

<p align="center"><b>Universidad Tecnológica Nacional</b> <b>Facultad Regional Avellaneda</b></p>										
<p align="center">Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos</p>										
<p>Materia: <b>Programación I</b></p>										
Apellido:					Fecha:					
Nombre:					Docente <sup>(2)</sup> :					
División:					Nota <sup>(2)</sup> :					
Legajo:					Firma <sup>(2)</sup> :					
Instancia <sup>(1)</sup> :	<b>PP</b>		<b>RPP</b>		<b>SP</b>		<b>RSP</b>		<b>FIN</b>	<b>X</b>

(1) Las instancias validas son: 1<sup>er</sup> Parcial (**PP**), Recuperatorio 1<sup>er</sup> Parcial (**RPP**), 2<sup>do</sup> Parcial (**SP**), Recuperatorio 2<sup>do</sup> Parcial (**RSP**), Final (**FIN**). Marque con una cruz.

(2) Campos a ser completados por el docente.

### En Lenguaje ANSI C:

- Dadas las siguientes afirmaciones. Indique si son V o F. Justifique:
  - (0.5 pto.)** A una función siempre hay que pasarle parámetros.
  - (0.5 pto.)** Una función puede devolver más de un valor.
- Explicar:
  - (0.5 pto.)** ¿Qué es un puntero? Declare uno.
  - (0.5 pto.)** ¿Cuáles son y para que se utilizan sus operadores?
- Explicar:
  - (0.5 pto.)** ¿Qué son parámetros formales y parámetros actuales?
  - (0.5 pto.)** ¿En una función a que se denomina pasaje por valor y pasaje por referencia?

- (1 pto.)** Sea el siguiente bucle for:

```
for (i=0; i<=100; i+=3) printf("%d\n", i);
```

¿Cuál de los siguientes bucles while es equivalente?

- `while (i<=100)`  
`{ i=0; printf("%d\n", i); i+=3; }`
- `i=0;`  
`while (i<=100) { printf("%d\n", i); i+=3; }`
- `i=0;`  
`while (i<=100) { i+=3; printf("%d\n", i); }`
- `while (i=0 && i<=100)`  
`{ printf("%d\n", i); i+=3; }`

- Se tiene:

```
int vec[10], *p;  
p=vec;
```

Escribir su equivalente en notación de array:

- (0.5 pto.)** `p+i` \_\_\_\_\_.
- (0.5 pto.)** `*(p+i)` \_\_\_\_\_.

6. **(1 pto.)** Explicar el error en el siguiente programa:

```
void main(){
    int vector[10];
    int x = 1;
    for(x=1; x<=10; x++)
        vector[x] = 23;
}
```

7. **(1 pto.)** Crear la siguiente jerarquía de estructuras:

- struct nota {calificación (entero)}
- struct parcial {p\_examen (struct nota); rec\_examen (struct nota)}
- struct materia {nombre (cadena,15); p\_parcial(struct parcial);s\_parcial (struct parcial)}
- struct alumno {apellido (cadena,30); legajo(entero); materias(struct materia)}

8. **(1 pto.)** Realizar la apertura, lectura (listar los datos) y cierre de un archivo.

9. **(1 pto.)** Se tiene:

```
typedef struct {
    int legajo;
    char nombre [25];
    float sueldo;
}Empleado;
Empleado personal [14];
```

Realizar prototipo y desarrollo de una función que ordene el vector personal alfabéticamente.

10. **(1 pto.)** Desarrollar una función llamada mostrarEmpleado que reciba por referencia una estructura del tipo del punto anterior y que muestre por pantalla el contenido de sus campos. (Se deben realizar el prototipo, la llamada, y el desarrollo).