

Programación Orientada a Objetos 1

Práctica Calificada 2

Pregrado 2021-I

Profesor: Rubén Rivas Medina

Lab. 106

Indicaciones específicas:

- Esta evaluación contiene 8 páginas (incluyendo esta página) con 3 preguntas. El total de puntos son 20.
- El tiempo límite para la evaluación es 100 minutos.
- Cada pregunta deberá ser respondida en un solo archivo con el número de la pregunta. Por ejemplo:
 - 1. p1.cpp y p1.h
 - 2. p2.cpp y p2.h
 - 3. p3.cpp y p3.h
- Deberás subir estos archivos directamente a www.gradescope.com, o se puede crear un .zip que contenga todos ellos y subirlo.

Competencias y criterios de desempeño:

• Para los alumnos de la carrera de Ciencia de la Computación

Aplica conocimientos de computación apropiados para la solución de problemas definidos y sus requerimientos en la disciplina del programa.(nivel 2)

Diseña, implementa y evalúa soluciones a problemas complejos de computación.(nivel 2)

Crea, selecciona, adapta y aplica técnicas, recursos y herramientas modernas para la práctica de la computación y comprende sus limitaciones.(nivel 2)

• Para los alumnos de las carreras de Ingeniería

Capacidad para aplicar conocimientos de matemática.(nivel 2)

Capacidad para diseñar un sistema, un componente o un proceso para satisfacer las necesidades deseadas dentro de restricciones realistas(nivel 2)

Calificación:

Tabla de puntos (sólo para uso del professor)

Question	Points	Score
1	7	
2	7	
3	6	
Total:	20	

- 1. (7 points) Escribir y diseñar una función ($obtener_indices$) que retorne un vector de enteros y que permita leer un arreglo dinámico con valores enteros de tamaño (n) y un carácter (opcion) que defina las siguientes opciones:
 - p par
 - i impar
 - r primo

```
vector<int> obtener_indices(int* arreglo,int n,char opcion);
```

La función deberá retornar un vector que contenga con los subíndices de los valores de acuerdo al parámetro opción, por ejemplo: si se elige la opción \mathbf{p} debera generar un vector con todos los subíndices de los valores pares. si se elige la opción \mathbf{i} debera generar un vector con todos los subíndices de los valores impar y si se elige la opción \mathbf{r} debera generar un vector con todos los subíndices de los valores primos. Algunos ejemplos:

Ejemplo #1 Input Format

```
10
1 2 10 7 6 5 11 8 4 14
p
```

Output Format

```
1 2 4 7 8 9
```

Ejemplo #2

Input Format

```
6
1 2 10 7 6 5
i
```

Output Format

```
0 3 5
```

Ejemplo #3

Input Format

```
12
1 2 10 7 6 5 11 31 27 2 1 9
r
```

Output Format

```
1 3 5 6 7 9
```

La rúbrica para esta pregunta es:

Criterio	Excelente	Adecuado	Mínimo	Insuficiente
Algoritmo y	Elabora un al-	Elabora un al-	Elabora un al-	Elabora un al-
codificación	goritmo preciso,	goritmo preciso,	goritmo preciso,	goritmo preciso,
(4 pts)	definido y finito	definido y finito	definido y finito	definido y finito
	que da solución	que da solución	que da solución	que hace menos
	exacta a lo que	al menos al 80	al menos al 65	del 65 % de lo
	el enunciado re-	% de lo que	% de lo que	que el enunciado
	quiere. Utiliza	el enunciado re-	el enunciado	requiere. Utiliza
	arrays dinámicos	quiere. Utiliza	requiere. Utiliza	arrays dinámicos
	y hace un buen	arrays dinámicos	arrays dinámicos	y hace un buen
	uso de memoria:	y hace un buen	y hace un buen	uso de memoria:
	dimensiona, usa	uso de memoria:	uso de memoria:	dimensiona, usa
	y libera memoria	dimensiona, usa	dimensiona, usa	y libera memoria
	de manera ade-	y libera memoria	y libera memo-	de manera ade-
	cuada al codi-	de manera ade-	ria de manera	cuada al codi-
	ficar el algoritmo	cuada al codi-	adecuada al	ficar el algoritmo
	y lo hace con	ficar el algoritmo	codificar el al-	y lo hace con
	el 100% de pre-	y lo hace con	goritmo y lo	menos el 65% de
	cisión. $(4pts)$	al menos el 80%	hace con al	precisión. (0
		de precisión. (menos el 65%	pts)
		3pts)	de precisión.	
Sintaxis y	El algoritmo es	El algoritmo es	(2pts) El algoritmo es	El algoritmo
Sintaxis y legibilidad	El algoritmo es correcto, y es	El algoritmo es correcto, y es	correcto, y es	es incorrecto o
(1 pt)	codificado sin	codificado con	codificado con	es codificado
(1 pt)	errores de sin-	algunos errores	algunos errores	con errores de
	taxis. El nombre	de sintaxis,	de sintaxis,	sintaxis, que
	de las variables	pero que no	que afectan el	afectan el resul-
	y funciones son	afectan el resul-	resultado de	tado de manera
	descriptivas.	tado de manera	manera mínima,	significativa.
	(1pts)	significativa.	o el nombre de	El nombre de
	(1 ***)	El nombre de	las variables y	las variables y
		las variables y	funciones no	funciones no
		funciones son	son descriptivas.	son descriptivas.
		descriptivas.	$(0.5 \mathrm{pts})$	(0pts)
		$(0.75 \mathrm{pts})$	/	
Optimización	El código es	El código es	El código es	El código es re-
de código	óptimo y efi-	óptimo en al	óptimo en al	dundante y/o no
(2 pt)	ciente (2pts)	menos el 80%	menos 65%	es $\acute{\mathrm{optimo}}(\mathtt{0pts})$
		(1.5pts)	(1pts)	

