

# 1-Definición:

$A[20] = 3, 4, 8, 20, 16, 14, 5, 6, 7, 8, 9, 12, 14, 15, 9, 14, 2, 10, 11, 12$

$$SP = 4 + 20 + 14 + 6 + 8 + 12 + 16 + 14 + 2$$

$$Si = 3 + 8 + 16 + 5 + 7 + 9 + 14 + 9 + 2 + 10$$

$$PMP = 20$$

## 2- Analisis:

```

E — [ Pseudo ] — Switch
    for(i=0, i<20, i=i+1)
        A[i] ← rand() % 21
        if((i+1) % 2 == 0)
            SP ← SP + A[i]
        else
            Si ← Si + A[i]
        if(PMP < A[i])
            PMP ← A[i]
    
```

## 3- Algoritmo:

Variable	Dominio	Tipo	de datos
i	0 a 19	int	Variable de control del ciclo for
A[20]	0 a 20	array	Arreglo con 20 datos
SP	0 a 200	int	Suma de los pares en la posición
Si	0 a 200	int	Suma de los impares en la posición
PMP	0 a 20	int	Variable que almacena el número más grande de los pares

```

1  Proceso ejercicio
2      Definir i, sp, ssi, pmp Como Entero
3          Dimension A[20]
4          sp ← 0
5          ssi ← 0
6          pmp ← 0
7          Para i ← 1 Hasta 20 Con Paso 1
8              A[i] ← Aleatorio(0,20)
9              Si (i Mod 2) = 0 Entonces
10                 sp ← sp + A[i]
11                 Si pmp < A[i] Entonces
12                     pmp ← A[i]
13                 FinSi
14             SiNo
15                 ssi ← ssi + A[i]
16             FinSi
17             Escribir A[i], " "
18         FinPara
19         Escribir "a) ", sp
20         Escribir "b) ", ssi
21         Escribir "c) ", pmp
22 FinProceso

```

PSelnt - Ejecutando proceso EJERCICIO

```

*** Ejecución Iniciada. ***
15
1
8
14
4
20
20
6
8
4
6
2
13
9
11
4
9
0
2
3
a) 63
b) 96
c) 20
*** Ejecución Finalizada. ***

```

En la prueba de escritorio nunca va a salir mal por que no se ingresan valores entonces siempre va a salir algo bueno siempre y cuando el código fue bien realizado

## 1-Definición:

$A[n] \rightarrow \{5, 6, 7, 14, 5, 4, 2, 8, 9, 16, 20, 17\}$

$$Sp = 6 + 14 + 4 + 2 + 8 + 16 + 20$$

$$Si = 5 + 7 + 5 + 9 + 17$$

$$St = Sp + Si$$

## 2-Análisis

E — Proceso

n  
 $for(i=0, i < n, i++)$   
 $if(A[i] \% 2 == 0)$   
 $Sp = Sp + A[i]$   
 else  
 $Si = Si + A[i]$

$$St = Sp + Si$$

— salida

$A[n]$

Sp

Si

St

## 3-Algoritmo

Diccionario de datos		
Variable	Tipo	Característica
n	int	cantidad de elementos
i	int	variable de control del ciclo for
Sp	int	suma de los números pares
Si	int	suma de los números impares
St	int	suma total
$A[n]$	array	Arreglo con los números



```

1  Proceso ejercicio2
2      Definir n, i, sp, ssi, st Como Entero
3          sp ← 0
4          ssi ← 0
5          st ← 0
6      Escribir "Ingresa cantidad de elementos 1-30: "
7      Leer n
8      Si  $n \geq 1$  Y  $n \leq 30$  Entonces
9          Dimension A[n]
10         Para i ← 1 Hasta n Con Paso 1
11             A[i] ← Aleatorio(0,20)
12             Si (A[i] Mod 2) = 0 Entonces
13                 sp ← sp + A[i]
14             SiNo
15                 ssi ← ssi + A[i]
16             FinSi
17             Escribir A[i], " "
18         FinPara
19         st ← sp + ssi
20         Escribir "a) ", sp
21         Escribir "b) ", ssi
22         Escribir "c) ", st
23     SiNo
24         Escribir "Cantidad invalida"
25     FinSi
26 FinProceso

