**計算機概論 期中專題**

**醫工一 葉芸茜 B812110004**

1. Version 1
2. 程式碼

#include<stdio.h>

int judge\_number(int);

void print\_element(int, int, int, int, int a[]);

int main(){

    int arr[] = {0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28};

    int size = sizeof(arr) / sizeof(int);

    int key, i;

    while(1){

        //the user inputs which number need to be searched

        printf("Enter a number between 0 and 28: ");

        scanf("%d", &key);

        //judge the key whether between 0 and 28

        if(judge\_number(key) == 1){

            break;

        }

    }

    printf("\nSubscripts:\n");

    //print the index of each element

    for(i=0; i<size; i++){

        printf("%2d", i);

        printf("%\*s", 3, " ");

    }

    printf("\n-------------------------------------------------------------------------\n");

    int left = 0, right = size-1;  //left等同索引值0，right等同索引值size-1

    int middle = left + (right - left) / 2;  //取中間值，middle = (left + right)/2

    int find\_index = 0;

    //print array

    print\_element(i, left, right, middle, arr);

    printf("\n");

    //binary search

    do{

        if(key < arr[middle]){

            right = middle - 1;

            middle = left + (right - left) / 2;

            print\_element(i, left, right, middle, arr);

        }

        else if(key > arr[middle]){

            left = middle + 1;

            middle = left + (right - left) / 2;

            print\_element(i, left, right, middle, arr);

        }

        else{

            find\_index = middle;

            break;

        }

        printf("\n");

    }while(left<=right);

    //if the number isn't existed, print not found

    if(find\_index == 0){

        printf("%d not found", key);

    }

    //print the index where the number be searched

    else{

        printf("\n%d found in array element %d", key, middle);

    }

    return 0;

}

int judge\_number(int num){

    if(num<0 || num>28){

        printf("\nThe number that you enter is out of range. Please enter again.\n\n");

        return 0;

    }

    return 1;

}

void print\_element(int i, int left, int right, int middle, int a[]){

    for(i=1; i<=left; i++){  //vertical alignment of the elements

        printf("%\*s", 5, " ");

    }

    for(i=left; i<=right; i++){

        if(i == middle){

            printf("%2d%c", a[i], '\*');  //label the middle index of the subarray

            printf("%\*s", 2, " ");

        }

        else{

            printf("%2d", a[i]);

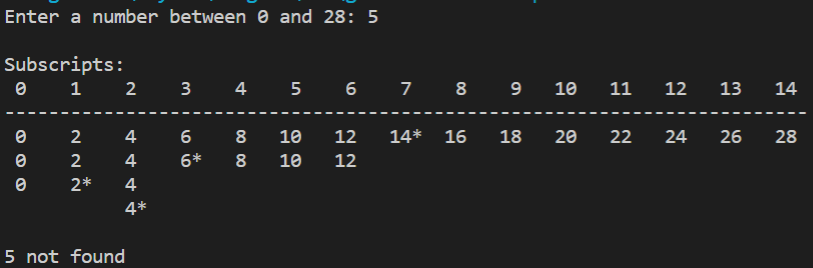
            printf("%\*s", 3," ");