2. (7 points) Escribir y diseñar una función ($generar_matriz_organizada$) que retorne una matriz cuadrada de enteros de lado (n) y que permita leer un arreglo dinámico con valores enteros de tamaño ($n \cdot n$).

```
int** generar_matriz_organizada(int* arreglo, int n);
```

La función debe tomar cada valor del arreglo y calcular su posición en la matriz usando las formulas: (i = valor/n) y (j = valor%n) y ubicar el valor en la posición correspondiente.

La matriz inicialmente debe tener sus valores en CERO.

Si la posición estuviera fuera del rango de posiciones válidas de la matriz entonces el valor se descarta.

Si anteriormente otro valor ocupará esa posición el valor deberá ser sobreescrito.

Algunos ejemplos:

Ejemplo #1

Input Format

```
3
1 2 5 7 6 5 4 8 10
```

Output Format

```
0 1 2
0 4 5
6 7 8
```

Ejemplo #2

Input Format

```
2
1 2 6 5
```

Output Format

```
0 1
2 0
```

Ejemplo #3

Input Format

```
4
1 2 10 7 6 5 11 15 10 3 1 9 13 20 14 4
```

Output Format

```
0 1 2 3
4 5 6 7
0 9 10 11
0 13 14 15
```

La rúbrica para esta pregunta es:

Criterio	Excelente	Adecuado	Mínimo	Insuficiente
Algoritmo y	Elabora un al-	Elabora un al-	Elabora un al-	Elabora un al-
codificación	goritmo preciso,	goritmo preciso,	goritmo preciso,	goritmo preciso,
(4 pts)	definido y finito	definido y finito	definido y finito	definido y finito
	que da solución	que da solución	que da solución	que hace menos
	exacta a lo que	al menos al 80	al menos al 65	del 65 % de lo
	el enunciado	% de lo que	% de lo que	que el enunci-
	requiere. Uti-	el enunciado	el enunciado	ado requiere.
	liza matrices	requiere. Uti-	requiere. Uti-	Utiliza matrices
	dinámicas y	liza matrices	liza matrices	dinámicas y
	hace un buen	dinámicas y	dinámicas y	hace un buen
	uso de memoria:	hace un buen	hace un buen	uso de memoria:
	dimensiona, usa	uso de memoria:	uso de memoria:	dimensiona, usa
	y libera memo-	dimensiona, usa	dimensiona, usa	y libera memo-
	ria de manera	y libera memo-	y libera memo-	ria de manera
	adecuada al	ria de manera	ria de manera	adecuada al
	codificar el algo-	adecuada al	adecuada al	codificar el
	ritmo y lo hace	codificar el al-	codificar el al-	algoritmo y
	con el 100%	goritmo y lo	goritmo y lo	lo hace con
	de precisión.	hace con al	hace con al	menos el 65%
	(4pts)	menos el 80%	menos el 65%	de precisión. (0
		de precisión. (de precisión.	$ ext{pts})$
Sintaxis y	El algoritmo es	3pts) El algoritmo es	(2pts) El algoritmo es	El algoritmo
Sintaxis y legibilidad	correcto, y es	El algoritmo es correcto, y es	correcto, y es	es incorrecto o
(1 pt)	codificado sin	codificado con	codificado con	es meorrecto o es codificado
(1 pt)	errores de sin-	algunos errores	algunos errores	con errores de
	taxis. El nombre	de sintaxis,	de sintaxis,	sintaxis, que
	de las variables	pero que no	que afectan el	afectan el resul-
	y funciones son	afectan el resul-	resultado de	tado de manera
	descriptivas.	tado de manera	manera mínima,	significativa.
	(1pts)	significativa.	o el nombre de	El nombre de
	,	El nombre de	las variables y	las variables y
		las variables y	funciones no	funciones no
		funciones son	son descriptivas.	son descriptivas.
		descriptivas.	$(0.5 \mathrm{pts})$	(0pts)
		$(0.75 \mathrm{pts})$		
Optimización	El código es	El código es	El código es	El código es re-
de código	óptimo y efi-	óptimo en al	óptimo en al	dundante y/o no
(2 pt)	ciente (2pts)	menos el 80%	menos 65%	es $optimo(0pts)$
		(1.5 pts)	(1pts)	

