

Informática II

Practica No. 2

Ejercicio 1. Codifique el programa que se muestra a continuación:

```
#include <iostream> using
namespace std;

void fun_a(int *px, int *py); void fun_b(int a[],
int tam);

int main()
{
    int array[10] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}; fun_b(array, 10);
}

void fun_a(int *px, int *py){ int tmp = *px;
    *px = *py;
    *py = tmp;
}

void fun_b(int a[], int tam){ int f, l; int *b = a; for (f = 0, l = tam - 1; f
< l; f++, l--) { fun_a(&b[f], &b[l]);
    }
}
```

Con ayuda del debugger, examine la representación en memoria del arreglo array y responda las siguientes preguntas.

- ¿Cuál es su dirección en memoria? ¿Cuántos bytes se dedican para almacenar cada elemento de array?
@0x61fdf0 y se dedican 4 bytes para almacenar cada elemento, ya que son tipo int.
- ¿Cuál es la dirección y el contenido en memoria del elemento array[3] ?
La dirección es 0x61fdfc y el contenido es 3.
- Describa el efecto que tiene la función fun_b, sobre el arreglo array.
La función fun_b tiene dos punteros que guardan la dirección de dos posiciones del array, una que va de forma ascendente y otro descendente y los entrega a la función fun_a.

Ejercicio 2. La función que se ilustra a continuación calcula el promedio de los elementos de un arreglo de tamaño n. promedio y suma apuntan a variables que la función modifica por referencia. Desafortunadamente, la función contiene errores; encuétralos y corrígelos, de tal manera que su operación sea correcta.

```
void fun_c(double *a, int n, double *promedio, double *suma){ int i; suma = 0.0; for (i =
    0; i < n; i++) suma += (a + i);

    promedio = suma / n;
}
```

Implemente un caso de prueba para la función anterior, que le permita verificar su correcto funcionamiento.

Ejercicio 3. Se tiene la siguiente declaración e inicialización para el arreglo b:

```
unsigned short b[4][2] = {{77, 50}, {5, 2}, {28, 39}, {99, 3}};
```

La Figura 1, que se presenta a continuación, ilustra la representación del arreglo b en memoria. Complete la numeración de las direcciones de memoria, en notación hexadecimal, para cada elemento del arreglo b.

| b[0] | | b[1] | | b[2] | | b[3] | |
|----------|----------|----------|----------|----------|----------|----------|----------|
| b[0][0] | b[0][1] | b[1][0] | b[1][1] | b[2][0] | b[2][1] | b[3][0] | b[3][1] |
| 0x61fd70 | 0x61fd72 | 0x61fd74 | 0x61fd76 | 0x61fd78 | 0x61fd7a | 0x61fd7c | 0x61fd7e |

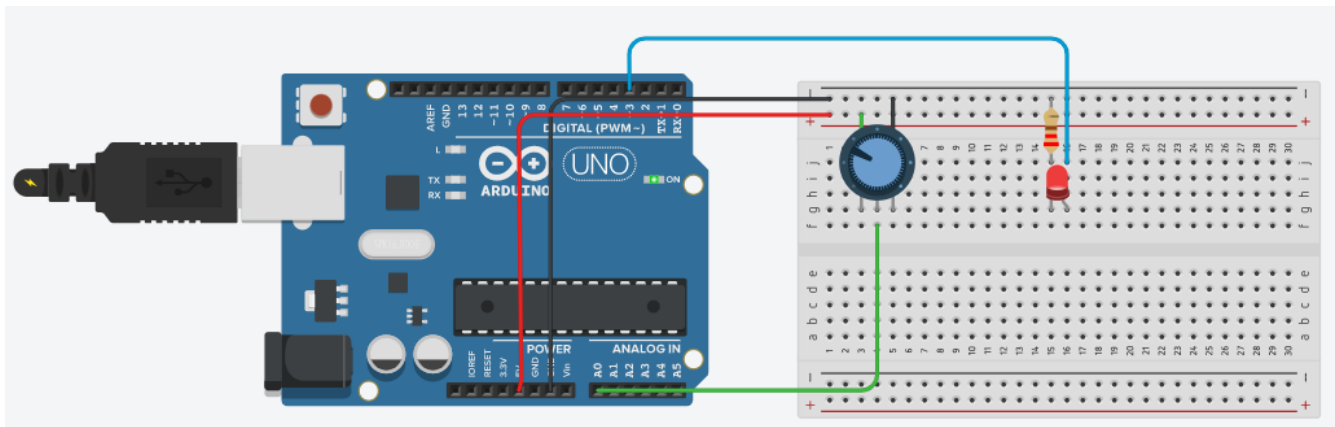
Figura 1: Elementos del arreglo b.

