

# **MODUL PENGGUNAAN**

## **micro:bit**

# **DALAM BILIK DARJAH**

**Direka khas untuk kandungan DSKP RBT Darjah 5 dan 6**



**REKA BENTUK TEKNOLOGI**

# Isi Kandungan



<b>A</b>	PENGENALAN KEPADA MIKROPENGAWAL DAN PENGATURCARAAN	3
<b>B</b>	ASAS ANTARA MUKA PERISIAN PENGATURCARAAN MICRO:BIT	6
<b>C</b>	STRUKTUR KAWALAN JUJUKAN DENGAN PAPARAN AKSARA DAN IMEJ	10
<b>D</b>	PENGATURCARAAN BERASASKAN PERISTIWA DAN INPUT PADA MICRO:BIT	14
<b>E</b>	LITAR DAN KOMPONEN LUARAN	17
<b>F</b>	PEMBOLEH UBAH DAN STRUKTUR KAWALAN PILIHAN	25
<b>G</b>	STRUKTUR KAWALAN ULANGAN	30
<b>H</b>	PENDERIA-PENDERIA LAIN DAN PROJEK DIY	33
<b>I</b>	LAMPIRAN	37



# PENGENALAN KEPADA MIKROPENGAWAL & PENGATURCARAAN



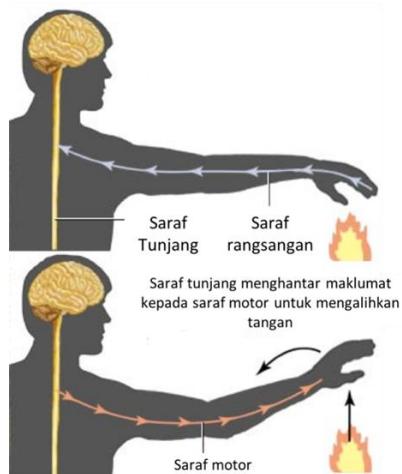
## Objektif Pembelajaran

Darjah 5

- 7.3.2 Mengenal pasti struktur kawalan pilihan dan ulangan dalam algoritma yang diberikan  
 7.3.3 Membezakan struktur kawalan jujukan, pilihan dan ulangan melalui algoritma dengan kaedah pseudokod dan carta alir

## Apakah itu komputer dan mikropengawal?

Komputer, secara umumnya, ditakrifkan sebagai sejenis peranti kawalan yang memproses maklumat satu atau lebih daripada satu input dan menghasilkan *output* yang sesuai.

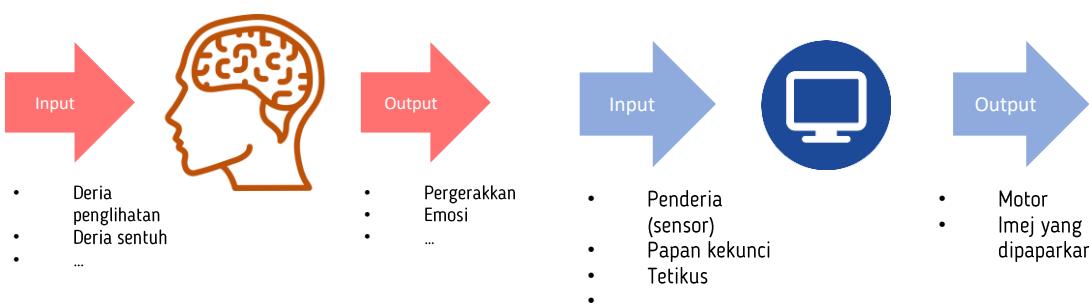


Rajah 1: Sistem saraf manusia

Komputer adalah seperti otak - contohnya, apabila tangan kita terkena api, saraf rangsangan akan terima maklumat daripada tangan dan menghantar maklumat tersebut ke saraf tunjang dan seterusnya ke otak untuk diproses.

Setelah otak kita memproses api itu sebagai maklumat yang berbahaya, maklumat output pula akan dihantar melalui saraf motor untuk mengalihkan tangan kita daripada api.

Kita boleh menulis kod arahan dan memasukkan kod itu dalam komputer untuk dilaksanakan. Mikropengawal adalah sejenis komputer yang mampu memproses *input* dan mengeluarkan *output* yang sesuai. Contoh mikropengawal adalah seperti micro:bit, Arduino, dan sebagainya.



Rajah 2: Perbandingan antara otak dan komputer

## Apakah itu pengaturcaraan?

Pengaturcaraan adalah penulisan arahan untuk memberitahu komputer cara untuk memproses maklumat input untuk menghasilkan output. Set arahan ini juga boleh dinamakan algoritma. Aturan arahan yang baik mestilah tertib, jelas dan berkesan bagi menyelesaikan sesuatu masalah.

Kita juga boleh mewakilkan algoritma dalam bentuk lain iaitu bentuk pseudokod ataupun carta alir

## Pseudokod

Pseudokod ialah satu set aturan yang ditulis dalam bahasa pertuturan untuk menyelesaikan sesuatu masalah. Pseudokod memerihalkan langkah-langkah algoritma dengan menggunakan ayat-ayat yang ringkas dan padat, dalam Bahasa Melayu atau Bahasa Inggeris. Indentasi digunakan untuk memudahkan pembacaan sesuatu pseudokod.

### Contoh 1

```

MULA
Bangun pagi
Gosok gigi
Cuci muka
Pakai baju
Pergi sekolah
TAMAT

```

### Contoh 2

```

MULA
SEMAK keadaan butang
JIKA butang ditekan
NYALAKAN lampu
TAMAT JIKA
TAMAT

```

### Contoh 3

```

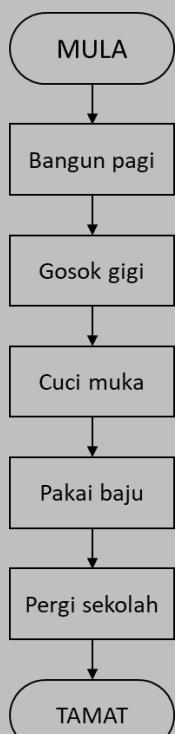
MULA
SELAGI lapar
MAKAN nasi
TAMAT SELAGI
TAMAT

```

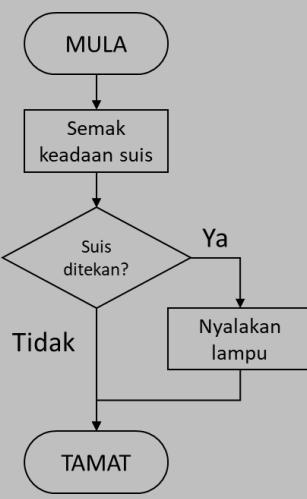
## Carta alir

Carta alir menggunakan simbol grafik seperti garis lurus, anak panah dan bentuk geometri untuk mewakili urutan langkah bagi algoritma yang perlu dilaksanakan.

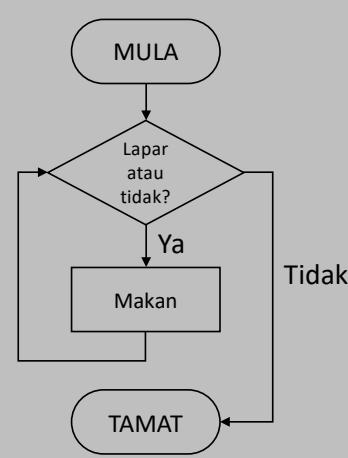
### Contoh 1



### Contoh 2



### Contoh 3



Mula / Tamat

Langkah

Pilihan

Rajah 3: Maksud bentuk dalam carta alir

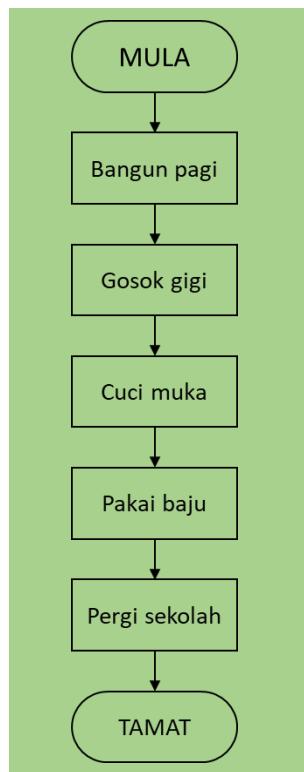
## Struktur kawalan

Struktur kawalan merujuk kepada cara ataupun urutan sesuatu algoritma. Terdapat 3 struktur kawalan yang lazim digunakan dalam algoritma, iaitu:

- i) Struktur kawalan jujukan
- ii) Struktur kawalan pilihan
- iii) Struktur kawalan ulangan

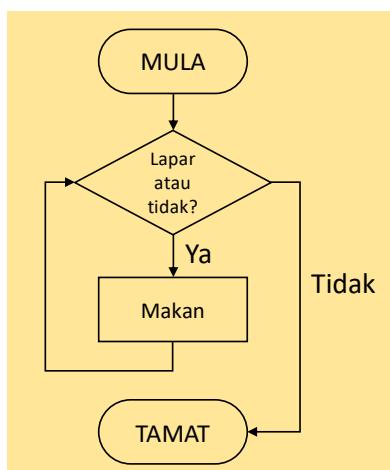
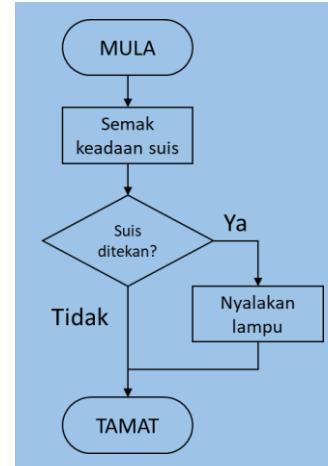
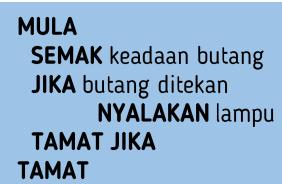
### Struktur kawalan jujukan

Struktur kawalan jujukan merujuk kepada algoritma yang tidak bercabang dan terdiri daripada langkah yang mempunyai urutan tertentu.



### Struktur kawalan pilihan

Struktur kawalan pilihan memecahkan aliran algoritma kepada dua atau lebih cabang, dan setiap cabang mempunyai set langkah yang berlainan untuk dilaksanakan. Set mana yang akan dilaksanakan adalah bergantung kepada syarat-syarat tertentu.



