

LOTTO



KB 7차반 차승화

목차



01

주제

02

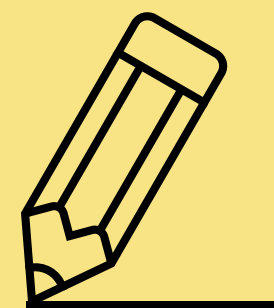
프로그래밍

03

실행 결과

01

주제



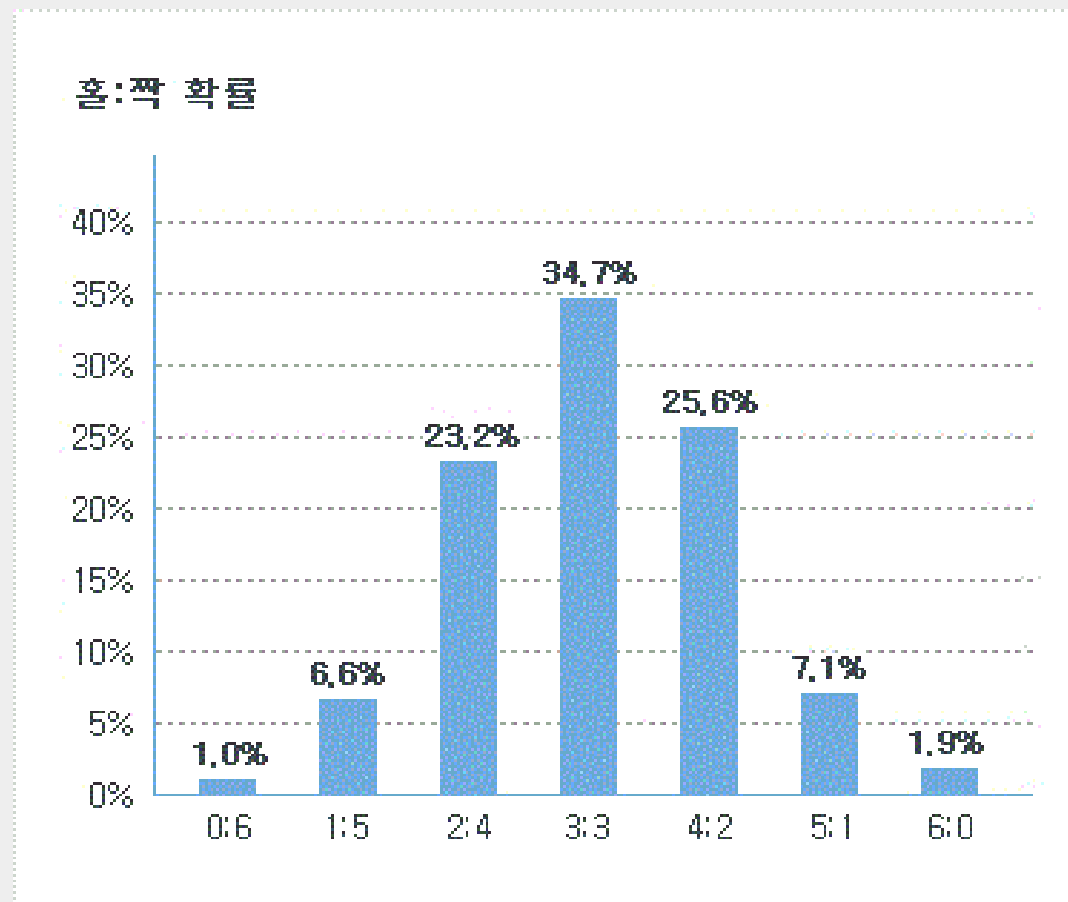
로또 프로그램



랜덤한 난수로 번호 생성



나만의 조건



나만의 조건 : 모두 홀, 짝 제외

역대 로또 당첨 숫자 6자리들의 통계이다.

보이는 바와 같이 한 회차 6자리 묶음의 통계 중

전체 홀수, 짝수의 확률은 각각 1.9%와 1.0%이다.

이 두 경우를 제외하여 6개의 숫자를 추천해보기 위해 만들었다.

