



Práctica 3

Sección: A1

Ciclo: 2016.2

1. [5 ptos.] Utilice punteros para completar y programar el prototipo:

```
1      int verificar (int * dado, int m, int n);
```

El cual verifica si en un arreglo de m elementos apuntado por *parr*, contiene un cierto número n ; en tal caso retorna un mensaje indicando que contiene dicho número, caso contrario, retorna un mensaje indicando que no contiene dicho número. En la función principal, se define un arreglo y su apuntador, asigna valores aleatorios entre 10 y 19 a sus elementos y asigna (aleatoriamente) un valor a n comprendido entre 10 y 19. Finalmente, verifica si n está en el arreglo.

2. [5 ptos.] Implemente un programa que almacene los enteros $2, 3, \dots, 2016$ en un arreglo numérico. Luego, utilizando la [criba de Eratóstenes](#) para este arreglo (método que encuentra los números primos dentro de este arreglo de la siguiente manera:

1. saltando el elemento $p = 2$ del arreglo, asigna 0 a todos los elementos múltiplos de 2 del arreglo,
2. saltando el elemento $p = 3$ del arreglo, asigna 0 a todos los elementos múltiplos de 3 del arreglo,
3. este proceso se detiene cuando $p * p > 2016$)

imprima los primos encontrados en el arreglo y la dirección de memoria en la que se encuentra. Por ejemplo:

2 está en la dirección 0x...0

3 está en la dirección 0x...a

...

3. [5 ptos.] Cree un programa que genere aleatoriamente una cadena de 20 letras minúsculas (ascii entre 97 y 122). Luego, reporte dos veces la cadena: una en el orden de su creación y otra en el orden inverso a su creación.

4. [5 ptos.] Cree un programa que genere aleatoriamente una cadena de 20 letras mayúsculas (ascii entre 65 y 90). Luego, reporte dos veces la cadena: una en el orden de su creación y otra luego de ordenar ascendentemente la cadena.

Martes, 8 de noviembre del 2016