        }

    }

}

1. 輸出結果
2. The output with test number 5



1. The output with test number 28

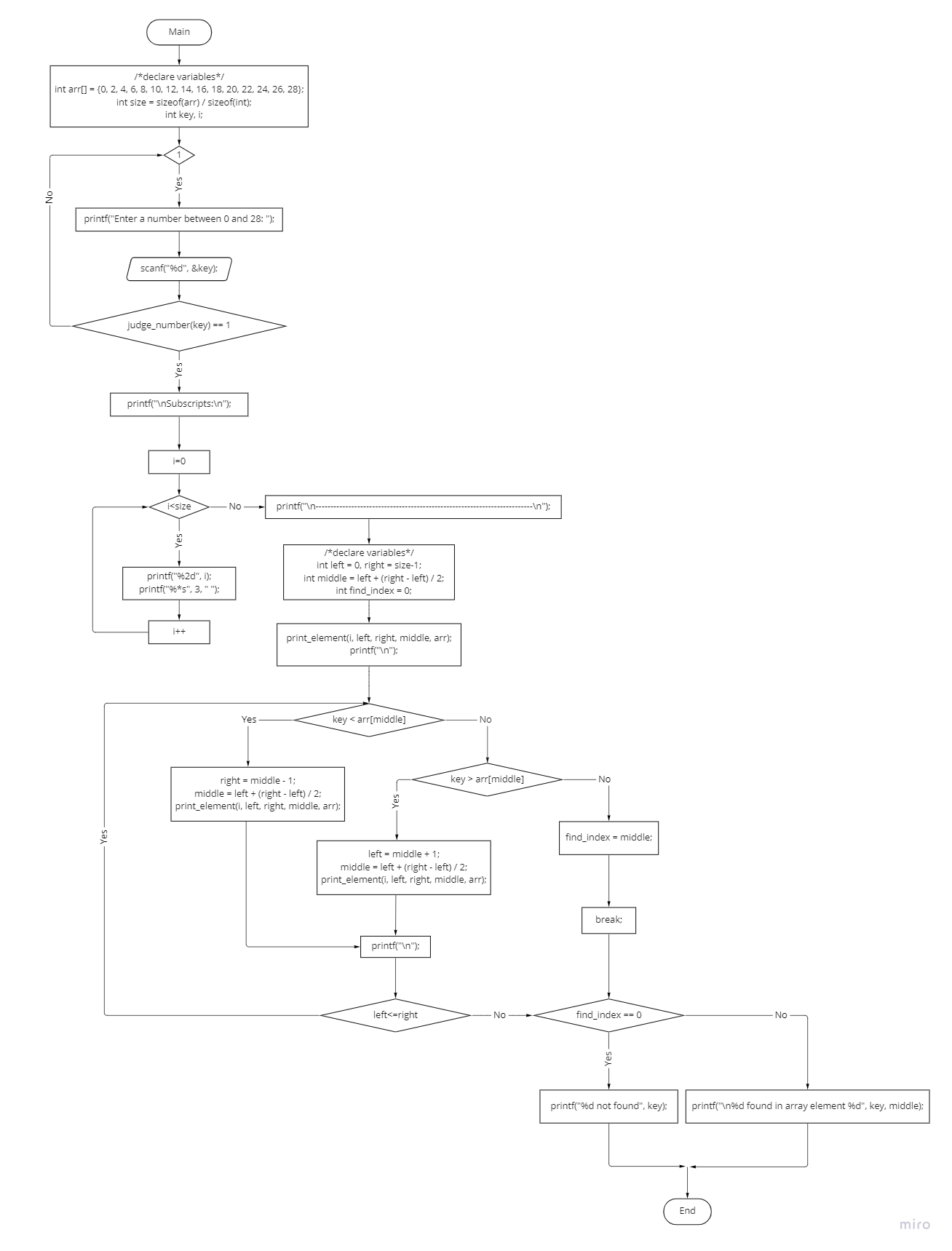
一張含有 文字, 黑色, 螢幕, 靠近 的圖片

