

SOLUCIONES ACTIVIDAD 7NP

- 1) Implementa un programa que permita introducir 24 temperaturas y calcule la media, el número de temperaturas que superan la media y el número de temperaturas que están por debajo. Utiliza los módulos que consideres adecuados.

Solución 1:

```
#include <iostream>
using namespace std;
const int KTEMP=24;

void leeTemp(int temp[]);
void calcula (int temp[], float &media, int &encima, int &debajo);

int main(){
    int temp[KTEMP];
    int encima, debajo;
    float media;

    leeTemp(temp);
    calcula(temp, media, encima, debajo);
    cout << "La media es " << media << endl;
    cout << "Hay " << encima << " temperaturas que la superan" << endl;
    cout << "Hay " << debajo << " temperaturas por debajo" << endl;
    return 0;
}

void leeTemp(int temp[]){
    int i;

    for (i=0; i<KTEMP; i++){
        cout << "Introduce la temperatura " << i+1 << ": ";
        cin >> temp[i];
    }
}

void calcula (int temp[], float &media, int &encima, int &debajo){
    int i;

    media=0.0;
    encima=0;
    debajo=0;
    for (i=0; i<KTEMP; i++)
        media=media+temp[i];
    media=media/KTEMP;
    for (i=0; i<KTEMP; i++){
        if (temp[i]<media)
            debajo++;
        else if (temp[i]>media)
            encima++;
    }
}
```

Solución 2:

```
#include <iostream>
using namespace std;
const int KTEMP=24;

void leeTemp(float []);
```

```

float tempMedia (float []);
void calcula (float [], float, float &, float &);

int main(){
    float temp[KTEMP];
    float encima, debajo;
    float media;

    leeTemp(temp);
    media = tempMedia(temp);
    calcula(temp, media, encima, debajo);
    cout << "La media es " << media << endl;
    cout << "Hay " << encima << " temperaturas que la superan" << endl;
    cout << "Hay " << debajo << " temperaturas por debajo" << endl;
}

void leeTemp(float temp[]){
    int i;

    for (i=0; i<KTEMP; i++){
        cout << "Introduce la temperatura " << i+1 << ": ";
        cin >> temp[i];
    }
}

float tempMedia (float temp[]) {
    int i;
    float media;

    media=0.0;
    for (i=0; i<KTEMP; i++)
        media=media+temp[i];
    media=media/KTEMP;

    return(media);
}

void calcula (float temp[], float media, float &encima, float &debajo){
    int i;

    encima=0;
    debajo=0;

    for (i=0; i<KTEMP; i++){
        if (temp[i]<media)
            debajo++;
        else if (temp[i]>media)
            encima++;
    }
}

```

- 2) Cuando se examinan grandes volúmenes de datos, una técnica para evaluar el conjunto de los datos es el histograma, que contiene la frecuencia de aparición de cada valor en el conjunto de datos.

Por ejemplo, dada la siguiente lista de valores:

3 7 4 2 3 6 7 8 2 3 9

La lista contiene 2 doses, 3 treses, 1 cuatro, 1 seis, 2 sietes, 1 ocho y 1 nueve. Por lo tanto, el histograma sería el siguiente:

1: 0
2: 2
3: 3
4: 1
5: 0
6: 1
7: 2
8: 1
9: 1
10: 0

De acuerdo con esto, realiza un programa en C que genere un histograma de una lista de 100 valores, donde cada valor está comprendido en el intervalo del 1 al 10. Para ello previamente deberá almacenar los datos y posteriormente deberá mostrarlos.

Nota: podemos generar los distintos valores mediante la función `rand()` .

```
#include <iostream>
#include <stdlib.h>
using namespace std;

const int KMAX=100;

void generaLista(int []);
void calcHistograma(int [], int []);
void iniciaHistograma(int []);
void muestraHistograma(int []);

int main(){
    int lista[KMAX];
    int histo[10];

    srand(time(NULL));
    generaLista(lista);
    iniciaHistograma(histo);
    calcHistograma(lista, histo);
    muestraHistograma(histo);
    return 0;
}

void generaLista(int lista[]){
    int i;

    for (i=0;i<KMAX;i++)
        lista[i]=rand()%10+1;
}

void muestraHistograma(int histo[]){
    int i;

    for (i=0; i<10; i++)
        cout << i+1 << ": " << histo[i]<< endl;
}
```

```

void iniciaHistograma(int histo[]){
    int i;

    for (i=0; i<10; i++)
        histo[i]=0;
}

void calcHistograma(int lista[], int histo[]){
    int i;

    //Los valores oscilan entre 1 y 10, mientras que los índices
    //del array varían entre 0 y 9
    for (i=0; i<KMAX; i++)
        histo[lista[i]-1]++;
}

```

3) Escribe un programa que compruebe si una matriz es simétrica.

```

#include <iostream>
using namespace std;
const int KTAM=5;

void leeMat(int[][KTAM]);
bool simetrica(int[][KTAM]);

int main(){
    int mat[KTAM][KTAM];

    leeMat(mat);
    if (simetrica(mat))
        cout << "La matriz es simétrica" << endl;
    else
        cout << "La matriz no es simétrica" << endl;
    return 0;
}

void leeMat(int mat[][KTAM]){
    int i, j;

    for (i=0; i<KTAM; i++)
        for (j=0; j<KTAM; j++){
            cout << "Introduce el valor [" << i << "][" << j << "]: ";
            cin >> mat[i][j];
        }
}

```

```
//Para comprobar que es simétrica sólo hay que recorrer
//la triangular superior y comparar cada elemento con el
//correspondiente
bool simetrica(int mat[][KTAM]){
    int i, j;
    bool test;

    test=true;
    i=0;
    while (i<KTAM-1 && test){
        j=i+1;
        while (j<KTAM && test){
            if (mat[i][j]!=mat[j][i])
                test=false;
            j++;
        }
        i=i+1;
    }
    return (test);
}
```