INSTITUTO TECNOLÓGICO DE CIUDAD MADERO.



UNIDAD 2

PRACTICA 2\_2

ALUMNA: CAVAZOS ARGOT ANA VICTORIA

N° CONTROL: 15071292

PROFESOR: DRA. CLAUDIA GUADALUPE GÓMEZ SANTILLÁN

MATERIA: PROGRAMACIÓN PARALELA

FECHA DE ENTREGA: 12 DE OCTUBRE 2018

Índice:

[Ejercicio: 3](#_Toc527157263)

[Introducción: 3](#_Toc527157264)

[Marco teórico: 4](#_Toc527157265)

[Media: 4](#_Toc527157266)

[Moda: 4](#_Toc527157267)

[Desviación estándar: 4](#_Toc527157268)

[Varianza: 4](#_Toc527157269)

[Números pares e impares: 4](#_Toc527157270)

[Números primos: 4](#_Toc527157271)

[Test de primalidad: 5](#_Toc527157272)

[Números amigos: 5](#_Toc527157273)

[Metodología: 6](#_Toc527157274)

[Experimentación y resultados: 8](#_Toc527157275)

[Conclusiones: 10](#_Toc527157276)

[Bibliografía: 10](#_Toc527157277)

Ejercicio:

Introducción:

Transcriba el siguiente programa y agregue los siguientes módulos en funciones independientes: Calcule a) la media, b) la moda, c) la desviación estándar, d) Cuantos valores son pares e impares, e) cuales valores son primos, f) Cuales valores son amigos, g) la varianza y h) sume todos los elementos del arreglo.

**#include** <stdio.h>

**#include** <stdlib.h>

**#include** <math.h>

**#define** RANDMAX 10000

**int** **main**(**void**) {

**int** num=100, i=0, j=0, aux;

**long** **int** numeros[num];

**puts**("!!!Hello Students!!!");

**puts**("METODO DE LA BURBUJA");

//puts("Cuantos Numeros quieres:");

//scanf("%d",&num);

// LLENAR ARREGLO CON VALORES ALEATORIOS

**for** (i=0; i < num; i++)

{

numeros[i]= **rand**();

**printf**("%d, ",numeros[i]);

}

//ORDENAR

**for** (i=0; i<num;i++)

**for** (j=i+1;j<num;j++)

{

**if** (numeros[j] < numeros[i])

{aux = numeros[i];

numeros[i] = numeros[j];

numeros[j]=aux;}

}

**printf** ("\n");

**for** (i=0; i < num; i++)

**printf**("%d, ",numeros[i]);

**printf** ("\n");

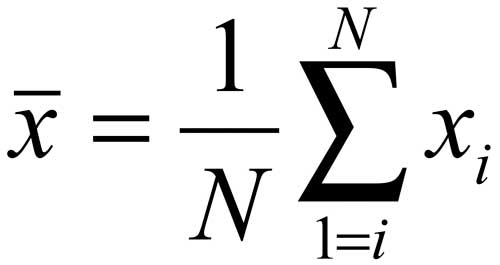
**printf** ("El elemento mas pequeño es: %d ", numeros[0]);

**printf** ("\n El elemento mas grande es: %d", numeros[num-1]);

**return** EXIT\_SUCCESS;

}

Marco teórico:

Media:

La media de un conjunto de números, algunas ocasiones simplemente llamada el promedio, es la suma de los datos dividida entre el número total de datos.

Moda:

La moda de un conjunto de datos es el dato que más veces se repite, es decir, aquel que tiene mayor frecuencia absoluta. Se denota por Mo. En caso de existir dos valores de la variable que tengan la mayor frecuencia absoluta, habría dos modas. Si no se repite ningún valor, no existe moda.

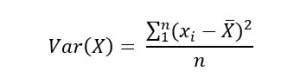
Desviación estándar:

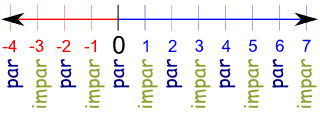
La desviación típica o desviación estándar (denotada con el símbolo σ o s, dependiendo de la procedencia del conjunto de datos) es una medida de dispersión para variables de razón (variables cuantitativas o cantidades racionales) y de intervalo. Se define como la raíz cuadrada de la varianza de la variable.