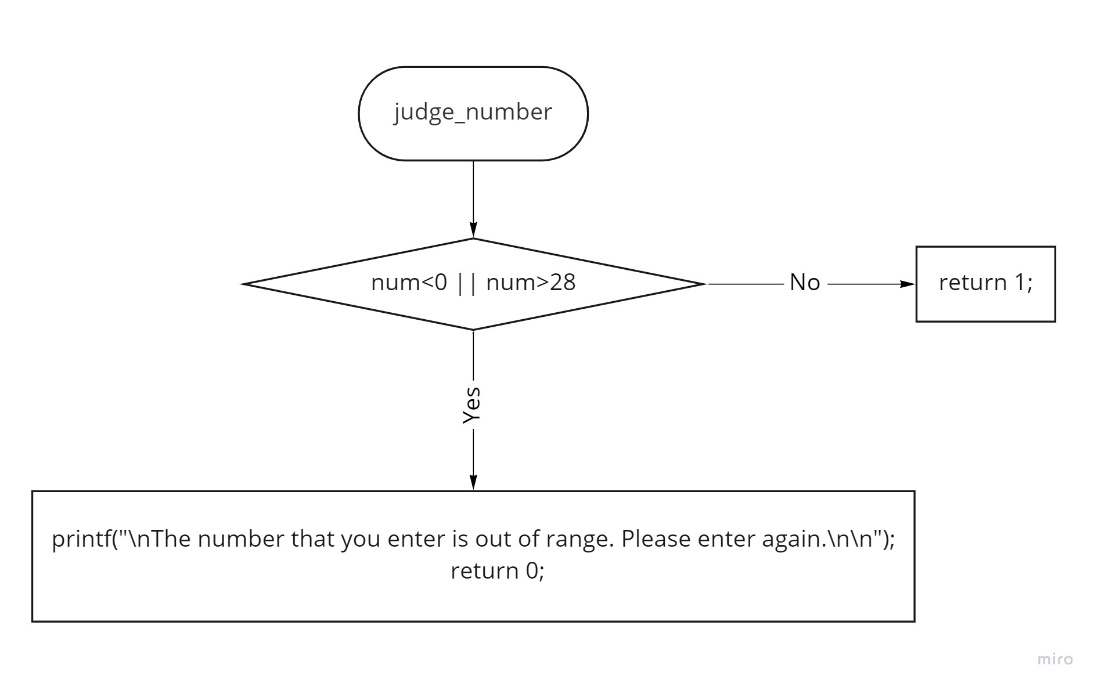
自動產生的描述

1. 流程圖(flowchart)

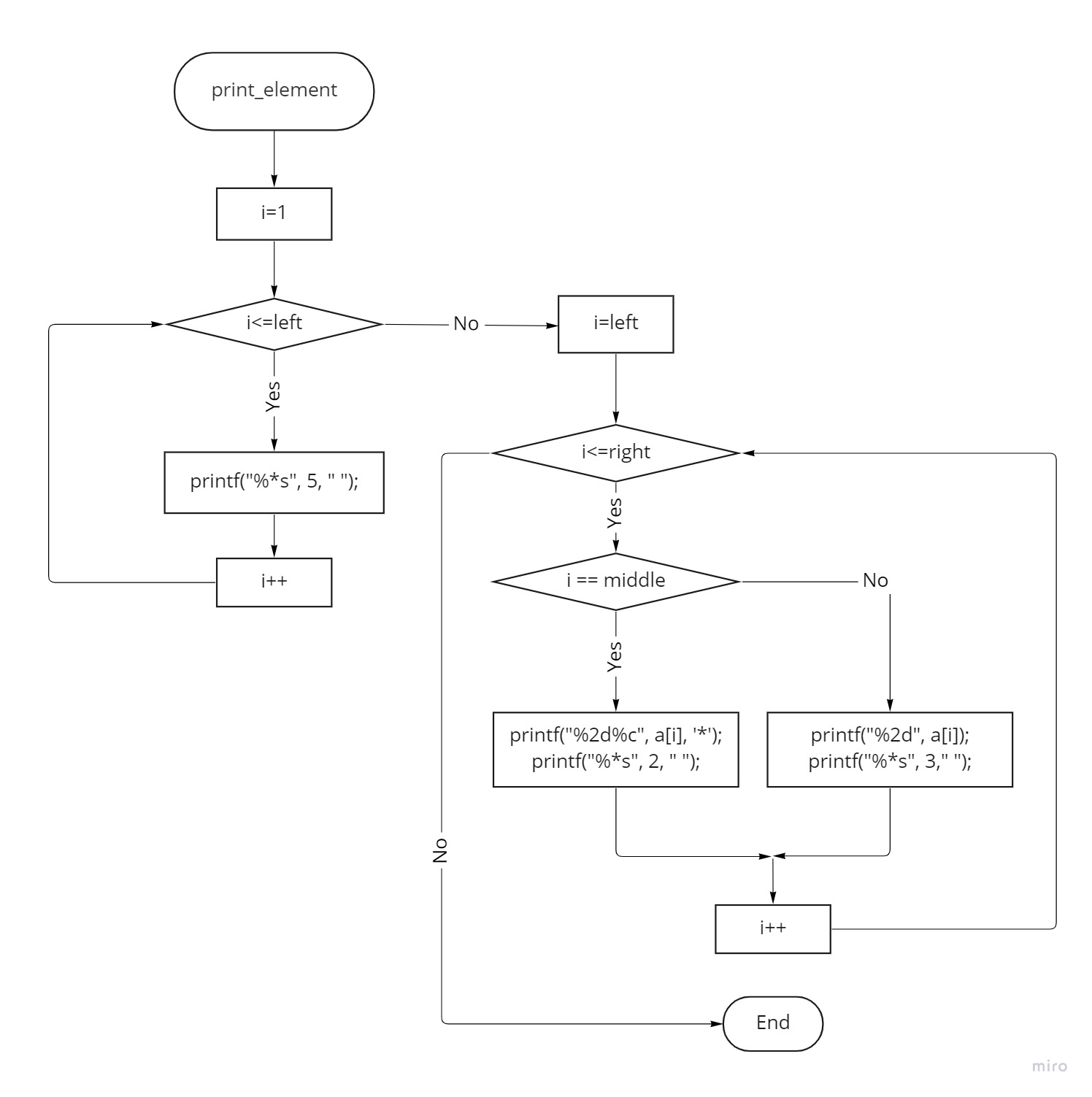
因為main function流程圖畫質有點模糊，所以另附網址:

<https://miro.com/app/board/uXjVO18V-aE=/?share_link_id=370553248973>

1. main
2. judge\_number



1. print\_element



1. version 2

改使用recursive function及ternary operator(簡化if-else)的概念來改寫。並以assert來判斷所輸入的key是否介於0~28之間

1. 程式碼

#include<stdio.h>

#include<assert.h>

void print\_element(int, int, int, int a[]);

int binary\_search(int, int, int, int arr[]);

int main(){

    int arr[] = {0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28};

    int size = sizeof(arr) / sizeof(int);

    int key, i;

    printf("Enter a number between 0 and 28: ");

    scanf("%d", &key);

    assert(key >= 0 && key <=28);

    printf("\nSubscripts:\n");

    //print the index of each element

    for(i=0; i<size; i++){

        printf("%2d", i);

        printf("%\*s", 3, " ");

    }

    printf("\n-------------------------------------------------------------------------\n");

    int result = binary\_search(0, size-1, key, arr);

    //ternary operator

    (result == -1)

        ? printf("%d not found", key)  //if result == -1 is true

        : printf("%d found in array element %d", key, result);  //if result == -1 is false

    return 0;

}

void print\_element(int left, int right, int middle, int a[]){

    int i;

    for(i=1; i<=left; i++){  //vertical alignment of the elements

        printf("%\*s", 5, " ");

    }

    for(i=left; i<=right; i++){

        if(i == middle){

            printf("%2d%c", a[i], '\*');  //label the middle index of the subarray

            printf("%\*s", 2, " ");

        }

        else{

            printf("%2d", a[i]);

            printf("%\*s", 3," ");

        }

    }

    printf("\n");

}

int binary\_search(int left, int right, int key, int a[]){

    int i, middle = left + (right-left) / 2;

    if(left <= right){

        if(key == a[middle]){

            print\_element(left, right, middle, a);

            return middle;

        }

        else if(key < a[middle]){

            print\_element(left, right, middle, a);

            return binary\_search(left, middle-1, key, a);

        }

        else{

            print\_element(left, right, middle, a);

            return binary\_search(middle+1, right, key, a);

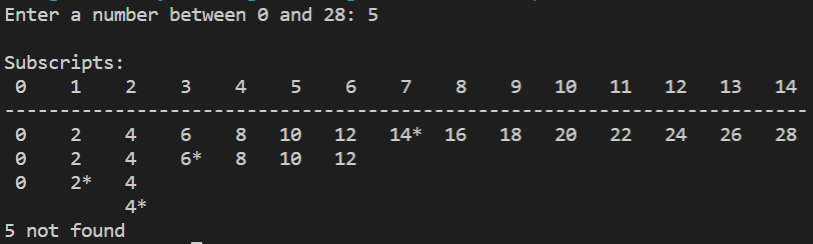
        }

    }

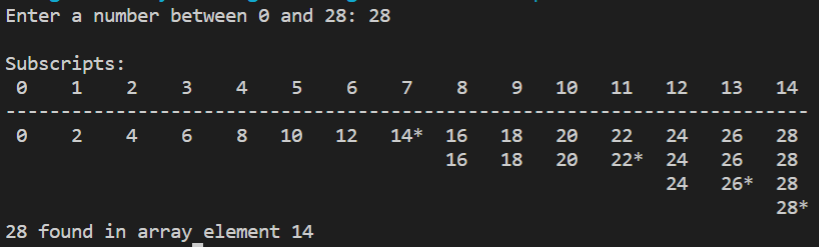
    return -1;