### Struktur kawalan ulangan

Struktur kawalan ulangan adalah algoritma di mana terdapat langkah yang akan dilaksanakan berulang kali. Bahagian langkah yang berulang kali dikenali sebagai gelung (*loop*) manakala setiap ulangan langkah dikenali sebagai lelaran (*iteration*)



# ASAS ANTARA MUKA PERISIAN PENGATURCARAAN MICRO:BIT



## Objektif Pembelajaran

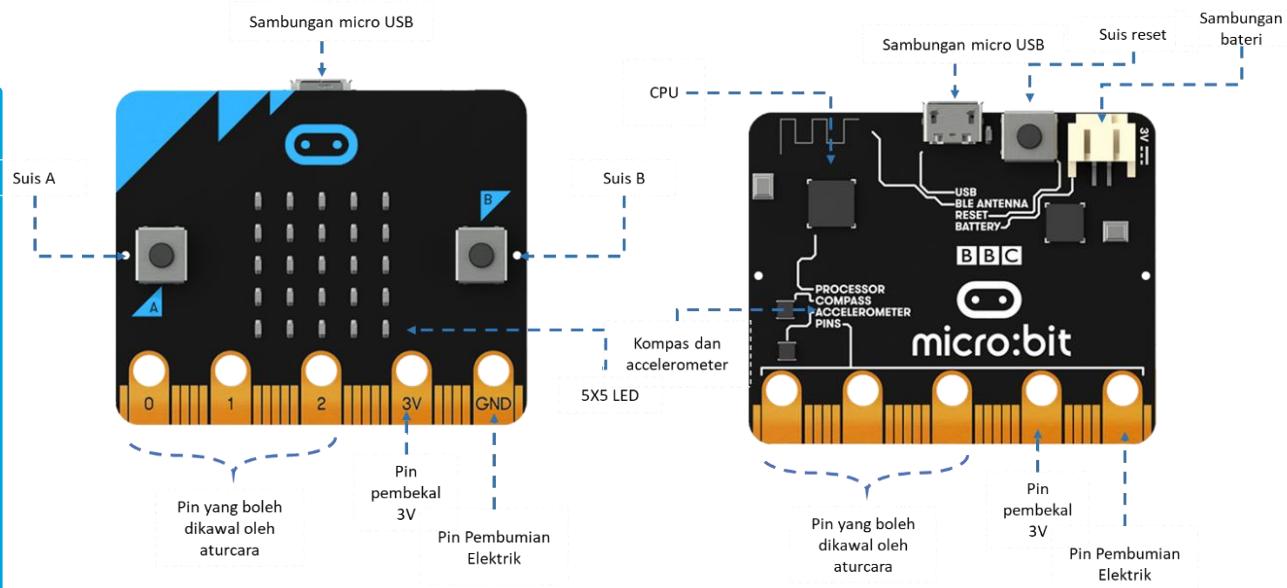
Darjah 5

7.4.1 Mengenal pasti fitur-fitur pada antara muka perisian pengaturcaraan

7.4.2 Mengenal pasti perkakasan yang akan digunakan bersama perisian pengaturcaraan

## Apakah itu micro:bit?

micro:bit merupakan komputer bersaiz poket yang mempunyai 2 butang dan panel LED yang mempunyai 5x5 LED. Selain daripada itu, micro:bit juga mempunyai beberapa penderia (*sensor*) dan juga pin-pin yang boleh diguna pakai untuk tujuan pengaturcaraan.



Rajah 4: Bahagian-bahagian micro:bit

Micro:bit juga mempunyai fitur-fitur dan penderia berikut:

<b>Accelerometer</b>	Penderia ini membolehkan micro:bit untuk mengesan pergerakan micro:bit dalam bentuk goncangan ( <i>shake</i> ), condong ( <i>tilt</i> ) dan jatuh ( <i>fall</i> )
<b>Kompas</b>	Mengesan arah micro:bit
<b>Antena BLE</b>	Membolehkan micro:bit berkomunikasi dengan peranti-peranti lain dengan menggunakan <i>bluetooth</i>
<b>Penderia suhu</b>	Mengesan suhu micro:bit
<b>Penderia cahaya</b>	Mengesan keterangan cahaya di sekitar micro:bit

Dengan semua fitur ini, micro:bit mampu mencipta pelbagai projek. Contohnya, menyalakan lampu amaran jika suhu terlalu panas atau menjadikan kompas digital dan sebagainya.

Untuk menghasilkan projek, kita boleh menggunakan komponen-komponen yang sedia ada pada micro:bit atau menyambungkan komponen-komponen luaran.

Kita perlu memasukkan arut cara dalam micro:bit untuk menghasilkan projek. Segala komputer di dunia ini, sebesar komputer di stesen angkasa lepas ataupun sekecil micro:bit memerlukan arut cara untuk berfungsi. Micro:bit membolehkan kita belajar pengaturcaraan dengan mudah.

## Pengaturcaraan micro:bit

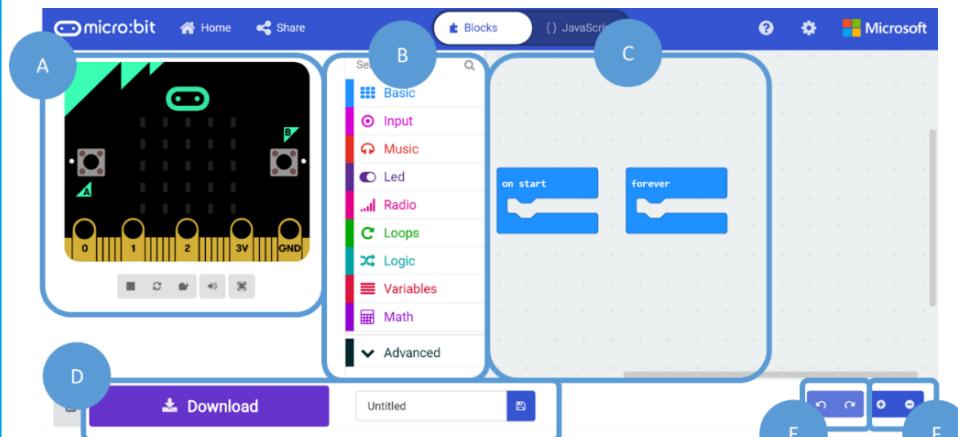
Terdapat pelbagai cara untuk mengaturcara micro:bit, seperti pengaturcaraan dengan Python, Javascript ataupun berasaskan blok. Cara yang paling mudah adalah pengaturcaraan micro:bit yang berasaskan blok dan amat mudah untuk digunakan. Dalam buku ini, kita akan fokus kepada pengaturcaraan berasaskan blok.

Untuk mulakan pengaturcaraan micro:bit, layari <https://makecode.microbit.org/#> dan tekanan "New Project".



Rajah 5: Perisian atas talian makercode

## Antara muka perisian pengaturcaraan



Rajah 6: Antara muka perisian pengaturcaraan micro:bit

## A – Tetingkap simulasi

Kita boleh menguji arur cara dengan tetingkap simulasi tanpa menyambungkan micro:bit. Butang pada tetingkap simulasi boleh ditekan umpama butang yang betul.

## B – Laci blok atur cara

Blok-blok atur cara boleh ditarik keluar daripada sini ke zon atur cara untuk membentuk kod atur cara. Blok-blok atur cara disusun mengikut kategori seperti asas, input, muzik dan sebagainya.

## C – Zon Atur cara

Atur cara micro:bit adalah dalam bentuk blok. Blok-blok boleh digabungkan untuk membentuk atur cara. Pada asasnya, terdapat dua blok khas yang wujud pada micro:bit iaitu blok “On Start” dan blok “Forever”.



Blok “On Start” – Blok ini akan berjalan sekali sahaja sebaik sahaja micro:bit melaksanakan atur cara.

Blok “Forever” – Blok ini akan berulang secara tidak terhingga apabila micro:bit melaksanakan atur cara

## D – Muat turun dan simpan

Apabila atur cara sudah siap disusun, kita boleh memuat turun atur cara dalam bentuk hex, iaitu bentuk yang boleh difahami oleh micro:bit ataupun muat turun projek sebagai fail projek untuk diubah nanti.

# Atur cara pertama anda

## Langkah 1

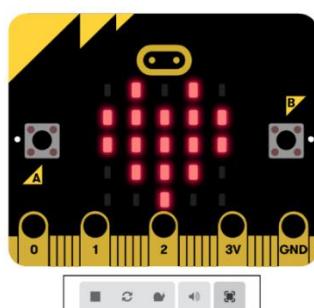
Cari dan gabungkan blok-blok untuk membentuk susunan blok berikut:



(Blok boleh didapati pada bahagian “Basic”)

## Langkah 2

Kita boleh melihat simulasi kod pada bahagian tetingkap simulasi. Sekiranya anda tidak sempat untuk melihat atur cara berjalan, boleh tekanan butang “refresh” untuk ulang semula semua atur cara.

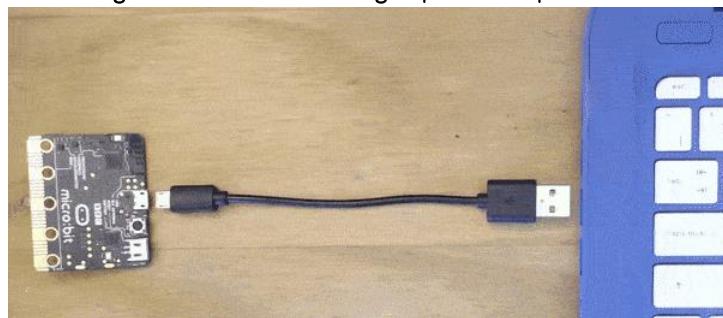


- 1) Stop – Hentikan perisian simulasi
- 2) Refresh – Set semula
- 3) Slow-mp – Simulasi berjalan secara gerak perlahan
- 4) Mute audio – Meredamkan bunyi
- 5) Launch in full screen – Melihat simulasi secara skrin penuh

Rajah 7: Tetingkap simulasi atur cara micro:bit

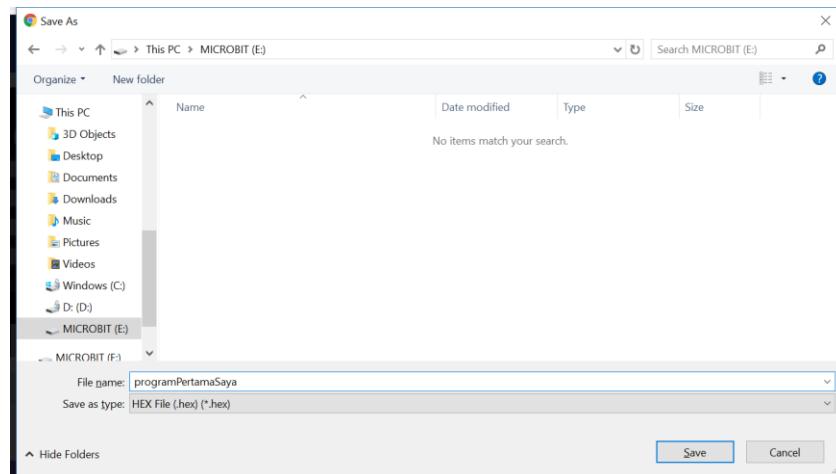
### Langkah 3

Sambungkan micro:bit pada komputer dengan menggunakan wayar USB seperti mana sambungan telefon untuk pengesanan elektrik. Bahagian kecil micro-USB bersambung kepada micro:bit di port micro-USB manakala bahagian besar bersambung kepada komputer.



