**P1: ¿Qué tamaño tiene el tipo de dato double? Compare el tipo double con el tipo float, ¿Qué puede concluir?**

R/ En arduino Uno y ATMEGA el tamaño del dato double es de 4 bytes y para Due el dato double tiene 8 bytes. El tipo de dato doublé es igual que el float pero sin ganancia en la precisión. El tipo flotante se utiliza para aproximar valores analógicos ya que tienen mayor precisión.

**P2: Explore algunas de las bibliotecas estándar de arduino. Explore tres de ellas y explique para qué sirven y de un ejemplo cada una.**

[**Esplora**](https://www.arduino.cc/en/Reference/EsploraLibrary) - esta biblioteca le permite acceder fácilmente a diversos sensores y actuadores montados en el tablero Esplora. P ej: Sensor de temperatura: leer el sensor de temperatura en grados F° o C°. <https://www.arduino.cc/en/Tutorial/EsploraTemperatureSensor>

**WiFi**: con el arduino WiFi shield, se permite la conexión a internet. <https://www.arduino.cc/en/Reference/WiFi>

**Audio:** La biblioteca de audio permite a una placa Arduino reproducir archivos.wav desde un dispositivo de almacenamiento como pej una tarjeta SD. <https://www.arduino.cc/en/Reference/Audio>

**P3: ¿Cuál es la estructura de un programa para un Arduino?**

Setup ()

Loop ()

**P4: ¿Cuál es la función de cada una de las partes de la estructura de un programa en Arduino?**

**P5: localice el archivo main.cpp en el directorio donde descomprimió el archivo .zip descargado de la página del proyecto Arduino. No dude en revisar cada directorio. La función main en un programa en C es el punto de entrada del programa.**

**P5a. ¿Cuántas veces se ejecuta la función setup?**

**P5b. ¿Cuántas veces se ejecuta la función loop?**

**P5c. ¿Cuál es la ruta del archivo main.cpp en el sistema de archivos?**

**P13.** Cree un arreglo con su nombre y escriba un programa que envíe cada uno de los caracteres del nombre al puerto serial. Observe el resultado utilizando CoolTerm.

char nom[] = {'J', 'E', 'F', 'E', 'R'};

void setup() {

Serial.begin(9600);

}

void loop() {

// put your main code here, to run repeatedly:

int x;

for (x = 0; x < 5; x = x + 1) {

Serial.println(nom[x]);

delay(9600);

}

}

**P14.** ¿Cuáles son los operadores de comparación? de ejemplos de cada uno

* [==](https://www.arduino.cc/en/Reference/If) (equal to)

x == y (x is equal to y)

* [!=](https://www.arduino.cc/en/Reference/If) (not equal to)
* x != y (x is not equal to y)
* [<](https://www.arduino.cc/en/Reference/If) (less than)
* x < y (x is less than y)
* [>](https://www.arduino.cc/en/Reference/If) (greater than)
* x > y (x is greater than y)
* [<=](https://www.arduino.cc/en/Reference/If) (less than or equal to)
* x <= y (x is less than or equal to y)
* [>=](https://www.arduino.cc/en/Reference/If) (greater than or equal to)

x >= y (x is greater than or equal to y)

P15. Construya programas ejemplo para las siguientes estructuras de control: if, if-else, if-else if- else, for, switch-case, while.

P16. Escriba e implemente el código que se muestra a la derecha. ¿Para qué sirve el modificador [const](https://www.arduino.cc/en/Reference/Const)?

Es una variable calificativa que modifica el comportamiento de una variable y hace que una variable sea de solo lectura, por lo tanto si le asigna un valor distinto a esta vatiable, se recibira un error como mensaje.

C:\Program Files (x86)\Arduino/hardware/tools/avr/bin/avr-objcopy -O ihex -R .eeprom C:\Users\ADMINI~1\AppData\Local\Temp\build5406998282943980105.tmp/punto\_16\_const.cpp.elf C:\Users\ADMINI~1\AppData\Local\Temp\build5406998282943980105.tmp/punto\_16\_const.cpp.hex

Sketch uses 1.054 bytes (3%) of program storage space. Maximum is 32.256 bytes.

P17. Compile de nuevo el programa. Compare el mensaje final del proceso de compilación con el mensaje que aparece en la figura de este slide. ¿Qué se puede concluir?

Sketch uses 1.112 bytes (3%) of program storage space. Maximum is 32.256 bytes.

Global variables use 13 bytes (0%) of dynamic memory, leaving 2.035 bytes for local variables. Maximum is 2.048 bytes.

* El tamaño pasa de 9 bytes a 13 bytes, es decir que ya no es una variable const y pasa a ser declarada como un int y este ocupa mayor memoria

P18. ¿Cómo funciona? explique cada línea de código

void setup() {

// put your setup code here, to run once:

Serial.begin(9600);

Serial.println("Enter a letter A - F: "); // muestra en pantalla la frase que esta entre comillas.

}

void loop() {

// put your main code here, to run repeatedly:

char c; //declara el character c

while (true){

if (Serial.available()>0){

c = Serial.read();

c = toupper(c);

switch (c) {

case 'A':

Serial.println("great job"); // si se ingresa el character ‘A’, muestra en pantalla ("great job")

break;

case 'B':

case 'C':

Serial.println("you passed"); // si se ingresan el character ‘B’o ‘C’, muestra en pantalla ("you passed")

break;

case 'D':

Serial.println("you are on the edge ");

break;

case'F':

Serial.println("see you again next semester. "); // si se ingresa el character ‘F’, muestra en pantalla ("see you again next semester. ")

break;

default:

Serial.println("u cant even follow instructions?");

break; // si se ingresa cualquier otro dato muestra ("u cant even follow instructions?")

}

}

}

}

avrdude done. Thank you.

P19. ¿Cómo funciona el programa que muestra la figura de la derecha?

Es un ciclo for donde se inicializa la variable k en k=0, y realiza un incremento de uno en uno para esta variable con los valores hasta los k<10

**P21.** Muestre cómo funciona el stack en este programa.

|  |  |
| --- | --- |
| **SP** |  |
| Value return | **1** |
| value | 0------1 |
| Increment | 1 |
|  | Direct Return |
| Final value | **1** |

P22. Invente un programa que haga uso de los siguientes operadores lógicos: &&, ||, ! .