

Unidad 6: Punteros

Laboratorio – Sesión 2

Profesores:

< Juan Flores Moroco >

Telegram:

1. Configurar tu cuenta

2. Link: <http://bit.ly/2OW5Ss9>

SISAP:



SISAP 2018
October 7-9
Lima, Perú

Evento: SISAP 2018 – 11th International Conference on Similarity Search and Applications

Fechas: October 7-9 Lima, Perú

Resumen: <http://www.sisap.org/2018/>

The 11th International Conference on Similarity Search and Applications (SISAP) is an annual forum for researchers and application developers in the area of similarity data management. It aims at the technological problems shared by numerous application domains, such as data mining, information retrieval, multimedia, computer vision, pattern recognition, computational biology, geography, biometrics, machine learning, and many others that make use of similarity search as a necessary supporting service.

Inscripciones: https://eventos.spc.org.pe/spire2018/registration_sisap.html

SPIRE:

SPIRE 2018: 25th International Symposium on String Processing and Information Retrieval

Fechas: October 9-11 Lima, Perú

Resumen: <https://eventos.spc.org.pe/spire2018/venue.html>

SPIRE 2018 is the 25th edition of the annual Symposium on String Processing and Information Retrieval. SPIRE has its origins in the South American Workshop on String Processing, which was first held in Belo Horizonte, Brazil, in 1993. Since 1998 the focus of the workshop has also included information retrieval, due to its increasing relevance to and inter-relationship with string processing.

SPIRE 2018 will be held in UTEC Lima, Peru.

Inscripciones: <https://eventos.spc.org.pe/spire2018/registration.html>

Logro de la sesión:

Al finalizar la sesión, los alumnos desarrollan sus programas utilizando punteros y arrays dinámicos.

Punteros



Arreglos dinámicos

Array dinámicos:

```
int *pia = new int[10];    // bloque de 10 unidades de int
```

```
int *pia2 = new int[10] (); // bloque de 10 int inicializados con cero
```

```
int *pia3 = new int[10] {0,1,2,3,4,5,6,7,8,9};
```