02

프로그래밍



```
#include<stdio.h>
#include<stdlib.h>
#include<time.h>

// 매크로 상수 선언
#define SIZE 6
```

```
void bubbleSort(int arr[], int size)
{
    int temp = 0;
    for (int i = 0; i < size; i++)
    {
        for (int j = 0; j < size; j++)
        {
            if (arr[i] < arr[j])
            {
                temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
    }
}
```



랜덤한 난수를 뽑기 위한 srand, rand, time 함수를 쓰기 위해 <stdlib.h>와 <time.h> 헤더 파일을 사용하였다.



출력한 6가지의 수를 오름차순으로 정렬하기 위해 버블 정렬 함수를 사용하였다.

```
int odd_check(int snum[], int size)
{
    int check[SIZE];
    for (int i = 0; i < size; i++)
    {
        check[i] = (snum[i] % 1);
    }
    if (check[0] == 1 && check[1] == 1 && check[2] == 1
        && check[3] == 1 && check[4] == 1 && check[5] == 1)
        return 1;
    else
        return 0;
}
```

✔ 모두 홀수인지 판별하는 함수이다.

```
int even_check(int snum[], int size)
{
    int check[SIZE];
    for (int i = 0; i < size; i++)
    {
        check[i] = (snum[i] % 2);
    }
    if (check[0] == 0 && check[1] == 0 && check[2] == 0
        && check[3] == 0 && check[4] == 0 && check[5] == 0)
        return 1;
    else
        return 0;
}
```

✔ 모두 짝수인지 판별하는 함수이다.


```
void rand_num(int n)
{
    int odd, even;
    int num[6]; // 랜덤 생성된 로또번호
    int lotto[6] = { 0 }; // 비교될 로또번호
    srand(time(NULL));

    for (int j = 0; j < n; j++)
    {
        odd = 0;
        even = 0;
        for (int i = 0; i < SIZE; i++)
        {
            num[i] = (rand() % 45 + 1);
            if (num[i] != lotto[0] && num[i] != lotto[1] && num[i] != lotto[2] &&
                num[i] != lotto[3] && num[i] != lotto[4] && num[i] != lotto[5])
            { // 같은 줄에 같은 숫자가 있는지 배열 전체와 비교
                lotto[i] = num[i]; // 일치하지 않으면 대입
            }
            else
            {
                i--;
            }
        }

        bubbleSort(lotto, 6);
        odd = odd_check(lotto, SIZE);
        even = even_check(lotto, SIZE);

        if (odd == 1 || even == 1) // 체크 후 하나라도 1이 나올 경우 번호 재생성
        {
            j--;
        }
        else
        {
            printf("%2d세트 : ", j+1);
            for (int i = 0; i < 6; i++) // 번호 출력
            {
                printf("%2d ", lotto[i]);
            }
            printf("\n");
        }
        for (int i = 0; i < 6; i++)
        {
            lotto[i] = 0;
        }
    }

    return 0;
}
```

- ✓ 랜덤한 로또번호의 생성과 출력을 하는 함수이다.
- ✓ 로또번호를 생성 후 한 줄에 중복된 수가 있는지 검사한다.
- ✓ 앞서 만들었던 버블 정렬 함수, 홀수 체크 함수, 짝수 체크 함수를 호출한다.
- ✓ 랜덤한 로또번호의 출력
- ✓ 기존 생성된 번호 초기화

```
int main()
{
    int n;
    printf("번호를 몇 세트 출력할까요? ");
    scanf_s("%d", &n);
    rand_num(n);
    return 0;
}
```



마지막으로 앞서 만들었던 함수를 메인에서 호출하여 6개의 랜덤한 번호 출력 후 종료.

03

실행 결과



```
C:\Users\KB\Desktop\CHA\
번호를 몇 세트 출력할까요?
```

```
Microsoft Visual Studio 디버그 콘솔
번호를 몇 세트 출력할까요? 7
1세트 : 25 29 31 36 40 42
2세트 : 5 16 17 23 29 34
3세트 : 2 10 20 25 32 42
4세트 : 1 17 19 21 33 42
5세트 : 12 24 29 34 37 39
6세트 : 6 12 13 16 32 44
7세트 : 11 21 23 42 44 45

C:\Users\KB\Desktop\CHA\나의 HelloCStudy\
되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...
```

✓ 실행 결과 화면

Thank you

KB 7차반 차승화

