

Samanburður á Python og Arduino C/C++

Sjá má allt um Arduino C/C++ hér: <https://www.arduino.cc/en/Reference/HomePage>

Breytur

Helsti munurinn er að taka þarf fram í **C** af hvaða tagi breytan á að vera.

Python	C
<pre>heiltala = 10 kommatala = 5.7 stafur = 'A' strengur = "Texti" Ekki hægt í python en oft notaðir hástafir til að tákna fasta t.d. PI = 3.14</pre>	<pre>int heiltala = 10; float kommutala = 5.7; <u>eða</u> double kommutala = 5.7; char stafur = 'A'; // einn stafur String strengur = "Texti"; const float pi = 3.14; // núna er pi 3.14 og ekki hægt að breyta því</pre>

Föll

Svipað og með breytur þá þarf að taka fram í **C** hvaða tagi fallið skilar og af hvaða tagi færíbreyturnar eru.

Python	C
<pre>def skilar_engu(): skyring = "og tekur enga færíbreytu."</pre>	<pre>void skilar_engu() { Serial.println("og tekur enga færíbreytu."); }</pre>
<pre># fallið leggur saman tvær tölur # og skilar niðurstöðunni def leggja_saman(a, b): return a + b</pre>	<pre>/* fallið leggur saman tvær <u>heiltölur</u> og skilar niðurstöðunni */ int leggja_saman(int a, int b) { return a + b; } /* fallið leggur saman tvær <u>kommutölur</u> og skilar niðurstöðunni */ float leggja_saman(float a, float b) { return a + b; }</pre>

Skrifað á skjá

Arduino er ekki með skjá en notaður er **Serial Monitor** sem hægt er að skrifa út á.

Python	C
<pre>x = 20 y = 10 print(f"X = {y}, Y = {x}") print("Texti í nýrri línu")</pre>	<pre>// Þessi lína er sett í setup fallið Serial.begin(9600); // Þessar línur geta svo verið // í loop fallinu int x = 20; int y = 10; Serial.print("X = "); Serial.print(x); Serial.print(", Y = "); Serial.println(y); Serial.println("Texti í nýrri línu"); Úttakið yrði: X = 20, Y = 10 Texti í nýrri línu</pre>

Lykkjur

for lykkjan í C er talsvert öðruvísi en í Python.

Python	C
<pre>for i in range(10): print(i)</pre>	<pre>for(int i = 0; i < 10; i++) { // <u>i++</u> jafngildir <u>i += 1</u> Serial.println(i); // og <u>i = i + 1</u> }</pre>
<pre>for i in range(10, 0, -1): print(i, end='')</pre>	<pre>for(int i = 10; i > 0; i--) { Serial.print(i); }</pre>

while lykkjan.

Python	C
<pre>i = 0 while i < 10: print(i) i += 1</pre>	<pre>int i = 0; while(i < 10) { Serial.println(i); i++; }</pre>

do-while lykkjan, athugið að hún keyrir alltaf að minnsta kosti einu sinni.

Python	C
Ekki til.	<pre>int i = 0; do { Serial.println(i); i++; } while(i < 10)</pre>

Listar

Í **C** eru notuð fylki þar sem listar eru notaðir í Python. Í **C** þarf að tilgreina af hvaða tagi fylkið er og er ekki hægt að vera með blönduð fylki.

Python	C
listi = [1, 2, 3, 4, 5]	int listi[] = {1, 2, 3, 4, 5};
listi[3] = 10	listi[3] = 10;
listi = [1, "abc", 4.3]	Ekki hægt
<pre>for l in listi: print(l)</pre>	<pre>for(int i = 0; i < 5; i++) { // 5 af því það eru 5 stök Serial.println(listi[i]); }</pre>

Skilyrðissetningar

Auk **if** býður **C** upp á **switch-case**.

Python	C
<pre>if a > b: print(a) elif a < b: print(b) else: print(a, b)</pre>	<pre>if(a > b) { Serial.println(a); } else if(a < b) { Serial.println(b); } else { Serial.print(a); Serial.println(b); }</pre>
Ekki til	<pre>switch (a) { // case tilfellin geta verið eins mörg og þarf case 4: Serial.println("Fjórir"); break; case 5: Serial.println("Fimm"); break; default: Serial.println("Keyrir ef a er hvorki fjórir né fimm"); break; }</pre>

Vinna með strengi

Meðhöndlun textastrengja er mjög svipuð og í Python ef strengurinn er búinn til sem **String**.

Python	C
<pre>strengur = "Texti" strengur += " annar texti"</pre>	<pre>String strengur = "Texti"; strengur += " annar texti");</pre>

Dæmi um Arduino forrit

Öll Arduino forrit þurfa að innihalda tvö föll **setup** sem er keyrt einu sinni og **loop** sem keyrir aftur og aftur á meðan Arduino tölvan er tengd rafmagni.

C	Arduino
<pre> /* Forrit sem lætur ljós tengt við pinna 2 blikka */ // Nota pinna 2 fyrir led peruna const int rautt_led = 2; // þetta er global breyta // önnur global breyta sem heldur utan um hvort kveikt er á perunni int ljós_stada = LOW; // ljósið byrjar slökkt // Í setup fallinu eru allar þær stilingar sem þarf að gera gerðar // setup fallið keyrir bara einu sinni void setup() { // Set pinna 2 sem úttakspinna pinMode(rautt_led, OUTPUT); } // loop fallið keyrir aftur og aftur endalaust void loop() { // Athuga hvort ljósið er slökkt if(ljós_stada == LOW) { // það er slökkt, ég breyti stöðunni á því ljós_stada = HIGH; } else { // það er kveikt ljós_stada = LOW; } // Skrifa stöðuna á ljósinu á pinna 2 digitalWrite(rautt_led, ljós_stada); // Býð í hálfa sekúndu áður en fallið er keyrt aftur delay(500); } </pre>	