Determine de acuerdo a lo ilustrado en la Figura 1 el valor correspondiente de las siguientes expresiones:

- b : 0x61fd70
- b+2: 0x61fd78
- *(b+2): 0x61fd78
- *(b+2)+1: 0x61fd7a
- (*(b+2)+1): 39
- b[3][1]: 3
- *b++: Error

Ejercicios con Arduino

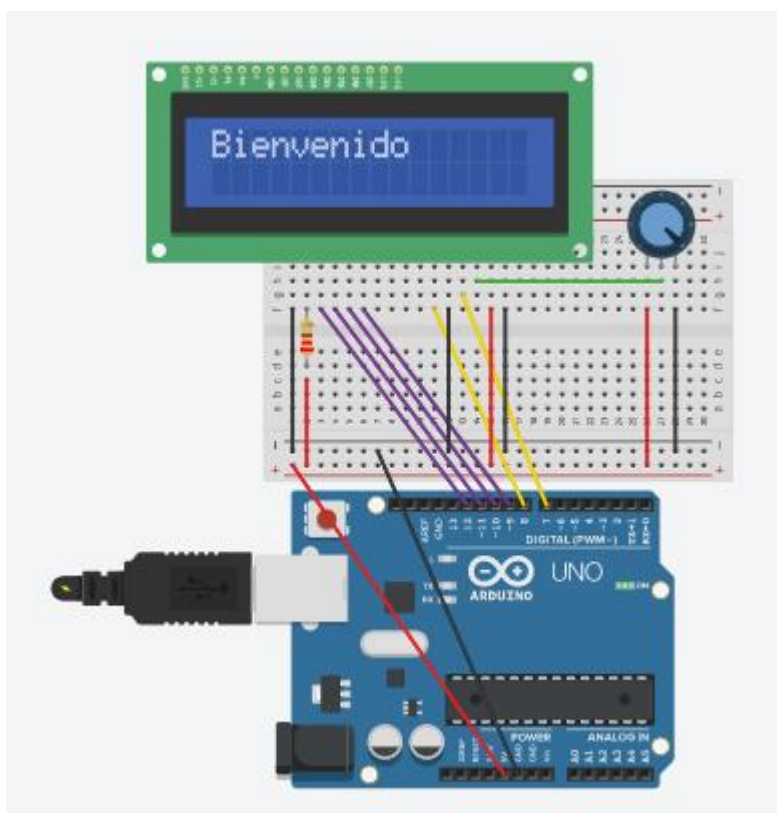
Ejercicio 4. Realice un circuito en Tinkercad donde la intensidad de iluminación de un LED se pueda variar con un potenciómetro. Debe agregar en Tinkercad una *Breadboard Small*, un *Potenciometer*, un *Resistor* de 220Ω, un *LED*, y un *Arduino*. Conecte todos los elementos del circuito de una manera apropiada y programe el Arduino para la tarea pedida. Considere usar el montaje de la figura 2.



```
int pot;
void setup()
{
}

void loop()
{
    pot = analogRead(0);
    pot = ((pot*255)/1023);
    analogWrite(3,pot);
}
```