}

1. 輸出結果
2. The output with test number 5



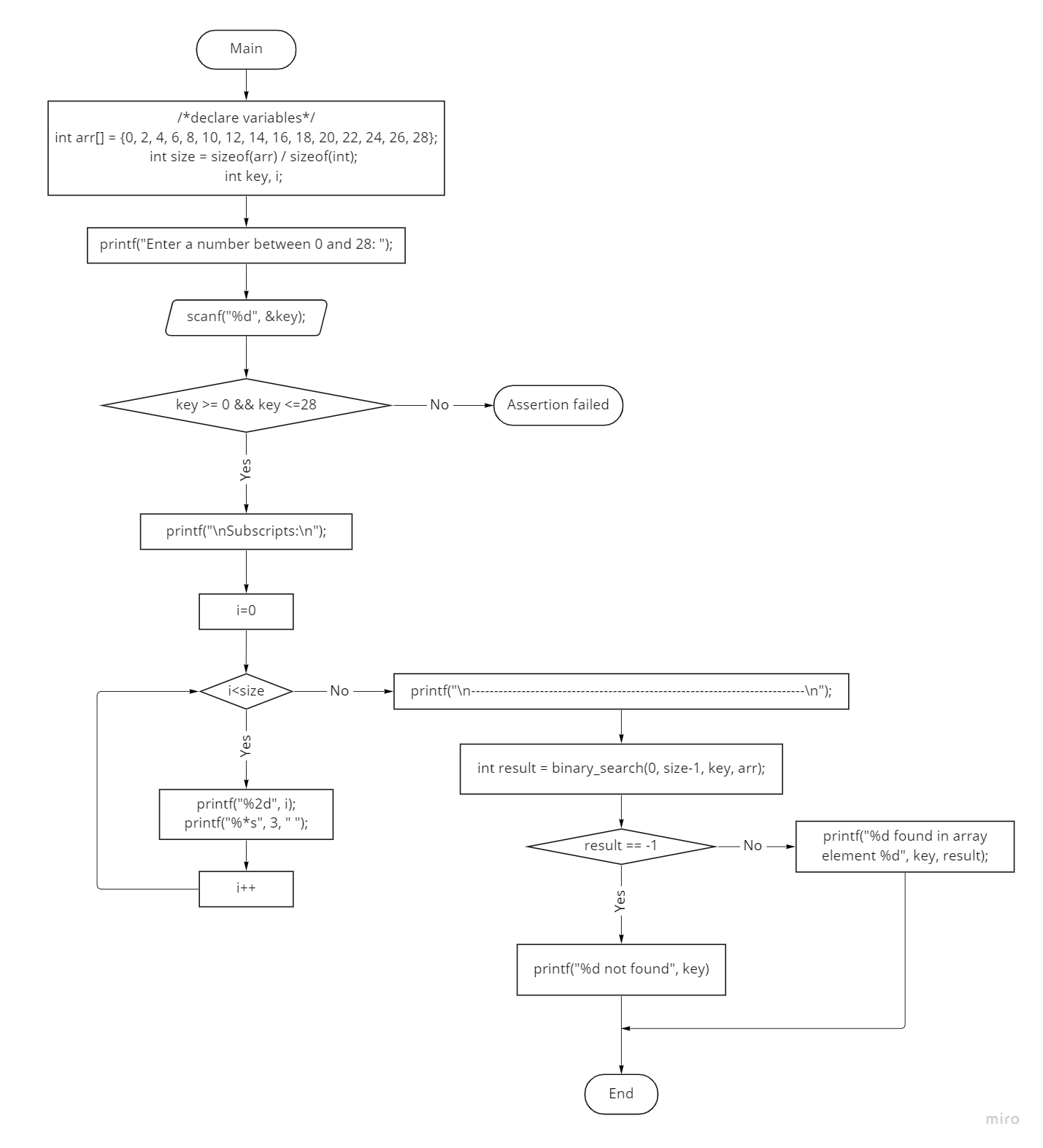
