PROGRAM NO: 1 DATE: 10-08-2023

**AIM**

To find minimum and maximum number present in an array.

**ALGORITHM**

1. START
2. Initialize min and max variable with first element of the array.
3. Start traversing the array from 1st element to last element (nth)
4. Compare each value with min and max variables.
5. If the value is higher than max, replace the max value.
6. If the value is lower than min, replace the min value.
7. Outside the for loop, print the min and max values.
8. STOP

**CODE**

// Find maximum and minimum element in an array.

#include <iostream>

using namespace std;

void findMinMax(int arr[], int size) {

    int i, min = arr[0], max = arr[0];

    for (i=1; i<size; i++) {

        if (arr[i] < min) {

            min = arr[i];

        }

        if (arr[i] > max) {

            max = arr[i];

        }

    };

    cout << "Minimum value is : " << min << endl;

    cout << "Maximum value is : " << max << endl;

    cout << endl;

}

int main() {

    int size, i;

    cout << "Enter size of array: ";

    cin >> size;

    int arr[size];

    for (i=0; i<size; i++) {

        cout << "Enter element " << i + 1 << " : ";

        cin >> arr[i];

    }

    cout << endl;

    findMinMax(arr, size);

    cout << "--------- Author ----------------" << endl;

    cout << "Ali Izzath Shazin" << endl;

    cout << "220071601028" << endl;

    cout << "B. Tech CSE A" << endl;

    return 0;

}

**OUTPUT**

Enter size of array: 5

Enter element 1 : 5

Enter element 2 : -2

Enter element 3 : -3

Enter element 4 : 2

Enter element 5 : 6

Minimum value is : -3

Maximum value is : 6

--------- Author ----------------

Ali Izzath Shazin

220071601028

B. Tech CSE A

PROGRAM NO: 2 DATE: 10-08-2023

**AIM**

To find consecutive letters if they appear in the given character array.

**ALGORITHM**

1. START
2. Initialize variables i, flag and count with 0.
3. Initalize variable ‘last’ with ASCII value of first character in the array.
4. Declare variable length with length of array.
5. Declare two arrays startIndexes and endIndexes with size length.
6. Initialize a for loop from i=1 to i<length
7. Inside for loop, initialize variable current with ASCII value of ith element.
8. If for loop is in the last iteration, check if flag is 1.
9. If flag is 1, check if current – 1 is last (include both upper and lower cases).
10. If true, set endIndexes[count] = i
11. If false, set endIndexes[count] = i - 1
12. If flag is not 1, check if current – 1 is last (include both upper and lower cases).
13. If true, set startIndexes[count] = i – 1 and endIndexes[count] = i
14. If loop is not in the last iteration and current – 1 is last (include both upper and lower cases) and flag is 0. Set startIndexes[count] = i – 1 and flag = 1
15. If loop is not in the last iteration and current – 1 is not last (include both upper and lower cases) and flag is 1. Set endIndexes[count] = i – 1 and flag = 0 and increment count
16. Outside for loop, print all the elements between indexes provides in startIndexes and endIndexes.
17. STOP

**CODE**

// Find consecutive letters if they appear in the given character array.

#include <iostream>

using namespace std;

void printTextPart(string text, int start, int end) {

    for (int i=start; i<=end; i++) {

        cout << text[i];

    }

    cout << endl;

}

int main() {

    int i, last, current, flag=0, count=0;

    string text;

    cout << "Enter the text to check : ";

    cin >> text;

    last = (int) text[0];

    int length = text.length();

    int startIndexes[length], endIndexes[length];

    for (i=1; i<length; i++) {

        current = (int) text[i];

        if (i == length - 1) {

            if (flag == 1) {

                if (current - 1 == last || current - last - 1 == 32 || last - a current + 1 == 32) {

                    endIndexes[count] = i;

                } else {

                    endIndexes[count] = i - 1;

                }

            } else {

                if (current - 1 == last || current - last - 1 == 32 || last – s current + 1 == 32) {

                    startIndexes[count] = i - 1;

                    endIndexes[count] = i;

                }

            }

        }

        else if (current - 1 == last || current - last - 1 == 32 || last - aaaaaa current + 1 == 32) {

            if (flag == 0) {

                startIndexes[count] = i - 1;

                flag = 1;

            }

        } else {

            if (flag == 1) {

                endIndexes[count] = i - 1;

                count++;

                flag = 0;

            }

        }

        last = current;

    }

    for (i=0; i<=count; i++) {

        printTextPart(text, startIndexes[i], endIndexes[i]);

    }

    cout << "--------- Author ----------------" << endl;

    cout << "Ali Izzath Shazin" << endl;

    cout << "220071601028" << endl;

    cout << "B. Tech CSE A" << endl;

    return 0;

}

**OUTPUT**

Enter the text to check : AbcDhlmNquvWz

AbcD

lmN

uvW

--------- Author ----------------

Ali Izzath Shazin

220071601028

B. Tech CSE A