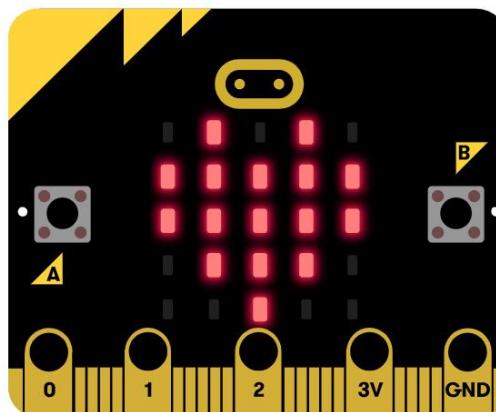
Rajah 8: Sambungan micro:bit pada komputer

Setelah sambungan dibuat, tekan “Download” pada perisian pengaturcaraan dan sama ada anda muat turun fail .hex ke komputer dan kemudian memindahkan fail itu ke dalam micro:bit (yang akan dipaparkan apabila buka “My Computer”) ataupun terus muat turunkan fail .hex ke dalam micro:bit..



Rajah 9: Muat turun fail .hex ke dalam micro:bit

Atur cara anda akan dimuat naik ke dalam micro:bit dan anda boleh lihat pelaksanaan atur cara dalam micro:bit - iaitu menunjukkan perkataan “Hello” dan bentuk hati pada LED.



Rajah 10: Projek lengkap

# STRUKTUR KAWALAN JUJUKAN DENGAN PAPARAN IMEJ DAN AKSARA



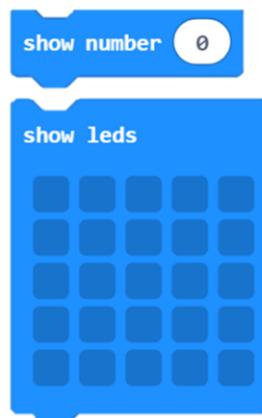
## Objektif Pembelajaran

Darjah 5

7.4.3	Menghasilkan carta alir projek reka bentuk yang akan dibuat iaitu yang mengeluarkan cahaya, bunyi dan pergerakan
7.4.4	Membuat sambungan perkakasan projek reka bentuk yang akan dihasilkan
7.4.5	Membina atur cara yang dikehendaki berpanduan carta alir yang dibuat
7.4.6	Membuat simulasikan pada atur cara yang telah dibina
7.4.7	Memindahkan atur cara yang telah dibina pada hardware
7.4.8	Menjalankan atur cara pada perkakasan yang mengeluarkan cahaya, bunyi dan pergerakan

## Mengawal LED terbina

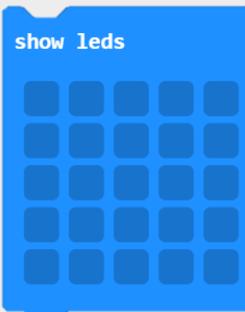
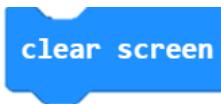
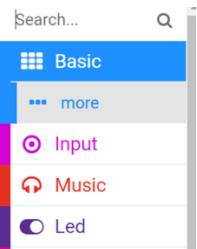
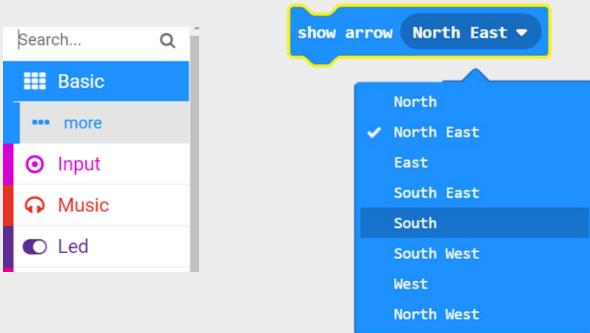
Micro:bit mempunyai 25 LED terbina (*built in*) yang boleh dikawal dengan atur cara. Dalam Bahagian B, kita sudah melihat cara untuk memaparkan bentuk dan perkataan pada LED dengan atur cara. Blok-blok di rajah 11 membolehkan kita mengawal 25 LED pada micro:bit untuk menunjukkan pelbagai corak dan aksara.



Rajah 11: Blok kawalan LED

Blok	Fungsi
	Memaparkan nombor dengan LED.
	Memaparkan huruf, ataupun ayat dengan LED

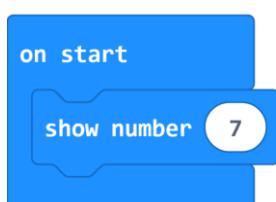
Blok	Fungsi
	Memaparkan corak yang telah disetkan dalam micro:bit. Tekan butang segi tiga putih untuk melihat pelbagai corak yang boleh dipaparkan dengan micro:bit

Blok	Fungsi
	Memaparkan corak yang telah disetkan dalam micro:bit. Tekan butang segi tiga putih untuk melihat pelbagai corak yang boleh dipaparkan dengan micro:bit.
	Memadamkan segala imej / aksara pada LED (blok ini boleh didapatkan dengan menekan butang "more") 
	Memaparkan anak panah yang akan menuju ke arah-arah yang ditetapkan (blok ini boleh didapatkan dengan menekan butang "more") 

Blok-blok yang hendak digunakan tidak boleh diletakkan begitu sahaja di luar - blok tersebut mesti diletakkan di dalam satu struktur kawalan - sekiranya blok berwarna seperti berikut, bermaksud blok itu tidak akan dilaksanakan oleh micro:bit



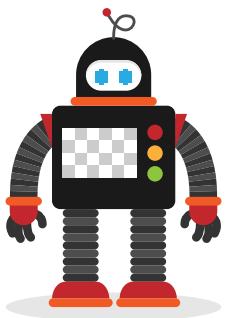
Rajah 12: Contoh blok yang tidak boleh dilaksanakan



Rajah 13: Contoh blok yang boleh dilaksanakan

Untuk mengatasi masalah ini, kita boleh meletakkan blok tersebut di dalam blok "On Start" (untuk dijalankan sekali sahaja pada permulaan atur cara) ataupun blok "Forever" (untuk dijalankan tidak terhingga kali pada atur cara)





### Cabaran micro:bit C1

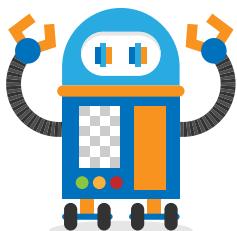
**C1-1**

Memaparkan nama anda pada micro:bit secara berulangan

**C1-2**

Memaparkan bulan lahir anda pada micro:bit sekali sahaja

## Struktur kawalan jujukan



Menghasilkan projek animasi pada micro:bit yang menunjukkan muka senyum untuk 2 saat dan seterusnya menunjukkan bentuk hati untuk 2 saat. Akhirnya, padamkan semua LED.

Untuk kehendak projek tersebut, kita boleh mula dengan merancang arah cara kita. Dalam projek ini, memandangkan kehendak ini tiada cabang pilihan, kita boleh menggunakan struktur kawalan jujukan untuk menghasilkan projek ini.

Langkah pertama dalam sebarang penghasilan projek adalah perancangan algoritma dengan melukiskan carta alir.

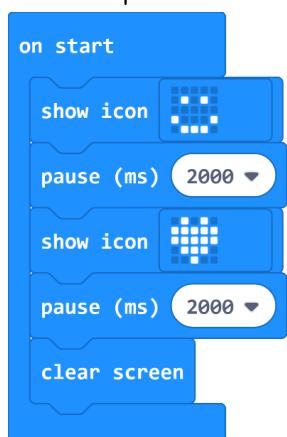
### Carta alir projek



Untuk menghasilkan projek ini, kita boleh menggunakan blok “pause” untuk menggantikan arahan tunggu pada carta alir. Blok ini boleh didapati di bahagian “Basic”.



Nombor di dalam pause itu adalah dalam ukuran millisaat. 1 saat mempunyai 1000 millisaat. Kita boleh mengubah nombor di dalam ruangan itu dengan menaip masuk nombor yang dikehendaki. Contohnya, 3000 untuk 3 saat.



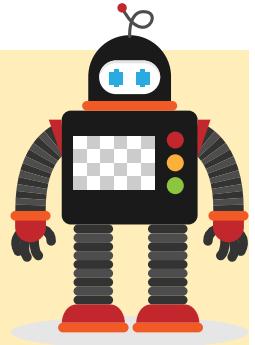
Oleh sebab kita perlu menjalankan program ini sekali sahaja, maka kita akan menggunakan blok “On Start”. Sebelum muat naik arah cara, kita boleh menyemak sama ada arah cara tersebut adalah betul atau tidak dengan menggunakan fungsi simulasi pada perisian arah cara.

Rajah 14: Atur cara projek

## Cabaran micro:bit C2

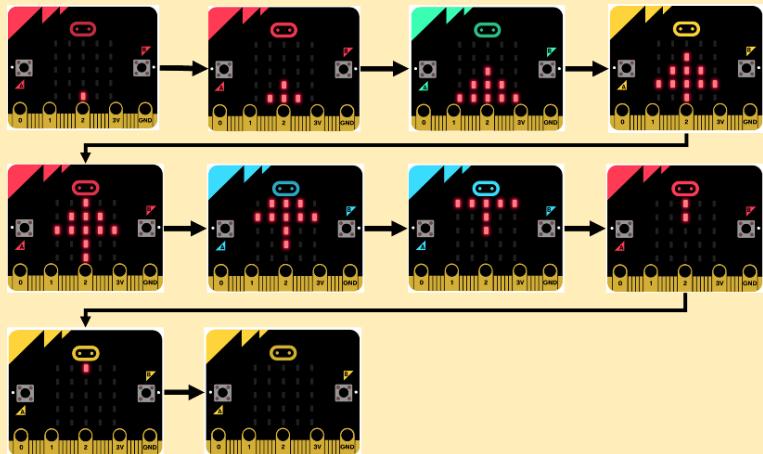
C2-1

Diberi gabungan kod berikut, apakah carta alir yang sesuai untuk algoritma ini?