1. The output with test number 28



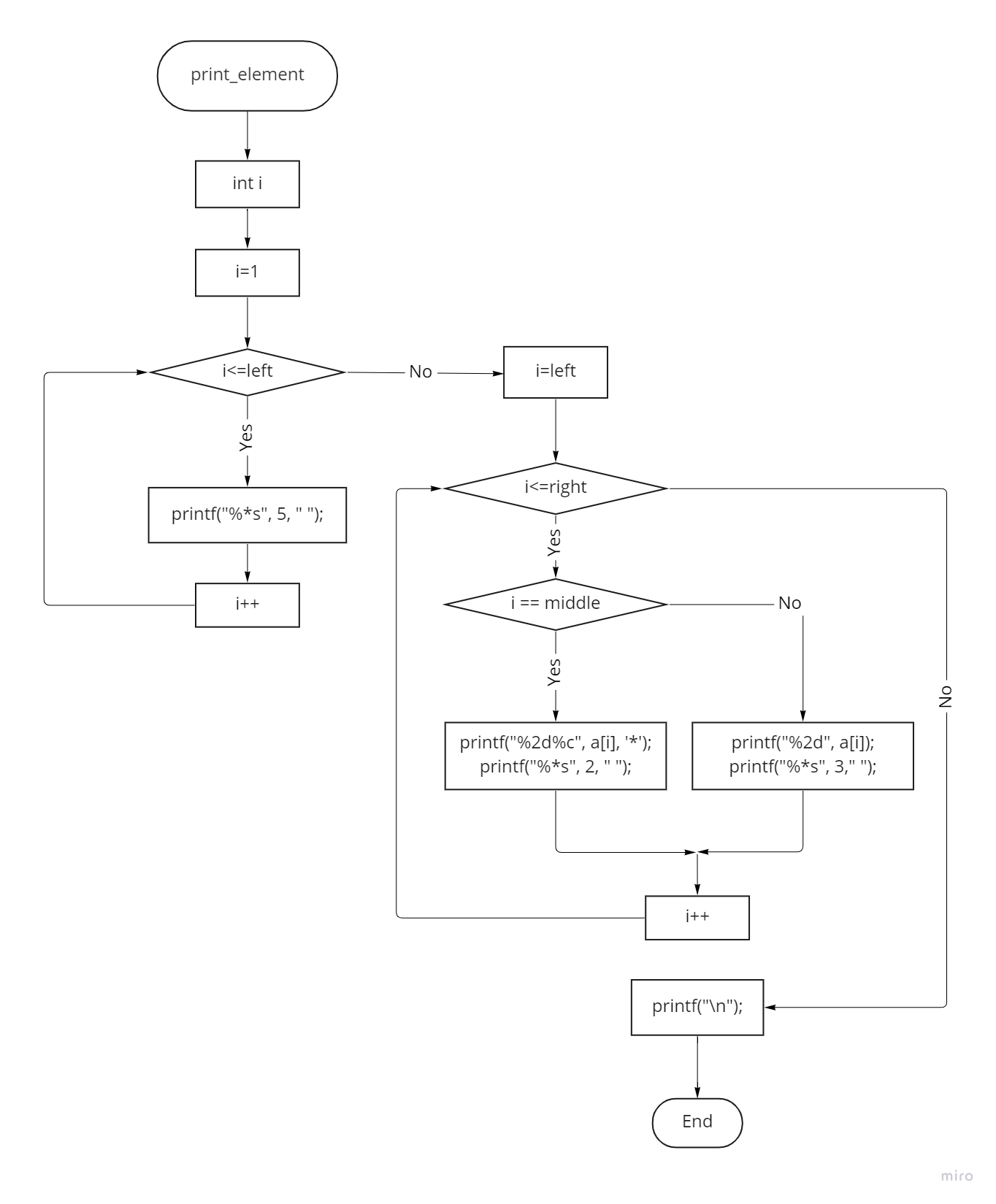
1. 流程圖(flowchart)