```

PSelnt - Ejecutando proceso EJERCICIO2

\*\*\* Ejecución Iniciada. \*\*\*

Ingresa cantidad de elementos 1-30:

> 22

17

15

5

16

3

5

16

7

10

1

12

0

0

20

13

4

5

PSelnt - Ejecutando proceso EJERCICIO2

\*\*\* Ejecución Iniciada. \*\*\*

Ingresa cantidad de elementos 1-30:

> 0

Cantidad invalida

\*\*\* Ejecución Finalizada. \*\*\*

a) 90

b) 110

c) 200

\*\*\* Ejecución Finalizada. \*\*\*

PSelnt - Ejecutando proceso EJERCICIO2

\*\*\* Ejecución Iniciada. \*\*\*

Ingresa cantidad de elementos 1-30:

> 34

Cantidad invalida

\*\*\* Ejecución Finalizada. \*\*\*

1- Definición

A[20] = {10, 15, 30, 11, 150, 110, 60, 70, 10, 22, 122, 131, 1, 2, 3, 199, 138, 131, 20, 7, 9}

L1[C1] = {10, 15, 30, 11, 10, 22, 1, 2, 3}

L2[C2] = {60, 70, 70}

L3[C3] = {150, 110, 122, 131, 149, 138, 131}

2- Analisis

E - Pseudocódigo

```
for(i=0, i<20, i=i+1)
  A[i] = rand()%151
  if(A[i]>=0 AND A[i]<=50)
    L1[C1] = A[i]
    C1 = C1 + 1
  else if(A[i]>=50 AND A[i]<=100)
    L2[C2] = A[i]
    C2 = C2 + 1
  else
    L3[C3] = A[i]
    C3 = C3 + 1
```

A[20]

L1[C1]

L2[C2]

L3[C3]

### 3- Algoritmo

#### Diccionario de datos

Variable	Tipo	comentario
A[20]	int	Arreglo original
L1[20]	>	Lista 1 del rango 0-50
L2[20]	>	Lista 2 del rango 51-100
L3[20]	>	Lista 3 del rango 101-150
i	int	variable de control del for
c1	int	contador de terminos de lista 1
c2	int	contador de terminos de lista 2
c3	int	contador de terminos de lista 3

```
1  Proceso ejercicio3
2      Dimensionar A[20], L1[20], L2[20], L3[20]
3      Definir i, c1, c2, c3 Como Entero
4      c1 ← 1
5      c2 ← 1
6      c3 ← 1
7      Escribir "Lista original:"
8      Para i ← 1 Hasta 20 Con Paso 1
9          A[i] ← Aleatorio(0,151)
10         Si A[i] ≥ 0 Y A[i] ≤ 50 Entonces
11             L1[c1] ← A[i]
12             c1 ← c1 + 1
13         SiNo
14             Si A[i] ≥ 51 Y A[i] ≤ 100 Entonces
15                 L2[c2] ← A[i]
16                 c2 ← c2 + 1
17             SiNo
18                 L3[c3] ← A[i]
19                 c3 ← c3 + 1
20         FinSi
21     FinSi
22     Escribir A[i], " "
23 FinPara
24 Escribir "a) "
25 Para i ← 1 Hasta c1 Con Paso 1
26     Escribir L1[i], " "
27 FinPara
28 Escribir "b) "
29 Para i ← 1 Hasta c2 Con Paso 1
30     Escribir L2[i], " "
31 FinPara
32 Escribir "c) "
33 Para i ← 1 Hasta c3 Con Paso 1
34     Escribir L3[i], " "
35 FinPara
36 FinProceso
37
```

PSelnt - Ejecutando proceso EJERCICIO3

\*\*\* Ejecución Iniciada. \*\*\*

Lista original:

63

129

144

15

79

57

95

149

145

145

41

6

139

1

106

56

3

147

123

109

a)

15

41

6

1

3

0

b)

63

79

57

95

56

0

c)

129

144

149

145

145

139

106

147

123

109

0

\*\*\* Ejecución Finalizada. \*\*\*



Como no se ingresan datos por el usuario lo único malo que tendría al hacer la prueba de escritorio es que si el algoritmo esta mal hecho entonces tendría errores en la prueba de escritorio