Imagen que contiene dibujo, señal, alimentos

Descripción generada automáticamenteINSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

ANALISIS DE ALGORITMOS

PROFESOR TITULAR: FRANCO MARTINEZ EDGARDO ADRIAN

1° PARCIAL

EJERCICIOS #3:

SIMULACION PRODUCTO2MAYORES

LEMUS RUIZ MARIANA ELIZABETH

2020630211

GRUPO: 3CM12

Un dibujo de una mujer

Descripción generada automáticamente con confianza media

**EJERCICIOS 03: SIMULACIÓN PRODUCTO2MAYORES**

**INSTRUCCIONES:**

Para el algoritmo analizado por casos en clase y video lección "Producto2Mayores", realice la simulación de su mejor, peor y caso medio; realizando las modificaciones y adaptaciones necesarias para verificar los tres casos en n=2500 y n=5000 considerando al menos 10,000 iteraciones del algoritmo con cada n y diferente distribución de los números.

* Para el mejor caso basta con tener un archivo de números que coloque en los dos primeros números a los dos mayores
* Para el peor caso basta con tener un archivo ordenado ascendentemente para cada n
* Para el caso medio se deberán de hacer al menos 10,000 iteraciones para cada n generando arreglos aleatorios en cada iteración y comprobar el número de operaciones básicas promedios totales para enfrentarlas al modelo del caso medio.

**PSEUDOCODIGO:**

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

**MEJOR CASO**

|  |
| --- |
| F(n) = 1+ (n - 2) + (n – 2) + 2  = 3 + 2(n – 2)  = 2n - 1 |

CODIGO DEL PROGRAMA:

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <string.h>  #include <math.h>  #define VECES 10000  int Producto2Mayores**(**int **\***A**,** int n**)**  **{**  int mayor1**,** mayor2**,** i**,** contador **=** 0**;**  contador**++;** //Condicion  **if** **(**A**[**0**]** **>** A**[**1**]){**  mayor1 **=** A**[**0**];**  contador**++;** //Asignacion  mayor2 **=** A**[**1**];**  contador**++;** //Asignaciom  **}**  **else{**  mayor1 **=** A**[**1**];**  mayor2 **=** A**[**0**];**  **}**  i **=** 2**;**  **while** **(**i **<** n**){**  contador**++;** //Condicion  **if** **(**A**[**i**]** **>** mayor1**){**  mayor2 **=** mayor1**;**  mayor1 **=** A**[**i**];**  **}**  **else{**  contador**++;** //Condicion  **if** **(**A**[**i**]** **>** mayor2**){**  mayor2 **=** A**[**i**];**  **}}**  i**++;**  **}**  **return** contador**;**  //return mayor1\*mayor2  **}**  int main**(**int narg**,** char **\*\***varg**)**  **{**  int n**;**  int **\***A**;**  double promedio **=** 0**;**  int auxiliar**;**  **if** **(**narg **!=** 2**)**  **{**  printf**(**"\nIntroduce una n"**);**  scanf**(**"%d"**,&**n**);**  exit**(**1**);**  **}**  n **=** atoi**(**varg**[**1**]);**  A **=** malloc**(sizeof(**int**)** **\*** n**);**  **for** **(**int i **=** 0**;** i **<** n**;** i**++)**  **{**  scanf**(**"%d"**,** **&**A**[**i**]);**  **}**  auxiliar **=** Producto2Mayores**(**A**,** n**);**  promedio **=** promedio **+** **(**float**)**auxiliar**;**  printf**(**"El promedio final es %lf"**,** promedio**);**  **return** 0**;**  **}** |

**PEOR CASO**

|  |
| --- |
| F(n) = 1 + (n - 2) + 2(n - 2) + 2  = 3+ 3(n - 2)  =3n - 3 |

CODIGO DEL PROGRAMA:

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <string.h>  int Producto2Mayores**(**int **\***A**,** int n**){**  int mayor1**,** mayor2**,** i**,** contador **=** 0**;**  contador**++;** //Condicion  **if** **(**A**[**0**]** **>** A**[**1**])** **{**  mayor1 **=** A**[**0**];**  contador**++;** //Asignacion  mayor2 **=** A**[**1**];**  contador**++;** //Asignaciom  **}**  **else{**  mayor1 **=** A**[**1**];** //Asignacion  contador**++;**  mayor2 **=** A**[**0**];**  contador**++;** //Asignacion  **}**  i **=** 2**;**  **while** **(**i **<** n**)** **{**  contador**++;** //Condicion  **if** **(**A**[**i**]** **>** mayor1**)** **{**  mayor2 **=** mayor1**;**  contador**++;** //Asignacion  mayor1 **=** A**[**i**];**  contador**++;** //Asignacion  **}**  **else{**  **if** **(**A**[**i**]** **>** mayor2**){**  mayor2 **=** A**[**i**];**  **}}**  i**++;}**  **return** contador**;**  **}**  int main**(**int narg**,** char **\*\***varg**)**  **{**  int n**;**  int **\***A**;**  double promedio **=** 0**;**  int auxiliar**;**  **if** **(**narg **!=** 2**)**  **{**  /\*printf("\nIntroduce una n");  exit(1);\*/  **}**  n **=** atoi**(**varg**[**1**]);**  A **=** malloc**(sizeof(**int**)** **\*** n**);**  **for** **(**int i **=** 0**;** i **<** n**;** i**++)**  **{**  scanf**(**"%d"**,** **&**A**[**i**]);**  **}**  auxiliar **=** Producto2Mayores**(**A**,** n**);**  promedio **=** promedio **+** **(**float**)**auxiliar**;**  printf**(**"Promedio final: %lf"**,** promedio**);**  **return** 0**;**  **}** |

