namespace std;
9
10 // 로또 번호 확인 (1~45까지의 랜덤한 6개 숫자)
11 bool checkLottoNumbers(const vector<int> &lottoNumbers, const vector<int> &winningNumbers) {
12     int count = 0;
13     for (int i = 0; i < 6; i++) {
14         if (find(winningNumbers.begin(), winningNumbers.end(), lottoNumbers[i]) != winningNumbers.end()) {
15             count++;
16         }
17     }
18     return count == 6;
19 }
20
21 // 로또 번호 생성 (1~45까지의 랜덤한 6개 숫자)
22 vector<int> generateLottoNumbers() {
23     vector<int> lottoNumbers;
24     for (int i = 0; i < 6; i++) {
25         int num = rand() % 45 + 1;
26         while (find(lottoNumbers.begin(), lottoNumbers.end(), num) != lottoNumbers.end()) {
27             num = rand() % 45 + 1;
28         }
29         lottoNumbers.push_back(num);
30     }
31     sort(lottoNumbers.begin(), lottoNumbers.end());
32     return lottoNumbers;
33 }
34
35 // 로또 번호 확인 (1~45까지의 랜덤한 6개 숫자)
36 bool checkLottoNumbers(const vector<int> &lottoNumbers, const vector<int> &winningNumbers) {
37     int count = 0;
38     for (int i = 0; i < 6; i++) {
39         if (find(winningNumbers.begin(), winningNumbers.end(), lottoNumbers[i]) != winningNumbers.end()) {
40             count++;
41         }
42     }
43     return count == 6;
44 }
45
46 int main() {
47     vector<int> lottoNumbers = generateLottoNumbers();
48     vector<int> winningNumbers = {1, 2, 3, 4, 5, 6};
49     bool isWinning = checkLottoNumbers(lottoNumbers, winningNumbers);
50     if (isWinning) {
51         cout << "로또 번호 생성 성공" << endl;
52     } else {
53         cout << "로또 번호 생성 실패" << endl;
54     }
55 }
```

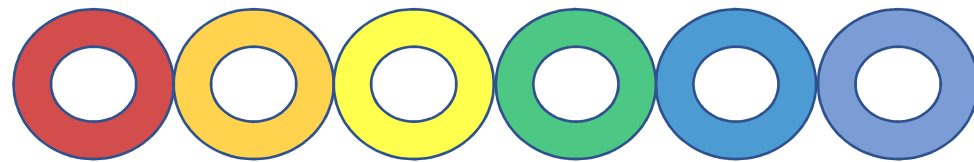
값의 분포도 (조건 1)



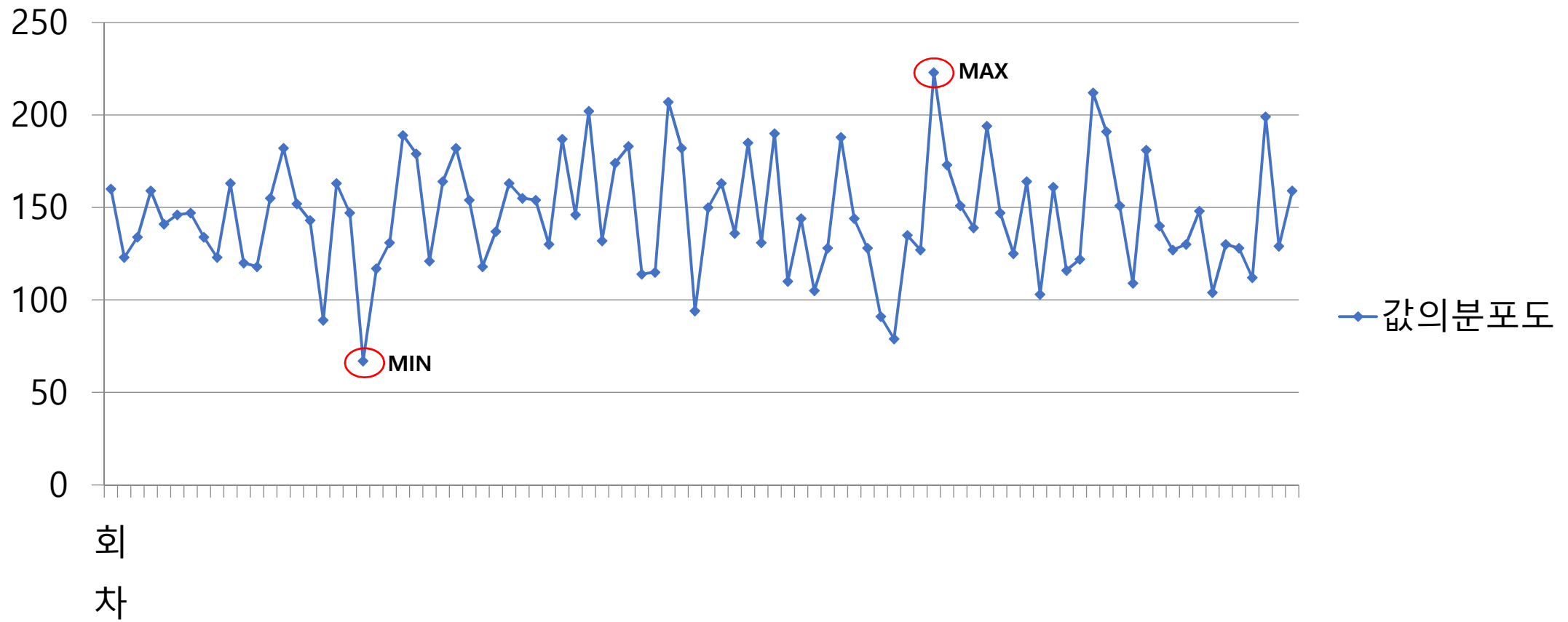
홀수 분포도(조건 3)



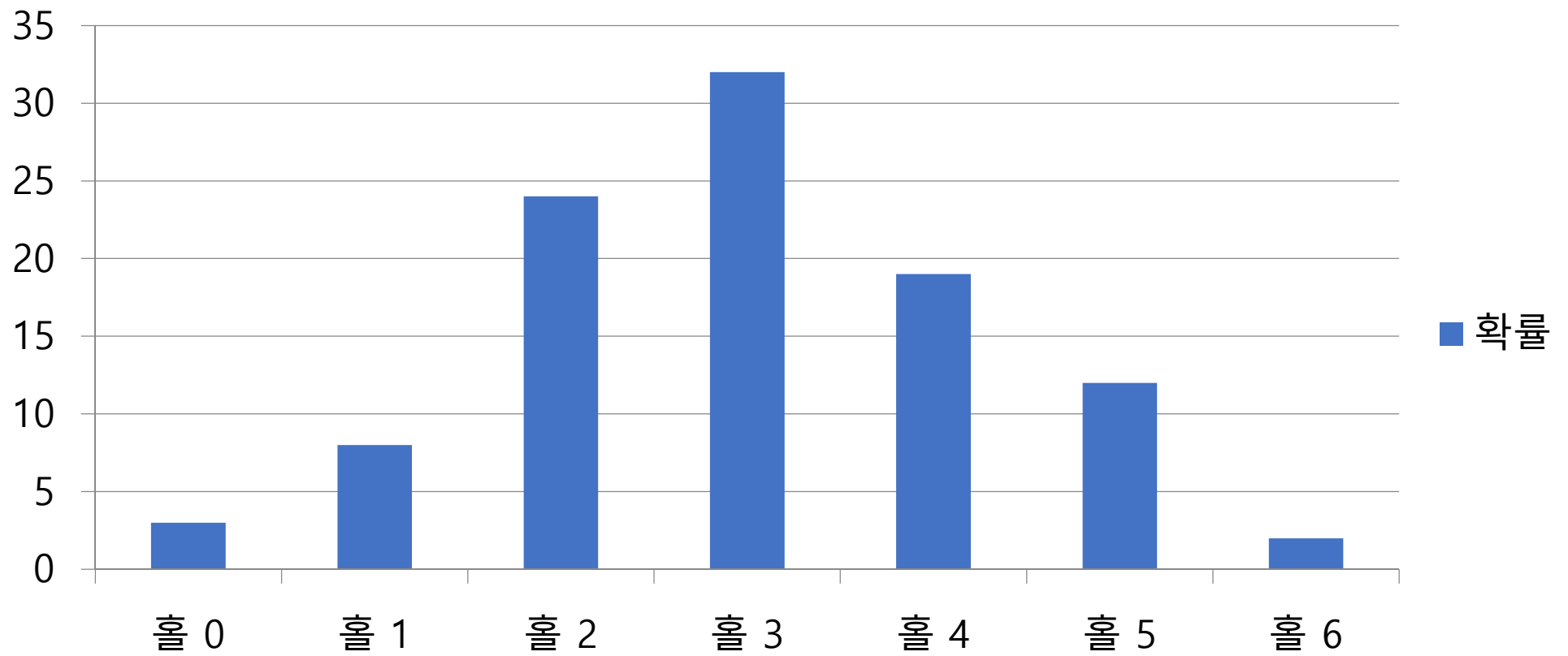
감사합니다.



값의 분포도



홀수 번호 확률



이
표