Ejercicio 5. Realice un circuito en Tinkercad para escribir mensajes en un *Liquid-Crystal Display* (LCD) con la ayuda de Arduino. Cambie de *Basic* a *All* en la opción *components* de Tinkercad. Debe agregar un *LCD 16 × 2*, y un Arduino. Conecte todos los elementos del circuito de una manera apropiada y programe el Arduino para la tarea pedida. Considere el esquemático de la Figura 3 y el montaje de la Figura 4.



```

#include <LiquidCrystal.h>

char mensaje[17]="Bienvenido      ";

LiquidCrystal lcd(7,8,9,10,11,12);
void setup()
{
  Serial.begin(9600);
  lcd.begin(16, 0);
}

void loop()
{
  Serial.println("Ingrese el mensaje: ");
  while(Serial.available()>0){
    for(int i=0; i<17; i++){
      mensaje[i]=Serial.read();
    }
  }
  lcd.setCursor(0, 0);
  lcd.print(mensaje);
  lcd.noDisplay();
  delay(500);
  lcd.display();
  delay(500);
}

```

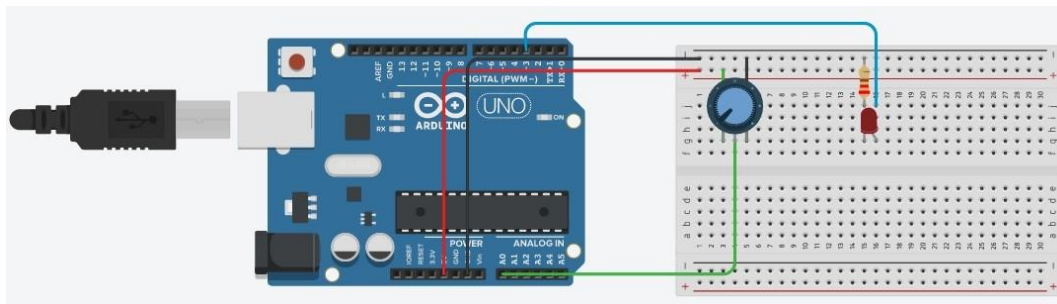


Figura 2: Montaje en protoboard del potenciómetro.

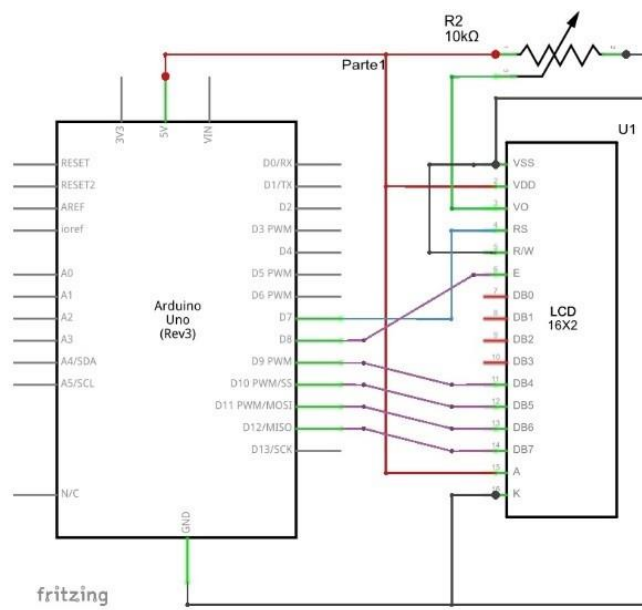


Figura 3: Esquemático Liquid-Crystal Display.

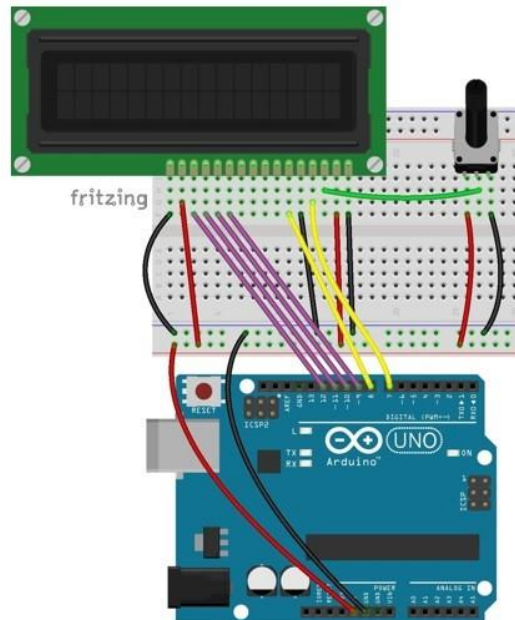


Figura 4: Montaje Liquid-Crystal Display.