**CASO MEDIO**

|  |
| --- |
| PARA: A[i] > MAYOR 1  ⅓(1+2+(n-2)+2(n-2) = ⅓ (3n -3)  PARA: A[i] ≤ MAYOR1 && A[i] >MAYOR2  ⅓(1+2+2(n-2)+(n-2)= ⅓(3n-3)  PARA: A[i] ≤ MAYOR1 && A[i] ≤ MAYOR2  ⅓(1+2+(n-2)+(n-2)) = ⅓ (2n-1)  CASO MEDIO  F(n) = ⅓ (2(3n – 3) + (2n – 1)) = ⅓(8n – 7) |

CODIGO DEL PROGRAMA:

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <string.h>  #define VECES 10000  int Producto2Mayores**(**int **\***A**,** int n**)**  **{**  int mayor1**,** mayor2**,** i**,** contador **=** 0**;**  contador**++;** //Condicion  **if** **(**A**[**0**]** **>** A**[**1**])**  **{**  mayor1 **=** A**[**0**];**  contador**++;** //Asignacion  mayor2 **=** A**[**1**];**  contador**++;** //Asignaciom  **}**  **else**  **{**  mayor1 **=** A**[**1**];**  contador**++;** //Asignacion  mayor2 **=** A**[**0**];**  contador**++;** //Asignacion  **}**  i **=** 2**;**  **while** **(**i **<** n**)**  **{**  contador**++;** //Condicion  **if** **(**A**[**i**]** **>** mayor1**)**  **{**  mayor2 **=** mayor1**;**  contador**++;** //Asignacion  mayor1 **=** A**[**i**];**  contador**++;** //Asignacion  **}**  **else**  **{**  contador**++;** //Condicion  **if** **(**A**[**i**]** **>** mayor2**)**  **{**  mayor2 **=** A**[**i**];**  contador**++;** //Asignacion  **}**  **}**  i**++;**  **}**  **return** contador**;**  **}**  int main**(**int narg**,** char **\*\***varg**)**  **{**  int n**,**i**,**k**;**  int **\***A**;**  int promedio **=** 0**;**  int auxiliar**;**  **if** **(**narg **!=** 2**)**  **{**  printf**(**"\nIntroduce una n"**);**  exit**(**1**);**  **}**  n **=** atoi**(**varg**[**1**]);**  A **=** malloc**(sizeof(**int**)** **\*** n**);**  srand**(**time**(NULL));**  **for** **(** k **=** 0**;** k **<** VECES**;** k**++)**  **{**  **for** **(**i **=** 0**;** i **<** n**;** i**++)**  **{**  **\*(**A **+** i**)** **=** rand**()** **%** **((**1000 **\*** n**));**  //printf("Vuelta %d. Numero %d\n",k,A[i]);  **}**  auxiliar **=** Producto2Mayores**(**A**,** n**);**  promedio **=** promedio **+** **(**float**)**auxiliar**;**  //printf("Iteracion %d, el contador vale %d\n",k,auxiliar);  **}**  //printf("El promedio es %lf\n", promedio);  promedio **=** promedio **/** VECES**;**  printf**(**"El promedio es %d"**,** promedio**);**  **return** 0**;**  **}** |

TABLA COMPARATIVA DE RESULTADOS

|  |  |  |
| --- | --- | --- |
|  | PRACTICO | TEORICO |
| PARA n= 2500 | | |
| MEJOR CASO | 4997.0000 | 4999.0000 |
| PEOR CASO | 7497.0000 | 7497.0000 |
| CASO MEDIO | 5012.0000 | 6664.3333 |
| PARA n= 5000 | | |
| MEJOR CASO | 9997.0000 | 9999.00000 |
| PEOR CASO | 14997.0000 | 14997.00000 |
| CASO MEDIO | 10014.0000 | 13331.0000 |