

Ejercicio Pre Parcial Informática I

Ing. Ignacio Bonelli, Ing. Héctor Levi, Leandro Kollenberger

Sábado 4 de Julio de 2015

Apellido y nombres:

Legajo:

1. ¿Qué es un vector? ¿Cuándo lo usaría? ¿Cómo se declara? ¿Cómo se accede a sus elementos?
2. ¿Es posible cambiar la dirección asignada a un vector previamente declarado?
3. ¿Qué es un puntero? ¿Se puede asignar la dirección de comienzo de un vector a un puntero del mismo tipo?
4. Describa brevemente la estructura de control de flujo *switch*. ¿En qué casos la usaría?
5. Dada la siguiente variable declarada en un programa: `char linea[20]`; Especifique el tipo, tamaño, valor y contenido de lo apuntado (si corresponde) por las siguientes expresiones:

- (a) `*linea;`
- (b) `linea;`
- (c) `&linea;`

6. Dado el siguiente prototipo de una función:

```
unsigned int strlen(char *str);
```

y el siguiente código:

```
unsigned int longitud;  
char str[] = "Hola, me llamo Ernesto y soy una cadena de caracteres."  
longitud = strlen(&str);
```

¿Está correctamente invocada la función `strlen()`? Justificar, y en caso de requerir cambios en la línea que la invoca, especifique cuales deberían ser.

7. Suponga un programa que comience con la siguiente línea:

```
#include <stdio.h>
```

- (a) ¿Qué función cumple la misma?
- (b) ¿Qué contiene el archivo `stdio.h`?
- (c) ¿Dónde está alojado el código binario de la función `printf()`?

8. Indique cuantos *bytes* ocupan las siguientes expresiones en memoria:

- (a) `'a'`;
- (b) `"a"`;

Justifique.

9. Dadas las siguientes cadenas de caracteres:

```
char *p = "Hola!";  
char *q = "Hola!";
```

Indique lo que imprime por pantalla el siguiente fragmento de código:

```
if(p == q) {
    printf("Son iguales!!\n");
} else {
    printf("Son diferentes!!\n");
}
```

Justifique su respuesta.

10. ¿Cuántos *bytes* ocupa un puntero a *double*? ¿Y cuántos un puntero a función? ¿Y uno a *char*? Justifique.
11. Dado el siguiente archivo fuente llamado `sumar.c`:

```
#include <stdio.h>
// Programa para sumar dos numeros como argumentos
int main(int argc, char *argv[]) {
    float a, b, resultado;
    if(argc == 1) { // si hay un solo argumento, el usuario olvido los valores
        fprintf(stderr, "Ingrese los valores a sumar!\n");
        return -1;
    }
    sscanf(argv[1], "%d", &a);
    sscanf(argv[2], "%d", &b);
    resultado = a + b;
    printf("El resultado es: %d\n", resultado);
    return 0;
}
```

Responda las siguientes preguntas:

- (a) ¿Compilará exitosamente? ¿Arrojará errores? Indique cuales si corresponde, y las correcciones al código que realizaría para tal fin.
 - (b) Al compilar, ¿arrojará *warnings*? Indíquelos si corresponde, y las correcciones que realizaría para que no imprima ninguna.
 - (c) Indique el comando que debería escribir para compilar este programa, habilitando todos los *warnings* posibles y generando un archivo ejecutable llamado `programa.exe` en la carpeta padre a la actual, por ejemplo, si el archivo `sumar.c` está ubicado en el directorio `/home/usuario/sumar/` y actualmente la línea de comandos está en ese directorio, compile de tal forma que el archivo de salida sea `/home/usuario/programa.exe`.
 - (d) Indique los posibles problemas que tenga este programa al ejecutarse, y los cambios al código que realizaría para que funcione correctamente.
 - (e) ¿Qué es `stderr`? ¿Por qué se lo usa en este programa?
12. ¿Cómo le eliminaría los permisos de ejecución al programa compilado en el ejercicio anterior?
 13. Indique los posibles problemas que pueda tener el siguiente fragmento de código:

```
FILE *archivo;
char buffer[100];
archivo = fopen("entrada.txt", "r");
fgets(buffer, 100, archivo);
```

¿Cómo los resolvería?

14. Linux, ¿es un sistema operativo? ¿Por qué?
15. ¿Qué es GCC? ¿Y GDB?
16. ¿Qué es un intérprete de comandos? Nombre alguno.