



Tercera Práctica Calificada: 12/11/2018

CC112-AB

Ciclo: 2018-2

Normas:

1. El alumno entregará esta hoja de examen debidamente llenada con sus datos.
2. La solución de la prueba se guardarán en **Escritorio**, carpeta: **ApellidoNombreCodigo** (sin espacios en blanco), la pregunta **n** se guardará en el archivo: **n.c** ($n = 1, 2, \dots$).
3. No se permite: El uso de celulares, internet, USB, ingresar después de 15 min. de iniciado el examen ni salir antes de la hora de finalización.
4. Todo acto anti-ético será amonestado y registrado en el historial del estudiante.

Apellidos : _____ Nombres : _____
Sección : _____ Grupo: _____

1. [5 ptos.] Escriba y pruebe una función que reciba una referencia a un arreglo de enteros formado por valores diferentes de cero, salvo el último término y retorne el mayor de los elementos de dicho arreglo. Para ello debe completar en los puntos suspensivos:

```
#include <stdio.h>
int maximo(int * p);
int main(){
    int edades[] = {22,19,17,23,18,0};
    printf(" edad máxima: %d\n", maximo(edades));
}
int maximo(int * p){/*...*/;}
```

Para el ejemplo, La salida debe ser:
edad máxima: 23

2. [5 ptos.] Escriba una función tal que al pasarle una referencia a un arreglo de enteros formado enteros diferentes de cero, salvo el último término, muestre todos los elementos distintos de cero de dicho arreglo pero en sentido contrario. También escriba otra función tal que al pasarle una referencia a un arreglo de enteros formado enteros diferentes de cero, salvo el último término, retorne la cantidad de elementos distintos de cero de dicho arreglo. Para ello debe completar en los puntos suspensivos:

```
#include <stdio.h>
void contrario(int * p);
int longitud(int * p);
int main() {
    int numeros[4] = {-1,-3,-2,0};
    printf(" cantidad de numeros:%3d\n", longitud(numeros));
    contrario(numeros);
}
int longitud(int * p){/*...*/;}
```

void contrario(int * p){/*...*/;}

Para el ejemplo, La salida debe ser:
cantidad de numeros: 3
-2 -3 -1

3. [5 ptos.] En el auditorio de la FC se dicta una conferencia, los asistentes anotan sus nombres en una lista, al final se rifa (número aleatorio de 1 a número de asistentes), un premio y se anuncia el número y nombre del ganador. Escriba un programa para controlar el

ingreso.

Suponga que los asistentes son:

```
char lista[30][10] = {"Juan", "María", "José", "Elena", "Fiore", "Rosita"}, nl= 0;
```

Cálculo de nl = número de asistentes:

```
for(i=0;i<10;i++) if(strlen(lista[i])) nl++ ;
```

La salida puede ser:

El ganador es: 3 José

4. [5. ptos.] Escriba un programa que valide (verifique) el formato de un correo electrónico: `nnnnn@ssss.ttt`, donde:

elementos de `nnnnn` ≥ 3

elementos de `ssss` ≥ 6

elementos de `ttt` ≥ 3

por ejemplo para: CBazanC5@hotmail.com, la salida debe ser:

Correo correcto: cbazanc5@hotmail.com

Sugerencia: complete el programa en `/* ... */`

```
#include<cstdio>
#include<ctype.h>
#include<cstring>
main(){
    char cc[30]= "CBazanC5@hotmail.com";
    int nn, narr=0, ss, npun=0, notro=0, n=strlen(cc), i=0;
    /*
        nn    = número de elementos alfanumericos en nnnnn
        narr  = número de arrobas (@)
        ss    = numero de elementos en nnnnn@ssss
        npun  = número de puntos (.)
        notro = número de elementos distintos de '@' y '.'
        n     = longitud del correo: nnnnn@ssss.ttt
    */
    for(;i<n ; i++) {
        cc[i] = tolower(cc[i]);
        if(!isalnum(cc[i])) {           // ¿es alfanumérico?
            /* ... */ // calcule narr, nn, npun, ss, notro
        }
        if(notro==0 && narr==1 && npun==1 && nn>=3 && ss-nn>=7
            && n-ss>=4) printf(/* ... */);    // correcto
        else
            printf(/* ... */);
    }
}
```