3. (6 points) Escribir un programa que lea (n) valores enteros y que los almacene en un vector, modificar el vector para que solo guarde en forma ordenada los valores sin repetirlos.

Algunos ejemplos: Ejemplo #1

Input Format

8 1 2 5 7 8 5 4 8

Output Format

1 2 4 5 7 8

Ejemplo #2

Input Format

20 1 2 6 5 4 12 31 20 11 2 6 3 4 5 11 59 21 22 50 11

Output Format

1 2 3 4 5 6 11 12 20 21 22 31 50 59

Ejemplo #3

Input Format

16 1 2 10 7 6 5 11 15 10 3 1 9 13 20 14 4

Output Format

1 2 3 4 5 6 7 9 10 11 13 14 15 20

La rúbrica para esta pregunta es:

Criterio	Excelente	Adecuado	Mínimo	Insuficiente
Algoritmo y	Elabora un al-	Elabora un al-	Elabora un al-	Elabora un al-
codificación	goritmo preciso,	goritmo preciso,	goritmo preciso,	goritmo preciso,
(3 pts)	definido y finito	definido y finito	definido y finito	definido y finito
	que da solución	que da solución	que da solución	que hace menos
	exacta a lo que	al menos al 80	al menos al 65	del 65 % de lo
	el enunciado	% de lo que	% de lo que	que el enunciado
	requiere. Uti-	el enunciado	el enunciado	requiere. Uti-
	liza vectores y	requiere. Uti-	requiere. Uti-	liza vectores y
	sus métodos al	liza vectores y	liza vectores	sus métodos
	codificar el algo-	sus métodos	al codificar el	al codificar el
	ritmo y lo hace	al codificar el	algoritmo y	algoritmo y
	con el $100%$	algoritmo y	lo hace con al	lo hace con
	de precisión.	lo hace con al	menos el 65%	menos el 65%
	(4pts)	menos el 80%	de precisión.	de precisión. (0
		de precisión. ((1pts)	pts)
		2pts)		
Sintaxis y	El algoritmo es	El algoritmo es	El algoritmo es	El algoritmo
legibilidad	correcto, y es	correcto, y es	correcto, y es	es incorrecto o
(1 pt)	codificado sin	codificado con	codificado con	es codificado
	errores de sin-	algunos errores	algunos errores	con errores de
	taxis. El nombre	de sintaxis,	de sintaxis,	sintaxis, que
	de las variables	pero que no	que afectan el	afectan el resul-
	y funciones son	afectan el resul-	resultado de	tado de manera
	descriptivas.	tado de manera	manera mínima,	significativa.
	(1pts)	significativa.	o el nombre de	El nombre de
		El nombre de	las variables y	las variables y
		las variables y	funciones no	funciones no
		funciones son	son descriptivas.	son descriptivas.
		descriptivas.	$(0.5 \mathrm{pts})$	(0pts)
04::	Tal -4-1*	(0.75pts)	Tel -4-11	Tel = 4 di = 0 = 0
Optimización		El código es	El código es	El código es re-
de código	óptimo y efi-	óptimo en al	óptimo en al	dundante y/o no
(2 pt)	ciente $(2pts)$	menos el 80%	menos 65%	es óptimo $(0pts)$
		(1.5pts)	(1pts)	