C2-2

Dengan menggunakan blok “show led”, hasilkan animasi berikut :



C2-3

Hasilkan arah cara berdasarkan carta alir berikut:



# PENGATURCARAAN BERASASKAN PERISTIWA DAN INPUT PADA MICRO:BIT



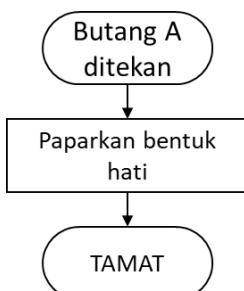
## Objektif Pembelajaran

Darjah 5

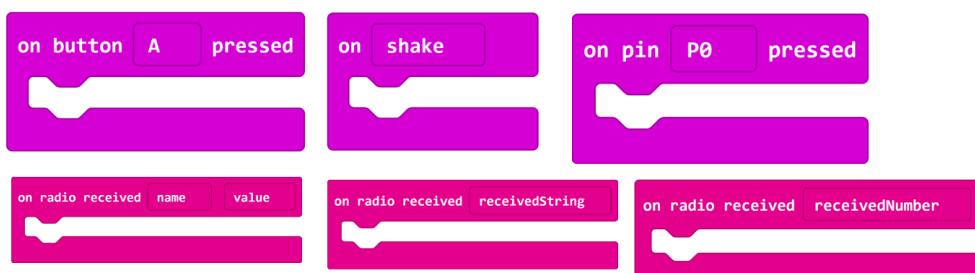
7.4.3	Menghasilkan carta alir projek reka bentuk yang akan dibuat iaitu yang mengeluarkan cahaya, bunyi dan pergerakan
7.4.4	Membuat sambungan perkakasan projek reka bentuk yang akan dihasilkan
7.4.5	Membina arur cara yang dikehendaki berpanduan carta alir yang dibuat
7.4.6	Membuat simulasii pada arur cara yang telah dibina
7.4.7	Memindahkan arur cara yang telah dibina pada hardware
7.4.8	Menjalankan arur cara pada perkakasan yang mengeluarkan cahaya, bunyi dan pergerakan

## Pengaturcaraan berasaskan peristiwa

Pengaturcaraan berasaskan peristiwa (*event-based programming*) merujuk kepada arur cara yang dicetuskan oleh peristiwa. Dalam konteks microbit, ini boleh merujuk kepada kemasukan input daripada pengguna. Dalam carta alir, perkataan MULA pada permulaan carta alir akan diubah kepada peristiwa yang berlaku.

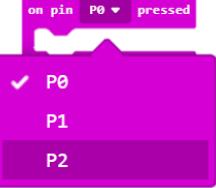
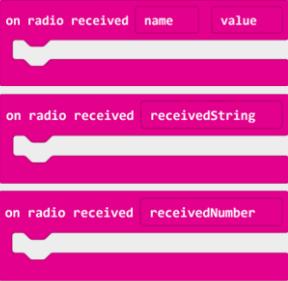


Rajah 15: Carta alir pengaturcaraan berasaskan peristiwa

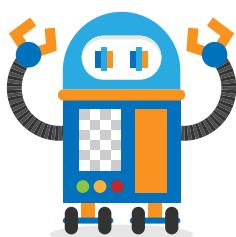


Rajah 16: Contoh blok pencetus peristiwa (event trigger)

Blok	Fungsi
	<p>Apabila butang A ditekan, blok yang terkandung di dalam blok ini akan dilaksanakan.</p> <p>Kita boleh mengubah butang yang dikehendaki dengan menekan huruf A pada blok.</p>

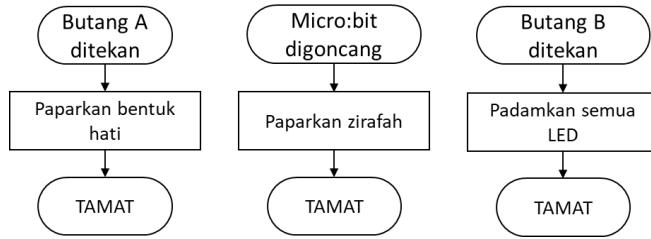
Blok	Fungsi
	<p>Apabila micro:bit mengesan goncang, blok yang terkandung di dalam blok ini akan dilaksanakan.</p> <p>Selain daripada goncangan, blok ini juga boleh digunakan untuk mengesan pencetus-pencetus lain seperti kecondongan dan orientasi micro:bit</p> 
	<p>Apabila pin 0 ditekan (terdapat sambungan dari pin 0 ke pin GND), blok yang terkandung di dalam blok ini akan dilaksanakan.</p> <p>Nama pin boleh diubahkan dengan menekan perkataan P0</p> 
	<p>Apabila micro:bit menerima arahan-nilai melalui bluetooth, blok yang terkandung di dalam akan dilaksanakan.</p> <p>“on radio received name value” – laksana apabila radio menerima pasangan nilai dalam bentuk 1 ayat dan 1 nilai nombor</p> <p>“on radio received receivedString” – laksana apabila radio menerima nilai dalam bentuk ayat atau perkataan</p> <p>“on radio received receivedNumber” – laksana apabila radio menerima nilai dalam bentuk nombor</p>

Dalam micro:bit, pelbagai blok pencetus peristiwa boleh digunakan pada masa yang sama. Dalam contoh berikut, kita mencipta projek yang mempunyai spesifikasi berikut:

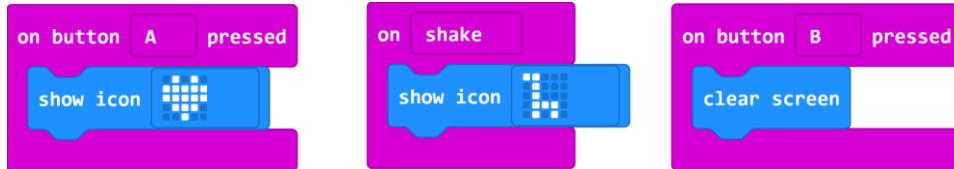


Apabila pengguna tekan butang A, gambar hati akan dipaparkan. Apabila pengguna menggoncangkan micro:bit, gambar zirafah akan dipaparkan. Apabila pengguna tekan butang B, semua LED akan dipadamkan.

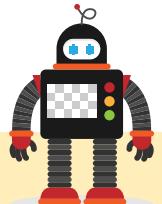
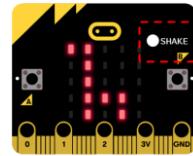
Untuk mereka projek ini, kita harus memerihalkan kehendak projek dan membina carta alir untuk setiap satu spesifikasi



Kita boleh membina atur cara berpandukan carta alir.

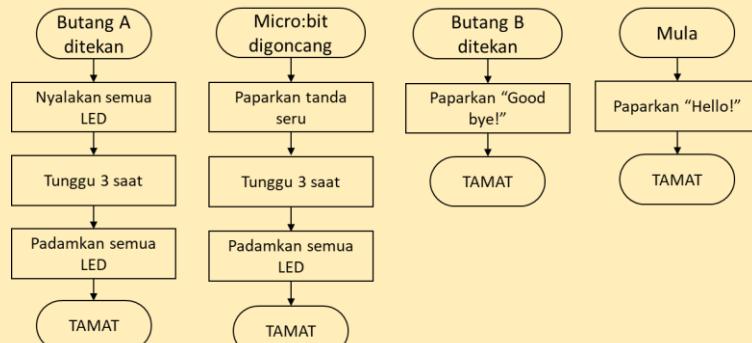


Sebelum kita muat naik atur cara ke micro:bit, kita boleh menguji atur cara tersebut dengan simulasi terlebih dahulu. Aksi goncang boleh disimulasikan dengan menekan butang "SHAKE" pada papan simulasi.



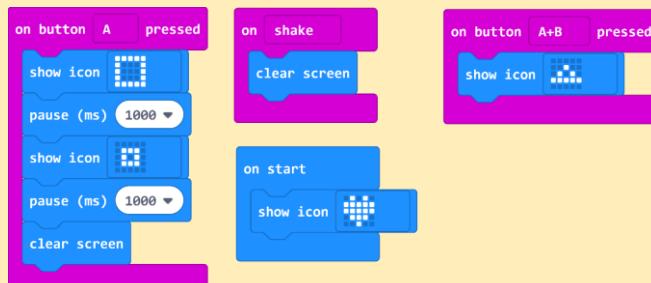
## Cabaran micro:bit D1

- D1-1** Hasilkan atur cara berdasarkan carta alir berikut (untuk gambar tanda seru, anda perlu melukiskan gambar itu dengan blok "show leds")



- D1-2**

Lukiskan carta alir untuk atur cara berikut:



- D1-3**

Hasilkan lampu suluh dengan menggunakan LED pada micro:bit. Lampu suluh anda perlu ada fungsi "On" dan "Off". Hasilkan carta alir untuk projek ini.



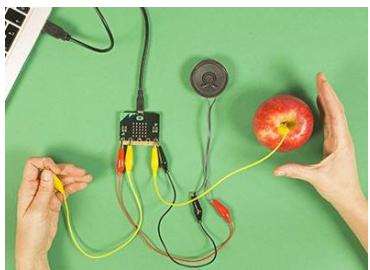
# LITAR DAN KOMPONEN LUARAN



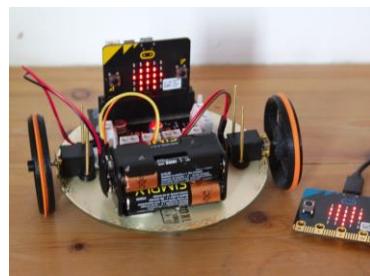
## Objektif Pembelajaran Darjah 5

7.4.3	Menghasilkan carta alir projek reka bentuk yang akan dibuat iaitu yang mengeluarkan cahaya, bunyi dan pergerakan
7.4.4	Membuat sambungan perkakasan projek reka bentuk yang akan dihasilkan
7.4.5	Membina atur cara yang dikehendaki berpandukan carta alir yang dibuat
7.4.6	Membuat simulasi pada atur cara yang telah dibina
7.4.7	Memindahkan atur cara yang telah dibina pada hardware
7.4.8	Menjalankan atur cara pada perkakasan yang mengeluarkan cahaya, bunyi dan pergerakan
7.4.9	Membentangkan atur cara yang telah dihasilkan

Kita boleh sambungkan komponen-komponen kepada micro:bit dan menulis atur cara untuk mengawal komponen-komponen yang bersambung kepadanya.