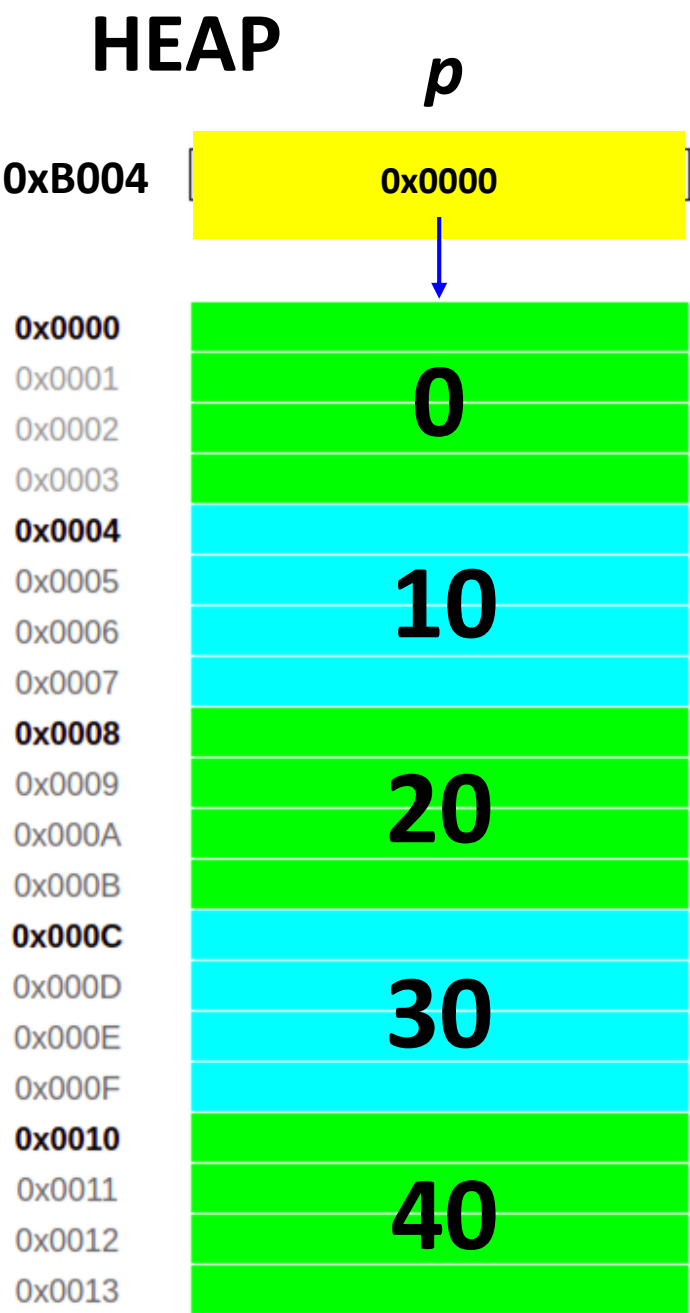
¿

Para liberar el espacio de memoria:

```
delete [] pia;
```


Se crea un array de 5 elementos en el heap:

```
int *p;  
  
p = new int[5];  
  
for(size_t i = 0; i < 5 ;i++)  
    p[i] = i*10;
```



Ejemplo 1:

Realice un programa que permita leer como dato un número que representa la cantidad de elementos que tendrá un array dinámico.

Luego realice lo siguiente:

- **Crear el array, llenarlo con números aleatorios entre 0 y 999.**
- **Imprimir el array**
- **Generar a partir de ese array dos nuevos array dinámicos, el primero con los múltiplos de 5 y el segundo con los múltiplos de 7 que tenga el primer array.**

```
1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4
5  void Imprimir(int *pArreglo, size_t n);
6
7  int main()
8  { size_t numDatos;
9    int *pA,*pMultiplosde5, *pMultiplosde7;
10   size_t contMulde5=0, contMulde7=0;
11
12   srand(time(nullptr));
13   cout <<"Numero de elementos : ";
14   cin >> numDatos;
15
16   pA = new int[numDatos];
17   for(size_t i=0; i<numDatos; i++)
18   {
19       pA[i] = rand()%1000;
20       if(pA[i]%5==0)
21           contMulde5++;
22       if(pA[i]%7==0)
23           contMulde7++;
24   }
```

```
25 //-- se crea los otros dos arrays
26 pMultiplode5 = new int[contMulde5];
27 pMultiplosde7= new int[contMulde7];
28 //-- se llenen los datos segun sea el caso
29 size_t indiceMul5=0, indiceMul7=0;
30 for(size_t i=0; i<numDatos; i++)
31 {
32     if(pA[i]%5==0)
33         pMultiplode5[indiceMul5++]=pA[i];
34     if(pA[i]%7==0)
35         pMultiplosde7[indiceMul7++]=pA[i];
36 }
37 cout<<"\n\nArray con datos\n";
38 Imprimir(pA,numDatos);
39 cout<<"\nArray con los multiplos de 5\n";
40 Imprimir(pMultiplode5,contMulde5);
41 cout<<"\nArray con los multiplos de 7\n";
42 Imprimir(pMultiplosde7,contMulde7);
43 delete [] pA;
44 delete [] pMultiplode5;
45 delete [] pMultiplosde7;
46 return 0;
47 }
```

```

48
49  ↔ void Imprimir(int *pArreglo, size_t n)
50    {
51        for(size_t i=0; i<n; i++)
52            cout << setw(5) << pArreglo[i];
53    }
54

```

Numero de elementos : 25

Array con datos

117 37 215 334 643 638 138 987 861 638 93 567 397 142 72 866 537 941 263 951 240 882 612 344 503

Array con los multiplos de 5

215 240

Array con los multiplos de 7

987 861 567 882

Ejemplo 2:

Desarrollar un programa que permita generar aleatoriamente un array de números enteros, el tamaño del array es un dato que ingresará el usuario.

Luego el programa imprimirá el array y pedirá al usuario que ingrese el número que desea buscar en el array.

El programa luego tendrá que indicar cuantas veces está el dato a buscar en el array y además creará un segundo array con las posiciones en que el dato se encuentra en el array.

Si el dato no está en el array, entonces se imprimirá solo el mensaje que indica que el dato no se encuentra en el array.