$ \sigma =\sqrt{\frac{1}{N} \sum_{i=1}^N (x_i-\mu)^2$

La desviación estándar es la raíz cuadrada de la varianza. Es una de las medidas de dispersión, una medida que es indicativa de como los valores individuales pueden diferir de la media.

Varianza:

La unidad de medida de la varianza será siempre la unidad de medida correspondiente a los datos, pero elevada al cuadrado. La varianza siempre es mayor o igual que cero. Al elevarse los residuos al cuadrado es matemáticamente imposible que la varianza salga negativa. Y de esa forma no puede ser menor que cero.

Números pares e impares:

En las matemáticas y especialmente en la aritmética, un número par es un número entero que es divisible entre dos. ​ Se trata de un número entero que se puede escribir de la forma: 2k (es decir, divisible de manera entera entre 2), donde k es un entero (los números pares son los múltiplos del número 2).

Los números enteros que no son pares se llaman números impares (o números menores), y pueden escribirse como 2k+1.2​

Números primos:

Un número primo es un número entero mayor que cero, que tiene exactamente dos divisores positivos. También podemos definirlo como aquel número entero positivo que no puede expresarse como producto de dos números enteros positivos más pequeños que él, o bien, como producto de dos enteros positivos de más de una forma. Conviene observar que con cualquiera de las dos definiciones el 1 queda excluido del conjunto de los números primos.

Ejemplos:

* El 7 es primo. Sus únicos divisores son 1 y 7. Sólo puede expresarse como producto de 7·1.
* El 15 no es primo. Sus divisores son 1, 3, 5 y 15. Puede expresarse como 3·5. (y también como 15·1)

Los 25 primeros números primos son 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89 y 97, que son todos los primos menores que 100.

Test de primalidad:

La cuestión de la determinación de si un número n dado es primo es conocida como el problema de la primalidad. Un test de primalidad (o chequeo de primalidad) es un algoritmo que, dado un número de entrada n, no consigue verificar la hipótesis de un teorema cuya conclusión es que n es compuesto.

Pseudocodigo:

**function** is\_prime(n)

**if** n ≤ 1

**return** false

**else if** n ≤ 3

**return** true

**else if** n mod 2 = 0 **or** n mod 3 = 0

**return** false

**let** i ← 5

**while** i \* i ≤ n

**if** n mod i = 0 **or** n mod (i + 2) = 0

**return** false

i ← i + 6

**return** true

Números amigos:

Dos números amigos son dos números enteros positivos a y b tales que la suma de los divisores propios de uno es igual al otro número y viceversa, es decir σ(a)=b y σ(b)=a, donde σ(n) es igual a la suma de los divisores de n, sin incluir a n. (La unidad se considera divisor propio, pero no lo es el mismo número.).

Un ejemplo es el par de naturales (220, 284), ya que:

* los divisores propios de 220 son 1, 2, 4, 5, 10, 11, 20, 22, 44, 55 y 110, que suman 284;
* los divisores propios de 284 son 1, 2, 4, 71 y 142, que suman 220.

Si un número es amigo de sí mismo (es igual a la suma de sus divisores propios), recibe entonces el nombre de número perfecto.

Primeros números amigos:

(220,284), (1184,1210),(2620,2924)

Metodología:

Conjunto de datos:  
**Datos:** 2, 8, 5, 1, 10, 5, 9, 9, 3, 5, 6, 6, 2, 8, 2, 2, 6, 3, 8, 7,

**Datos ordenados:** 1, 2, 2, 2, 2, 3, 3, 5, 5, 5, 6, 6, 6, 7, 8, 8, 8, 9, 9, 10,

**a) Media**:

Formula:

Media = (1 + 2 + 2 + 2 + 2 + 3 + 3 + 5 + 5 + 5 + 6 + 6 + 6 + 7 + 8 + 8 + 8 + 9 + 9 + 10)/20 = 107/20 = 5.35

**b) Moda**:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Datos | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Total |
| Repeticiones | 1 | 4 | 2 | 0 | 3 | 3 | 1 | 3 | 2 | 1 | 20 |

Valor que más se repite: 2

Moda = 2

**c) Desviación estándar**:

Formula:

Desviación estándar =

Sqrt(((1-5.35)^2 + (2-5.35)^2 + (2-5.35)^2 + (2-5.35)^2 + (2-5.35)^2 +

(3-5.35)^2 + (3-5.35)^2 + (5-5.35)^2 + (5-5.35)^2 + (5-5.35)^2 +

(6-5.35)^2 + (6-5.35)^2 + (6-5.35)^2 + (7-5.35)^2 + (8-5.35)^2 +

(8-5.35)^2 + (8-5.35)^2 + (9-5.35)^2 + (9-5.35)^2 + (10-5.35)^2) / 20) =

Sqrt((148.550003)/20) = 2.7253

**d) ¿Cuántos valores son pares e impares?**:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Datos** | 1 | 2 | 3 | 5 | 6 | 7 | 8 | 9 | 10 |
| **Par/Impar** | impar | par | impar | impar | par | impar | par | impar | par |
| **Repeticiones** | 1 | 4 | 2 | 3 | 3 | 1 | 3 | 2 | 1 |

Pares: 4 + 3 + 3 + 1 = 11

Impares= 1 + 2 + 3 + 1 + 2 = 9

Total = 11 + 9 = 20

**e) ¿Cuáles valores son primos?**:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Datos** | 1 | 2 | 3 | 5 | 6 | 7 | 8 | 9 | 10 |
| **Primo** | No | Si | Si | Si | No | Si | No | No | No |
| **Repeticiones** | 1 | 4 | 2 | 3 | 3 | 1 | 3 | 2 | 1 |

**Test de primalidad:**

**n = 1**

function is\_prime(n)

**if n ≤ 1**

**return false**

else if n ≤ 3

return true

else if n mod 2 = 0 or n mod 3 = 0

return false

let i ← 5

while i \* i ≤ n

if n mod i = 0 or n mod (i + 2) = 0

return false

i ← i + 6

return true

**n = 5**

function is\_prime(n)

if n ≤ 1

return false

else if n ≤ 3

return true

else if n mod 2 = 0 or n mod 3 = 0

return false

let i ← 5

while i \* i ≤ n

if n mod i = 0 or n mod (i + 2) = 0

return false

i ← i + 6

**return true**

**n = 8**

function is\_prime(n)

if n ≤ 1

return false

else if n ≤ 3

return true

else if **n mod 2 = 0** or n mod 3 = 0

**return false**

let i ← 5

while i \* i ≤ n

if n mod i = 0 or n mod (i + 2) = 0

return false

i ← i + 6

return true

En cada caso de prueba se resaltó el valor, la condición que cumplió y el resultado obtenido.

**f) ¿Cuáles valores son amigos?**:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Datos** | 1 | 2 | 3 | 5 | 6 | 7 | 8 | 9 | 10 |
| **Amigos** | No | No | No | No | No | No | No | No | No |
| **Repeticiones** | 1 | 4 | 2 | 3 | 3 | 1 | 3 | 2 | 1 |

Datos: 9 y 10

Divisores de 9:

9%1=0 Divisor

9%2=1

9%3=0 Divisor

9%4=1

9%8=1

Divisores de 10:

10%1=0 Divisor

10%2=0 Divisor

10%4=0 Divisor

Suma de divisores 9: 1 + 3 = 4

Suma de divisores 10: 1 + 2 + 5 = 8

4 es distinto que 10

8 es distinto que 9

No son amigos

**g) Varianza**:

Formula:

Desviación estándar =

((1-5.35)^2 + (2-5.35)^2 + (2-5.35)^2 + (2-5.35)^2 + (2-5.35)^2 +

(3-5.35)^2 + (3-5.35)^2 + (5-5.35)^2 + (5-5.35)^2 + (5-5.35)^2 +

(6-5.35)^2 + (6-5.35)^2 + (6-5.35)^2 + (7-5.35)^2 + (8-5.35)^2 +

(8-5.35)^2 + (8-5.35)^2 + (9-5.35)^2 + (9-5.35)^2 + (10-5.35)^2) / 20 =

(148.550003/20) = 7.4275

**h) Sume todos los elementos del arreglo**:

Formula:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Datos | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Repeticiones | 1 | 4 | 2 | 0 | 3 | 3 | 1 | 3 | 2 | 1 |

Suma = 1 + 2 + 2 + 2 + 2 + 3 + 3 + 5 + 5 + 5 + 6 + 6 + 6 + 7 + 8 + 8 + 8 + 9 + 9 + 10 = 107

Experimentación y resultados:

Información sobre el equipo:

**Modelo**: Dell OptiPlex 7010

**Procesador**: Intel(R) Core(TM) i5-3550 CPU @ 3.30GHz

**Memoria RAM**: 4.00 GB

**Tipo de sistema**: Sistema operativo de 64 bits

**Sistema operativo utilizado**: Windows 7 Ultimate Service Pack 1

Tabla de resultados:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Experimento | SEED | Datos | Menor | Mayor | Suma | Media | Moda | Varianza | Desviación estándar | Par | Impar | Primos | Amigos | Tiempo  Secuencial (segundos) | Hilos | Tiempo paralelo OMP (segundos) |
| 1 | 20 | 500 | 42 | 32610 | 8025276 | 16050.5 | 6002 | 9638.23 | 92895568 | 252 | 248 | 103 … | - | **23.192** | **2** | 9.303 |
| **4** | 3.133 |
| **6** | 1.879 |
| **8** | 1.444 |
| **10** | 1.142 |
| **20** | 0.512 |

Información sobre el clúster:

**Modelo**:

**CPU**: Intel(R) Xeon(R) CPU E5-2650 v3 @ 2.30GHz

**PCI:** Xeon E7 v3/Xeon E5 v3/Core i7 DMI2

Tabla de resultados:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Experimento | SEED | Datos | Menor | Mayor | Suma | Media | Moda | Varianza | Desviación estándar | Par | Impar | Primos | Amigos | Tiempo  Secuencial (segundos) | Hilos | Tiempo paralelo OMP (segundos) |
| 1 | 20 | 500 | 3 | 9995 | 2401682 | 4803.363770 | 1335 | 8911026 | 2985.134277 | 238 | 262 | 3. 41, 73, …,9967 | - | **8.26** | **2** | 4.09 |
| **4** | 2.23 |
| **6** | 1.58 |
| **8** | 1.18 |
| **10** | 1.01 |
| **20** | 0.75 |
| 2 | 20 | 5000 | 1 | 9998 | 24741026 | 4948.205078 | 490 | 2899.73999 | 8408492 | 2383 | 2425 | 3,17,29…,9967 | 284 Y 220 | **+60** | **2** | +60 |
| **8** | +60 |
| **20** | 64.18 |
| **40** | 37.46 |
| **80** | 19.94 |
| **100** | 15.8 |
| 3 | 20 | 10000 | 1 | 9998 | 47519056 | 4751.905762 | 3782 | 8327573.5 | 2885.753662 | 4906 | 4921 | 3,17,…,9967 | 1184 Y 1210 | **+60** | **200** | 31.83 |
| **400** | 16.27 |
| **600** | 10.99 |

En el caso de los resultados de los valores primos no es posible mostrar todos los resultados generados, por lo que solo se muestra el inicial.

Conclusiones:

Esta práctica permitió observar de mejor manera las ventajas de la paralelizacion. En este caso al aumentar la cantidad de hilos disponibles permitió utilizar conjuntos de datos grandes y realizar el proceso en un menor tiempo que de manera secuencial.

Durante esta práctica se realizó un paralelismo mixto, es decir, se dividió el conjunto de datos entre los hilos disponibles. Estos hilos se encargan de calcular la suma, contabilizar los números pares e impares y contabilizar los números primos. Después se realiza un paralelismo de funciones, donde cada hilo toma una sección disponible que no dependen una de otra.

Además, en algunas funciones que involucran ciclos que acumulan datos se decidió paralelizarlos para agilizar el conteo que realizan mediante un reduction.

Bibliografía:

<https://es.wikipedia.org/wiki/Desviaci%C3%B3n_t%C3%ADpica>

<http://www.alcula.com/es/calculadoras/estadistica/desviacion-estandar/>

<http://economipedia.com/definiciones/varianza.html>

<https://www.varsitytutors.com/hotmath/hotmath_help/spanish/topics/mean-median-mode>

<https://www.portaleducativo.net/octavo-basico/790/Media-moda-mediana-rango>

<http://mimosa.pntic.mec.es/jgomez53/matema/conocer/primos.htm>

[www.vaxasoftware.com/doc\_edu/mat/numamigos\_esp.pdf](http://www.vaxasoftware.com/doc_edu/mat/numamigos_esp.pdf)

<https://es.wikipedia.org/wiki/N%C3%BAmeros_amigos>

<https://en.wikipedia.org/wiki/Primality_test>

<http://diagramas-de-flujo.blogspot.com/2013/01/determinar-si-dos-numeros-son-amigos-en-Cpp.html>