Sumber: <https://make.techwillsaveus.com/microbit/activities/touch-arpeggiator>



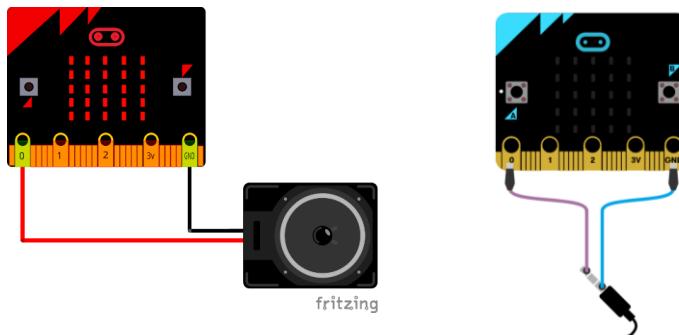
Sumber: <https://www.instructables.com/id/Remote-Controlled-Microbit-Robot/>

Rajah 17: Contoh projek micro:bit dengan sambungan komponen luaran

## Bunyi

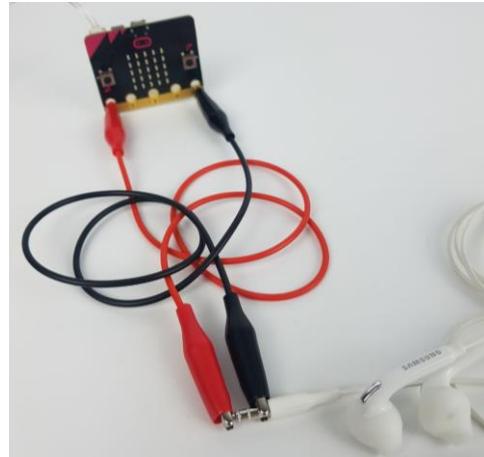
Buzzer (penggera) atau speaker boleh dipasangkan kepada pin 0 dan pin GND untuk menghasilkan bunyi.

Sambungkan buzzer dengan menggunakan klip buaya yang disediakan. Jika buzzer anda mempunyai keikutuan (terdapat tanda + dan - pada kaki buzzer), sambungkan terminal positif kepada pin 0 dan terminal negatif kepada GND. Selain daripada menggunakan buzzer seperti dalam gambar, anda juga boleh menggunakan fon telinga (earphone) untuk menggunakan fungsi bunyi. Pastikan pin 0 disambungkan pada bahagian hujung kabel dan pin GND pada bahagian atas kabel seperti dalam gambar.



Rajah 18: Rajah sambungan buzzer / fon telinga kepada micro:bit

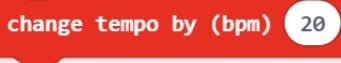
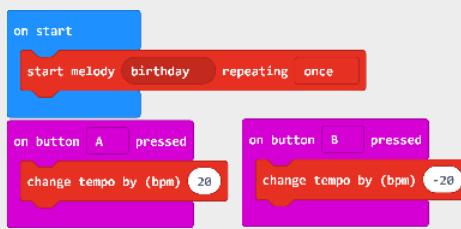
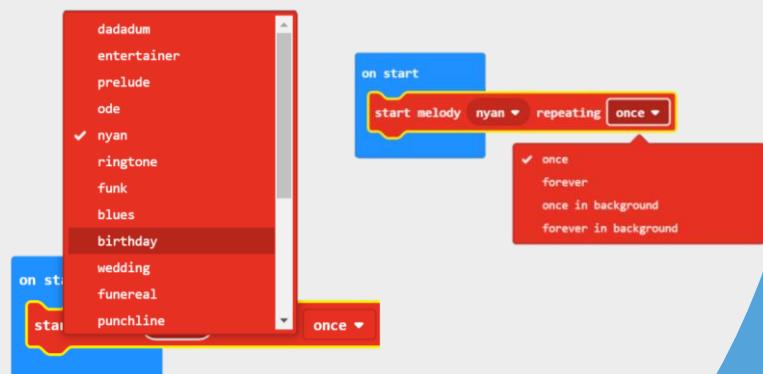
Dengan adanya sambungan bateri, micro:bit boleh berfungsi tanpa wayer USB dan komputer

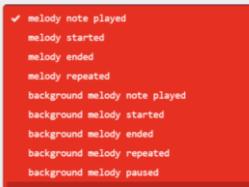


Rajah 19: Sambungan buzzer / fon telinga pada micro:bit

Blok berkaitan bunyi boleh didapati di bahagian blok “Music”

Blok	Fungsi
	<p>Blok ini memainkan nota muzik “Middle C” sebanyak “1” beat.</p> <p>Nota muzik dan tempoh nota muzik boleh diubah dengan menekankan perkataan “Middle C” untuk ubah nota muzik ataupun “1” untuk mengubah tempoh nota muzik. 1 beat adalah bersamaan dengan 1 saat.</p>
	<p>Memainkan nota muzik “Middle C” tanpa berhenti.</p> <p>Nota muzik boleh diubah dengan menekan “Middle C”</p>
	Menetapkan tempo muzik ke 120

Blok	Fungsi
	<p>Menambahkan tempo muzik (jika nombor positif) dan mengurangkan tempo muzik (jika nombor negatif).</p> <p>Contoh berikut menunjukkan atur cara di mana butang A membuat tempo muzik menjadi laju manakala butang B memperlambahkan muzik.</p> 
	<p>Berhenti main bunyi selama “1” beat.</p> <p>Tempoh “beat” boleh diubah dengan menekan “1” untuk ubah ke tempoh lain</p> 
	<p>Memainkan muzik yang telah dimasukkan dalam atur cara micro:bit. Blok ini membolehkan buzzer memainkan muzik yang telah ditetapkan – contohnya, <u>“Happy Birthday”</u>.</p> <p>“dadadum” merujuk kepada muzik yang dikehendaki, manakala “once” merujuk kepada jumlah kali yang hendak dimainkan.</p> 

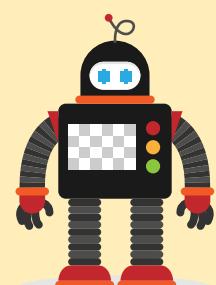
Blok	Fungsi
	<p>Blok peristiwa muzik. Melaksanakan atur cara yang terkandung di dalam blok ini apabila peristiwa yang berkaitan muzik berlaku.</p> <p>Contoh peristiwa yang boleh dikesan adalah “apabila muzik dimainkan” ataupun “apabila muzik tamat” – peristiwa boleh diubah dengan menekan “melody note played”</p>  

Contoh berikut adalah blok di mana gambar muka senyum akan dipaparkan apabila muzik bermula dan gambar muka sedih akan dipaparkan apabila muzik berhenti.



## Cabarán micro:bit E1

- E1-1** Hasilkan projek di mana apabila micro:bit digoncangkan, lagu “wawawawa” akan dimainkan. Lukiskan carta alir untuk projek ini.
- E1-2** Lukiskan carta alir untuk atur cara berikut dan menguji atur cara tersebut di dalam micro:bit



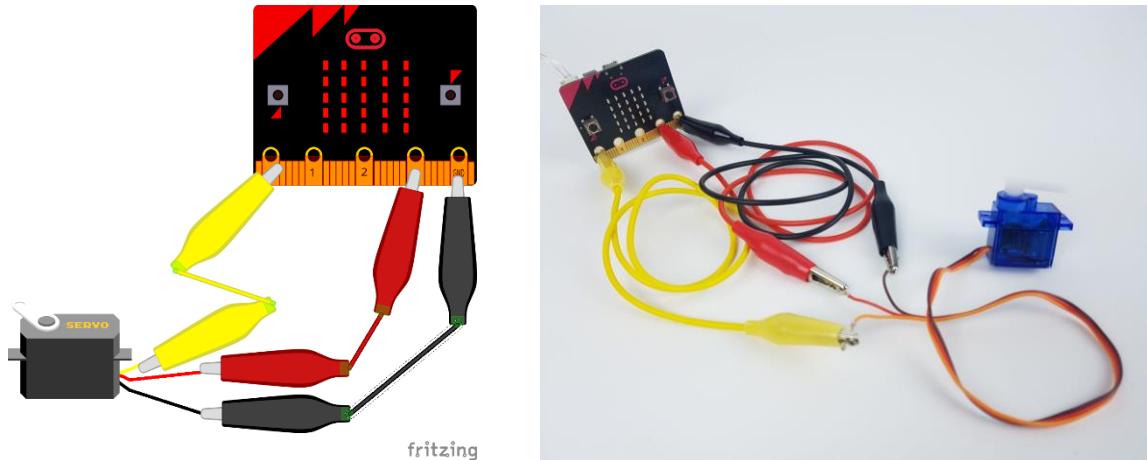
- E1-3** Hasilkan projek berdasarkan carta alir berikut:



## Motor

Untuk sambungan motor, kita akan menggunakan motor “servo motor”. Servo motor merupakan sejenis motor di mana pergerakannya boleh dikawal dengan atur cara. Contohnya, membuat motor itu berpusing 90 darjah.

Servo motor yang digunakan di dalam buku ini adalah model EF92A. Servo motor yang lain juga mungkin boleh digunakan untuk tujuan projek tetapi voltan motor itu perlu menggunakan 3V sahaja kerana micro:bit hanya mampu membekalkan 3V untuk menghidupkan motor itu.

