1. 정수 배열에서 최대값, 최소값을 찾는 C 코드

#include <stdio.h>

void findMaxMin(int arr[], int size, int \*max, int \*min) {

    \*max = arr[0];

    \*min = arr[0];

    for (int i = 1; i < size; i++) {

        if (arr[i] > \*max) {

            \*max = arr[i];

        }

        if (arr[i] < \*min) {

            \*min = arr[i];

        }

    }

}

int main() {

    int arr[] = {3, 5, 2, 10, 7};

    int size = sizeof(arr)/sizeof(arr[0]);

    int max, min;

    findMaxMin(arr, size, &max, &min);

    printf("Maximum value in array: %d\n", max);

    printf("Minimum value in array: %d\n", min);

    return 0;

}

결과

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

For loop을 이용해서 array를 순회하며 초기값 보다 큰 element를 발견할 때마다 주소를 갱신한다.

2. 정수 배열에서 index 3 인 element 제거하기

#include <stdio.h>

void delete\_element(int arr[], int size, int index) {

    if (index < 0 || index >= size) {

        printf("Invalid index\n");

        return;

    }

    for (int i = index; i < size - 1; i++) {

        arr[i] = arr[i + 1];

    }

    arr[size - 1] = 0;

}

int main() {

  int arr[] = {1, 2, 3, 4, 5};

  int size = sizeof(arr)/sizeof(arr[0]);

  int index = 3;

  delete\_element(arr, size, index);

  for (int i = 0; i < size - 1; i++) {

      printf("%d ", arr[i]);

}

  printf("\n");

  return 0;

}

결과

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

Array에서 인덱스 3인 element를 제거하기 위해서는 for loop를 이용해서 index 3이전의 값은 두고 이후의 값을 한 칸 옆으로 밀어주면 된다.

3. Upper Triangle 만들기

#include <stdio.h>

void makeUpperTriangle(int arr[3][3], int size) {

    for (int i = 0; i < size; i++) {

        for (int j = 0; j < i; j++) {

            arr[i][j] = 0; // set lower triangle values to 0

        }

    }

}

void printArray(int arr[3][3], int size) {

    for (int i = 0; i < size; i++) {

        for (int j = 0; j < 3; j++) {

            printf("%d ", arr[i][j]);

        }

        printf("\n");

    }

}

int main() {

    int arr[3][3] = {{12, 34, 67}, {13, 56, 65}, {14, 45, 63}};

    int size = 3;

    makeUpperTriangle(arr, size);

    printArray(arr, size);

    return 0;

}

결과

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

i>j 일 때의 값들을 0으로 바꿔주면 되는데, 이를 nested for loop를 사용해서 2D array의 row와 column을 순회하면서 바꿔준다.

2D array의 경우 결과값을 출력해서 확인하기 위해서 마찬가지로 nested for loop로 element들을 순회해준다.

4. 배열 정렬

#include <stdio.h>

void sort(char \*arr[], int n) {

    char \*temp;

    int i, j;

    for(i = 0; i < n-1; i++) {

        for(j = i+1; j < n; j++) {

            if(arr[i][0] > arr[j][0]) {

                temp = arr[i];

                arr[i] = arr[j];

                arr[j] = temp;

            }

        }

    }

}

int main() {

    char \*names[4] = {"Zamal", "Fakir", "Amal", "Kalam"};

    int n = 4, i;

    printf("Original array:\n");

    for(i = 0; i < n; i++) {

        printf("%s\n", names[i]);

    }

    sort(names, n);

    printf("\nArray after sorting:\n");

    for(i = 0; i < n; i++) {

        printf("%s\n", names[i]);

    }

    return 0;

}

결과

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

For loop를 통해서 순회하며 각 단어의 첫 글자를 비교하고 temp(임시저장)를 이용해서 서로의 위치를 바꾸어준다. 이후 정렬의 비교를 위해서 출력해주었다.