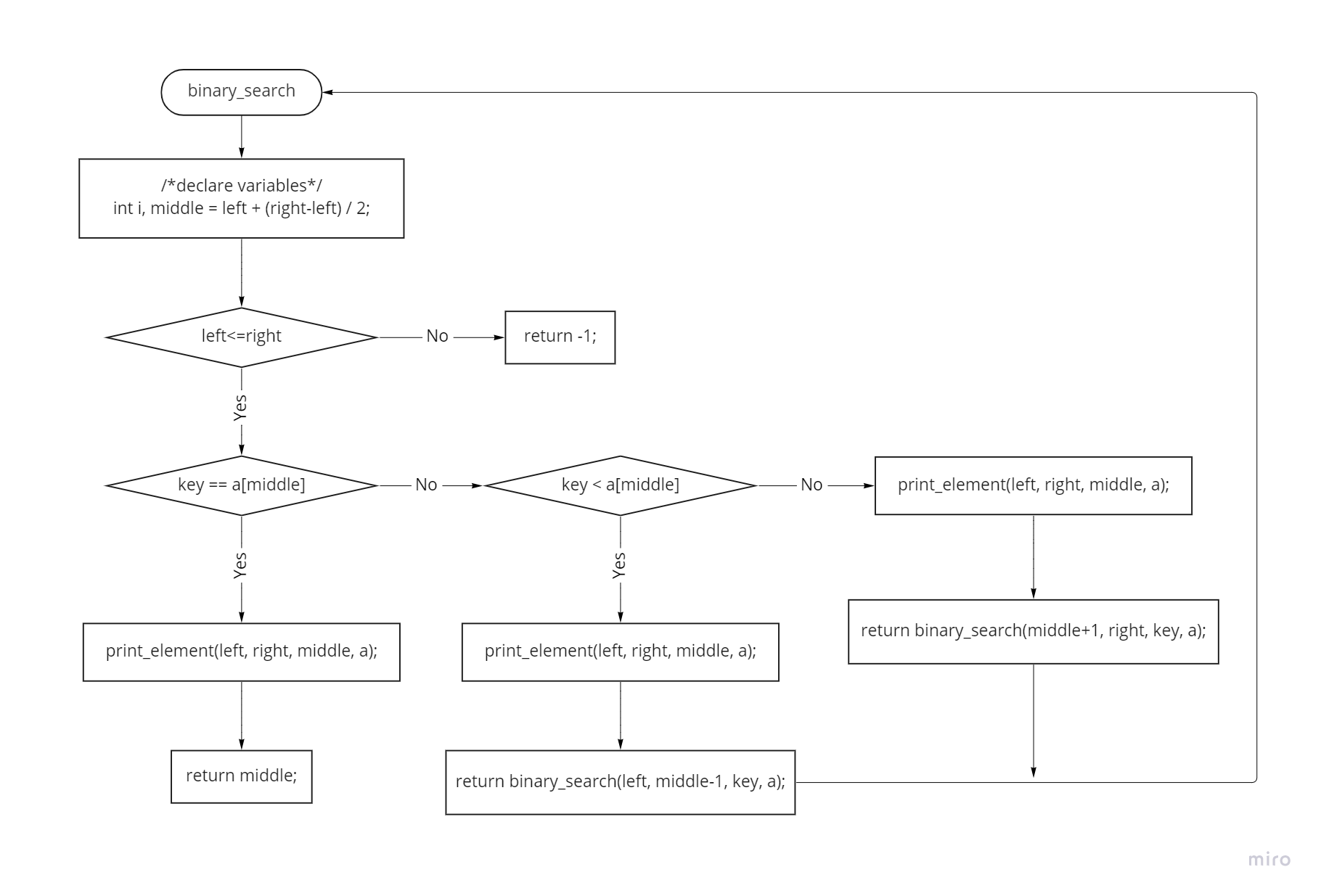
<https://miro.com/app/board/uXjVO0NgLms=/?share_link_id=481868842412>

1. main



1. print\_element



1. binary\_search