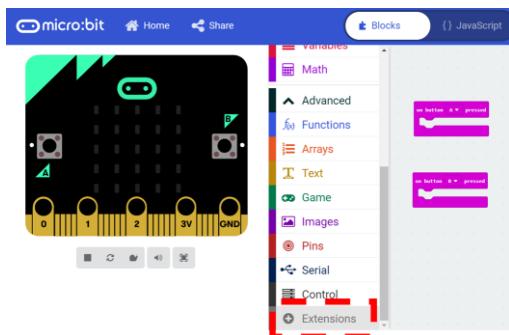


Rajah 20: Sambungan servo motor pada micro:bit

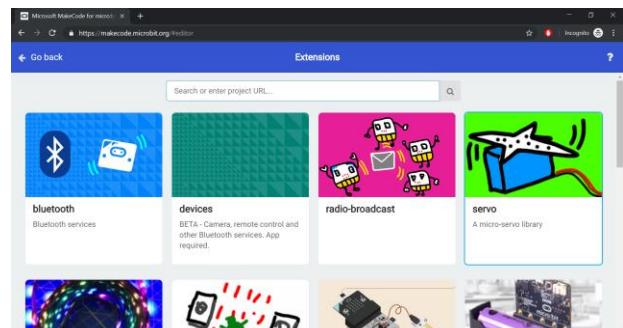
Sambungkan wayar merah pada 3V pada micro:bit, wayar hitam pada GND pada micro:bit dan wayar kuning kepada pin 0 pada micro:bit

Atur cara servo motor memerlukan blok khas - blok khas ini boleh dimasukkan dalam senarai blok dengan langkah berikut:

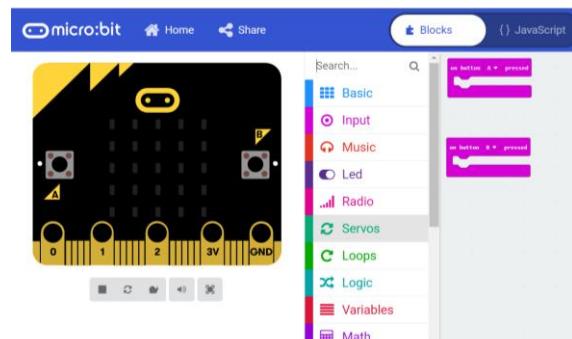
### Langkah 1: Tekan “Extension”



### Langkah 2: Tekan “Servo”



### Langkah 3: Carikan blok “Servo”

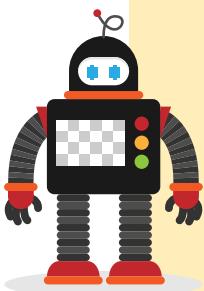


Blok berikut membolehkan kita untuk mengawal darjah pusingan servo motor. Darjah pusingan boleh diubah suai dengan menekan “90”. Pin motor juga boleh diubah suai dengan menekan P0.

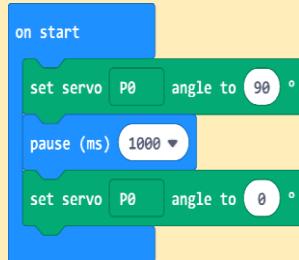
set servo P0 angle to 90 °

### Cabaran micro:bit E2

#### E2-1



Cuba nyatakan apakah fungsi atur cara berikut dan lukiskan carta alir untuk atur cara berikut:



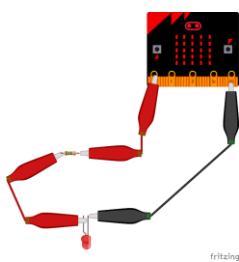
#### E2-2

Hasilkan projek di mana anda boleh mengawal pergerakan motor dengan butang A dan B. Hasilkan carta alir untuk projek tersebut.

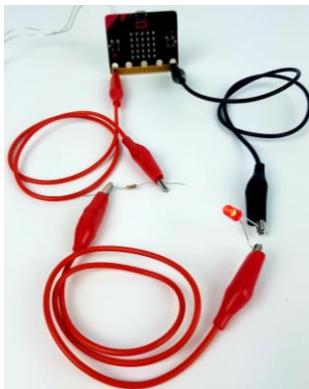
## LED Luaran

Selain daripada LED yang sedia bina (built-in) dalam micro:bit, kita juga boleh menyambungkan LED luaran (external) pada micro:bit.

(a)



(b)



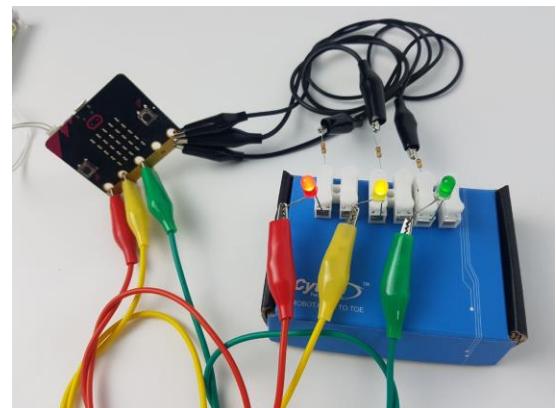
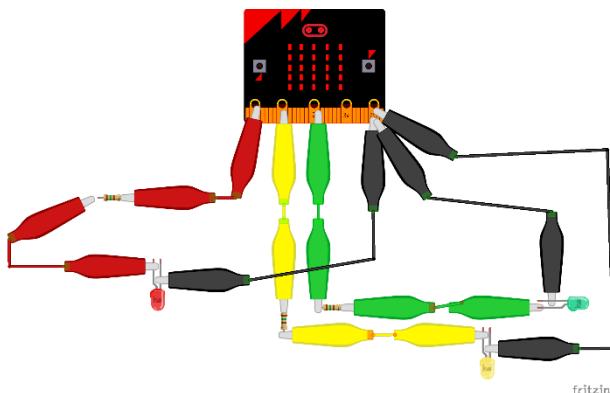
(c)



Rajah 21: Sambungan LED luaran pada micro:bit

Sambungkan LED seperti dalam gambar rajah. LED mempunyai dua kaki, pastikan kaki panjang disambungkan pada perintang bernilai 100 ohm dan seterusnya pada pin 0. Kaki pendek akan disambungkan pada pin GND pada micro:bit. Jika anda mempunyai bongkah penyambung, sambungan antara perintang dan LED boleh disambungkan dengan bongkah penyambung seperti (c) di rajah 21 .

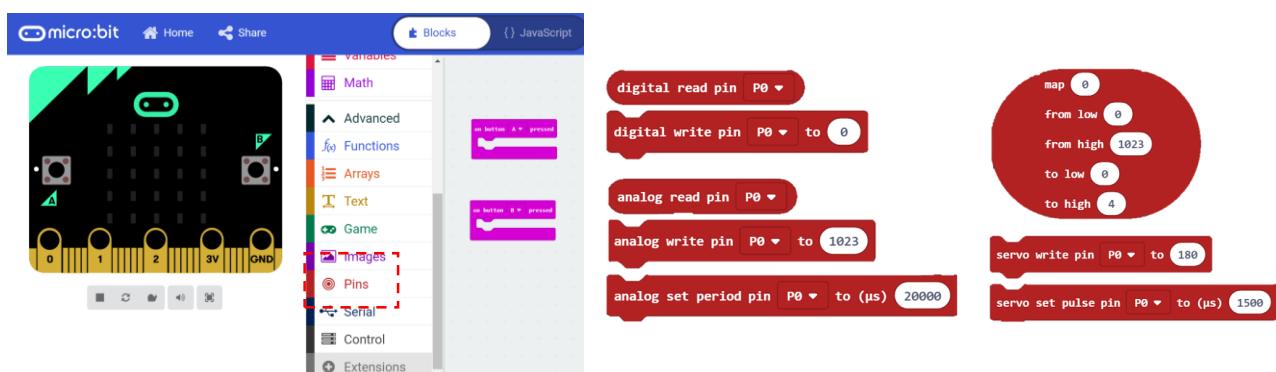
Anda juga boleh menyambung lebih daripada satu LED dengan cara ini.



Rajah 22: Sambungan 3 LED luaran pada micro:bit

Untuk menyalaikan LED pada pin-pin, kita perlu mengawal / menghantar isyarat kepada pin tersebut.

Blok mengawal pin boleh didapati pada bahagian “Pin” setelah menekan “Advanced”



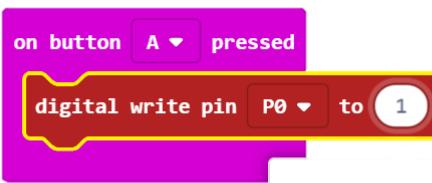
Rajah 23: Blok kawalan pin

Blok kawalan pin membolehkan kita membaca maklumat daripada pin (“digital read” / “analog read”) serta menghantar maklumat kepada pin (“digital write” / “analog write”)

Untuk tujuan topik ini, kita hanya akan fokus kepada blok ini sahaja.

**digital write pin P0 ▾ to 0**

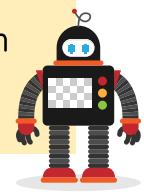
Blok ini menghantar isyarat on/off pada pin yang dinyatakan. Untuk mengubah pin, tekan P0 dan untuk mengubah isyarat yang akan dihantar kepada pin itu, tekan 0. Dengan ini, kita boleh mengawal LED yang dipasangkan di pin yang dinyatakan.



## Cabaran micro:bit E3

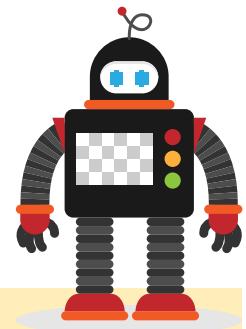
### E3-1

Dengan menggunakan 3 LED (merah, hijau, kuning), hasilkan satu sistem lampu isyarat. Hasilkan carta alir untuk projek tersebut.



## Penghasilan projek

Untuk menghasilkan sesuatu projek, kita boleh mengikuti langkah-langkah berikut untuk memudahkan penghasilan projek.



## Cabaran micro:bit E4

### E4-1

Hasilkan model rumah api dengan 1 LED merah dan LED itu harus berkelip

### E4-2

Dengan menggunakan LED terbina ataupun LED luaran dalam micro:bit, hasilkan signal kereta. (2 LED berkelip)

### E4-3

Hasilkan model lampu isyarat dengan 3 LED

### E4-4

Hasilkan model pintu pejabat berpenggara (tekan butang, pintu buka)

### E4-5

Hasilkan model penggera dengan lampu bernyala

### E4-6

Hasilkan satu projek untuk menyambut hari jadi dengan LED dan muzik

# PEMBOLEH UBAH DAN STRUKTUR KAWALAN PILIHAN



## Objektif Pembelajaran

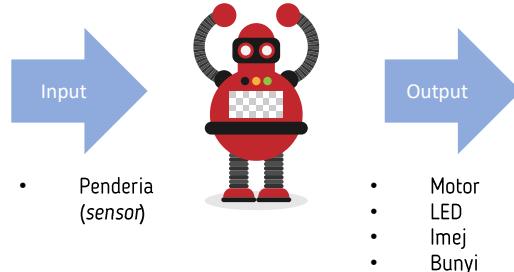
Darjah 6

7.5.1	Menyatakan elemen sistem robotik
7.5.2	Membincangkan elemen sistem robotik yang terdapat pada sebuah robot
7.5.6	Menghasilkan atur cara dalam bentuk pseudokod atau carta alir berdasarkan produk yang dihasilkan
7.5.7	Membuat simulasi pada atur cara yang telah dibina dan membuat penambahbaikan

## Elemen sistem robotik

Sesuatu sistem robotik mesti mempunyai elemen-elemen yang membolehkannya berinteraksi dengan dunia luar dan melakukan tindak balas yang bersesuaian. Dalam erti lain, sistem robotik perlu ada penderia (*sensor*) untuk mengesan maklumat dunia luar serta komponen *output* seperti motor untuk melakukan sesuatu tindak balas pergerakan, buzzer untuk mengeluarkan bunyi, LED untuk memaparkan imej dan sebagainya.

Sistem robotik juga perlu ada pembekalan kuasa serta pengawalan dalam komputer dalaman.



Rajah 24: Elemen yang terkandung dalam satu sistem robotik

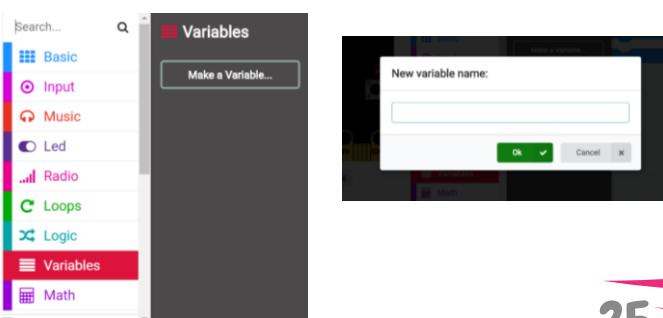
## Pemboleh ubah

Robot boleh mengingati maklumat dunia luar dengan menggunakan pemboleh ubah.

Pemboleh ubah ialah ruang simpanan sementara untuk nombor, teks dan objek. Pemboleh ubah membolehkan kita menyimpan nilai dan merujuk kepada nilai itu tanpa mengetahui alamat memori dalam komputer. Pemboleh ubah boleh berubah nilainya sewaktu proses pengaturcaraan dilaksanakan.

Kita boleh menggunakan pemboleh ubah dalam pengaturcaraan micro:bit untuk menyimpan sesuatu nilai yang boleh digunakan kemudian.

Untuk menggunakan pemboleh ubah, kita perlu mengisytiharkan pemboleh ubah dengan menekan "Variables" kemudian "Make a Variable". Berikan nama yang sesuai untuk pemboleh ubah.



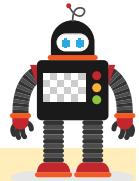
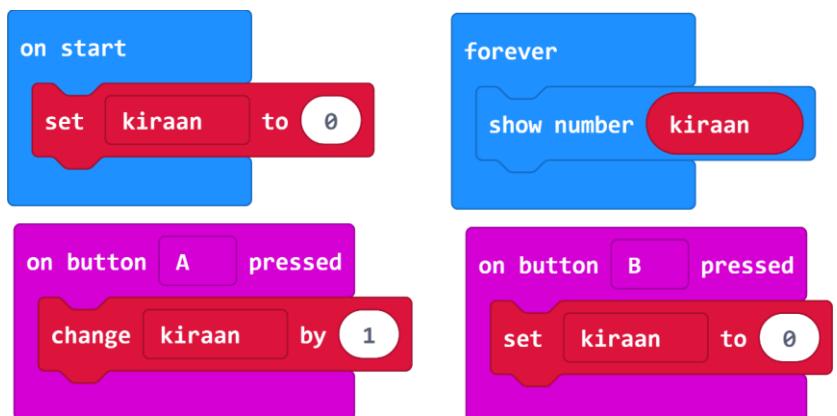
Setelah anda mengisytiharkan pemboleh ubah, beberapa blok baru akan muncul di ruangan “Variable”.

kiraan ▾

set kiraan ▾ to 0

change kiraan ▾ by 1

Contoh berikut menunjukkan dengan jelas cara blok-blok ini digunakan. Atur cara berikut memaparkan jumlah kali kita menekan butang A manakala butang B mengeset semula nombor tekanan ke 0.



### Cabaran micro:bit F1

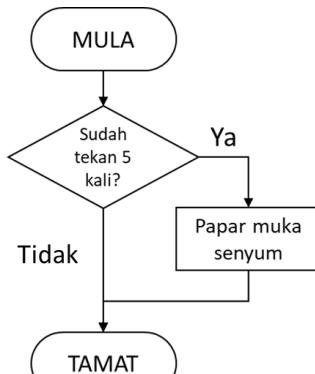
F1-1

Mengubah suai projek contoh supaya butang B akan mengurangkan kiraan bilangan kita menekan butang. Kiraan akan diset semula apabila micro:bit digoncangkan. Hasilkan carta alir untuk kesemua atur cara bagi projek ini.

## Struktur kawalan pilihan

Salah satu sebab sistem robotik memerlukan pemboleh ubah adalah untuk mengingati pengalaman interaksi sistem itu dengan dunia luar. Contohnya, nilai yang dibaca oleh penderia, bilangan kali pengguna menekan butang dan sebagainya.

Bayangkan situasi di mana sistem robotik hanya akan diaktifkan apabila pengguna menekan butang A sebanyak 5 kali.



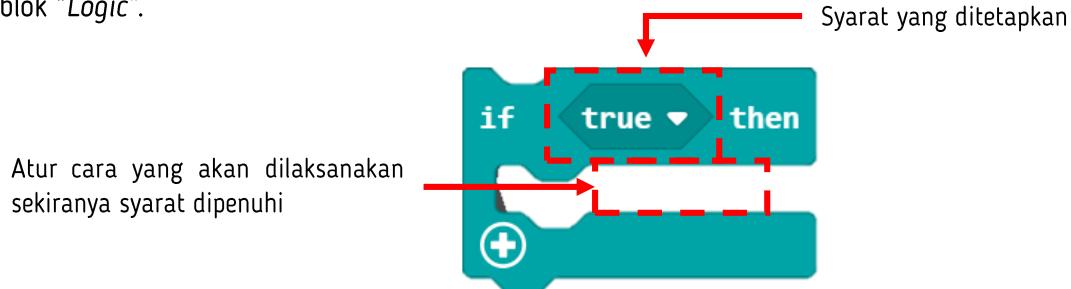
Untuk melaksanakan atur cara ini, komputer bukan sahaja perlu mengingati jumlah kali pengguna menekan butang A, tetapi juga membuat keputusan untuk memaparkan muka senyum dengan LED.

Kita boleh menggunakan blok kumpulan “Logic” untuk melaksanakan struktur kawalan pilihan.

## Pilihan Tunggal

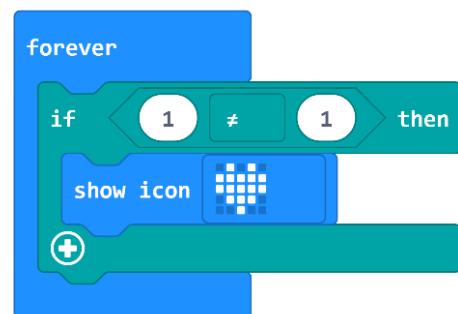
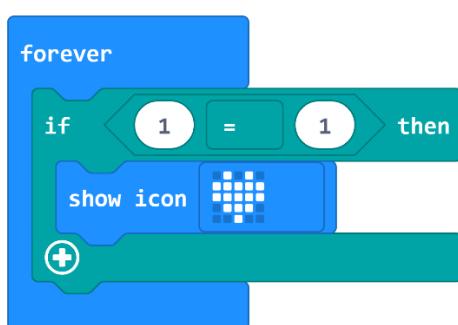
Contoh carta alir di atas menunjukkan struktur kawalan pilihan tunggal (*single choice conditional*) di mana sistem hanya mempunyai satu cabang sahaja berdasarkan syarat yang ditetapkan.

Untuk mengaturcara carta alir itu, kita boleh gunakan blok kawalan pilihan tunggal di dalam kumpulan blok “Logic”.



Jika syarat dipenuhi (syarat boleh diubah dengan meletakkan blok ke dalam “true”), blok yang terkandung di dalam blok pilihan ini akan dilaksanakan. Kalau tidak, tiada atur cara akan dilaksanakan.

Syarat boleh ditetapkan dengan menggunakan blok perbandingan. Jika perbandingan adalah benar, blok ini akan menghasilkan nilai “benar” (True)



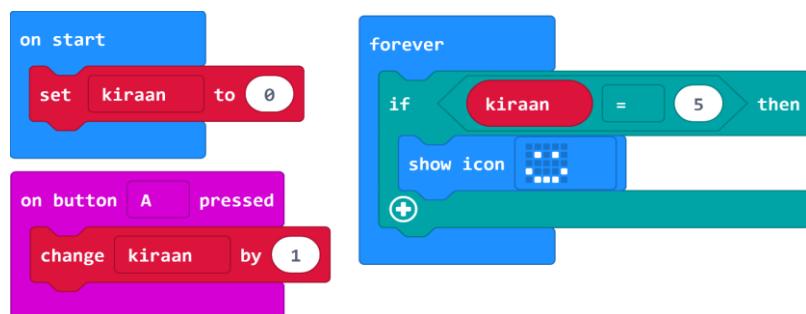
Oleh sebab  $1 = 1$  adalah benar, maka bentuk hati akan dipaparkan. Perhatikan bahawa blok perbandingan adalah sentiasa di dalam blok “*forever*” kerana kita hendak micro:bit sentiasa menyemak sama ada syarat dipenuhi atau tidak.

Jika simbol sama dengan ditukarkan dengan simbol ini (ketidaksamaan), maka bentuk hati tidak akan dipaparkan. Ini kerana  $1$  tidak sama dengan  $1$  adalah tidak benar, dan blok di dalam blok pilihan ini tidak akan dilaksanakan.

Simbol antara dua nombor itu boleh diubah dengan menekan pada simbol. Maksud simbol-simbol itu adalah seperti di bawah:

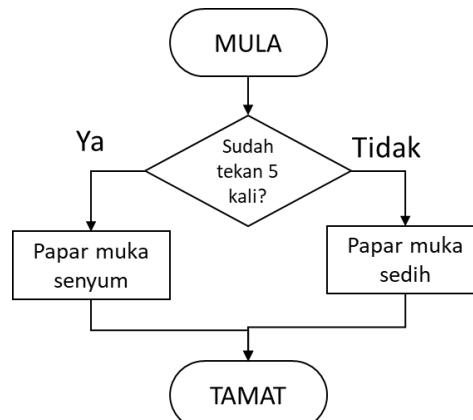
Simbol	Maksud
=	Sama dengan
≠	Tidak sama dengan
<	Lebih kecil daripada
≤	Lebih kecil atau sama dengan
>	Lebih besar daripada
≥	Lebih besar atau sama dengan

Dengan itu, kita boleh menyusun blok untuk membentuk atur cara yang dinyatakan dalam carta alir

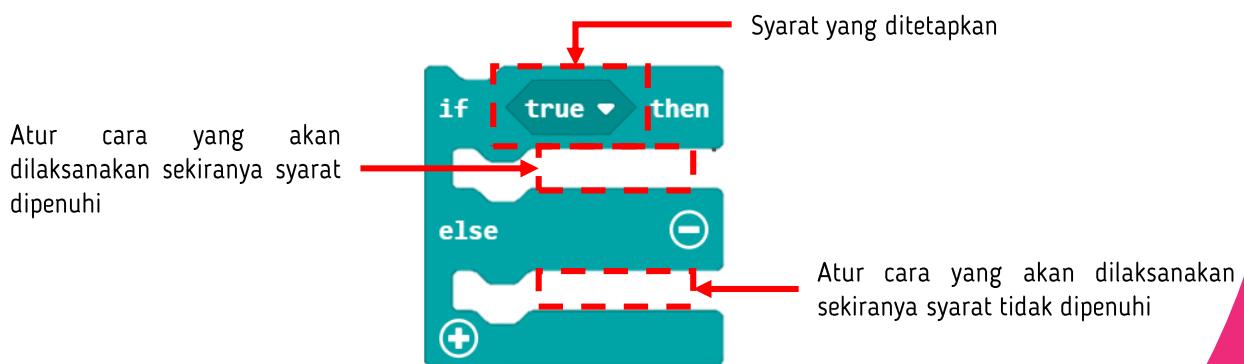


## Struktur Kawalan Dwipilihan

Bagaimana pula dengan carta alir berikut – di mana micro:bit hanya akan memaparkan muka senyum jika bilangan tekanan adalah 5 dan paparkan muka sedih untuk semua bilangan yang lain?

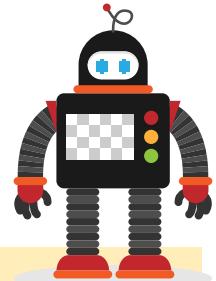
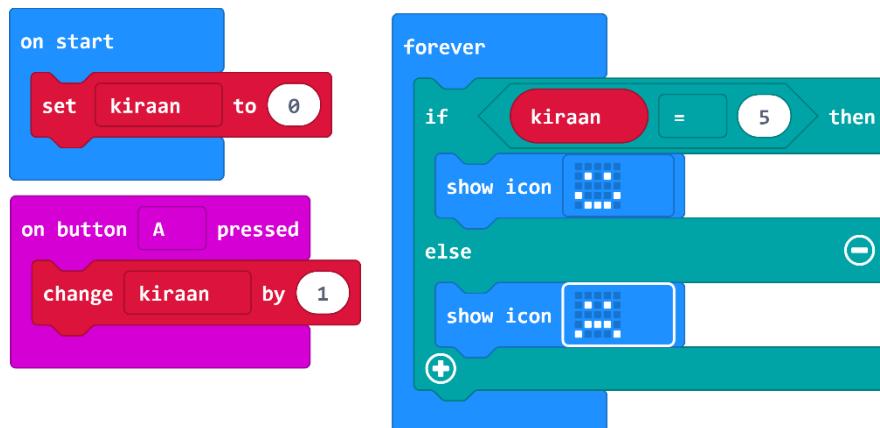


Untuk atur cara ini, kita boleh menggunakan blok struktur kawalan dwipilihan. Struktur kawalan dwipilihan mempunyai dua cabang di mana satu set atur cara akan dilaksanakan jika syarat dipenuhi manakala satu set atur cara lain akan dilaksanakan jika syarat tidak dipenuhi.



Jika syarat dipenuhi, atur cara pada bahagian atas akan dilaksanakan, manakala jika syarat tidak dipenuhi, atur cara pada bahagian bawah akan dilaksanakan.

Dengan itu, kita boleh menyusun blok untuk membentuk atur cara yang dinyatakan dalam carta alir



### Cabaran micro:bit F2

- F2-1** Bayangkan anda mencipta sebuah robot pengesan gempa bumi, di mana jika robot itu berasa digoncang sebanyak 10 kali, maka robot tersebut akan mengeluarkan bunyi amaran. Hasilkan carta alir dan juga projek ini.



# STRUKTUR KAWALAN ULANGAN



## Objektif Pembelajaran

Darjah 6

7.5.4	Melakar secara kreatif reka bentuk produk yang mempunyai elemen sistem robotik
7.5.5	Menghasilkan reka bentuk produk berdasarkan lakaran yang dibuat
7.5.6	Menghasilkan atur cara dalam bentuk pseudokod atau carta alir berdasarkan produk yang dihasilkan
7.5.7	Membuat simulasi pada atur cara yang telah dibina dan membuat penambahbaikan
7.5.8	Memindahkan atur cara yang telah dibina pada produk yang dihasilkan
7.5.9	Menguji atur cara pada produk yang telah dihasilkan

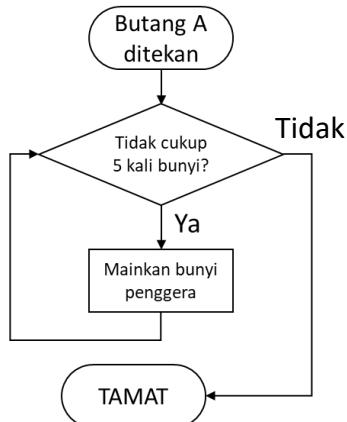
## Struktur kawalan ulangan

Kadangkala, sistem robot perlu melaksanakan sesuatu atur cara secara berulang kali. Dalam Bahagian F, kita sudah lihat bahawa kita boleh menggunakan meletakkan blok “Logic” ke dalam blok “Forever” supaya micro:bit sentiasa menyemak sama ada syarat dipenuhi atau tidak.

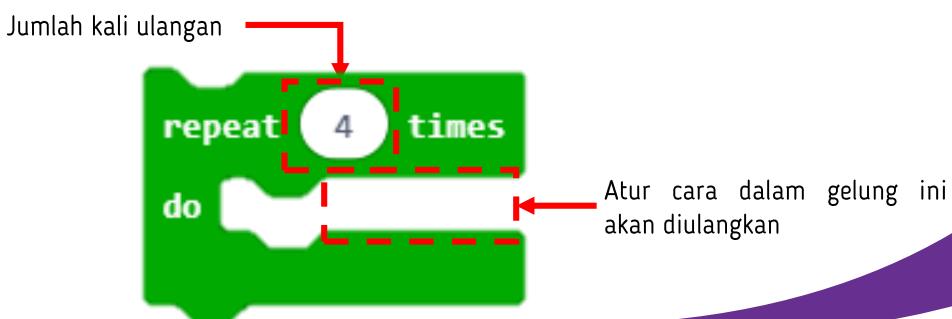
Blok “Forever” sebenarnya merupakan salah satu jenis blok untuk struktur kawalan ulangan. Blok-blok struktur kawalan ulangan lain boleh didapati dalam kumpulan blok “Loop”

### Ulang X kali

Blok “Forever” bermaksud atur cara berjalan sebanyak tidak terhingga kali. Bagaimana pula jika kita ingin robot melakukan sesuatu atur cara sebanyak 5 kali sahaja? Contohnya, robot hanya perlu memainkan bunyi penggera sebanyak 5 kali sahaja?



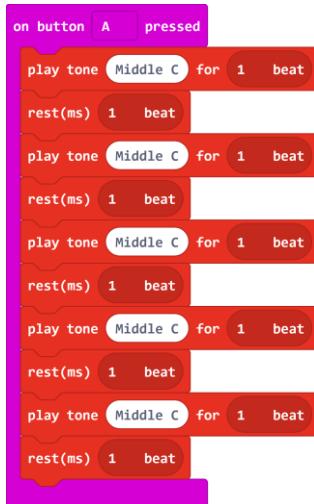
Untuk mengulangi sesuatu atur cara sebanyak 5 kali, atau sebarang nombor kali, kita boleh menggunakan blok “Repeat X times” dalam kumpulan blok “Loop”



Dengan itu, kita boleh menyusun blok untuk membentuk atur cara yang dinyatakan dalam carta alir

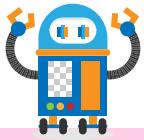


Terdapat beberapa kelebihan menggunakan struktur kawalan ulangan. Contohnya, kriteria projek yang dinyatakan di atas juga boleh diperoleh dengan susunan blok berikut:



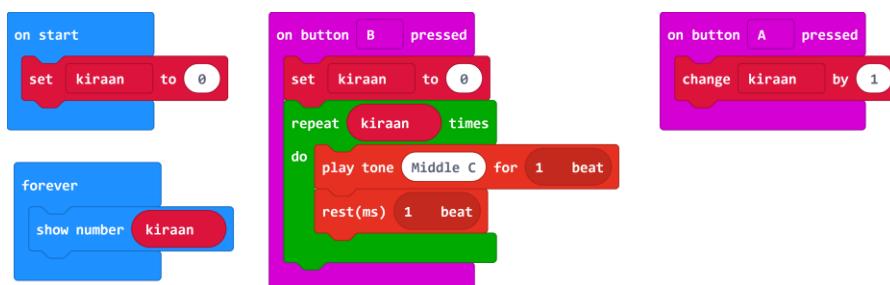
Rajah 25: Atur cara yang kurang teguh

Namun, dalam dunia pengaturcaraan, atur cara seperti itu adalah tidak efisien dan tidak teguh (*robust*) – bayangkan jika bunyi penggera perlu dibunyikan 10 kali, 100 kali mahupun 1000 kali. Struktur kawalan ulangan seperti “Repeat X Times” boleh mengulang sesuatu atur cara untuk beratus dan beribu kali dengan menukar nombor pada blok atur cara.



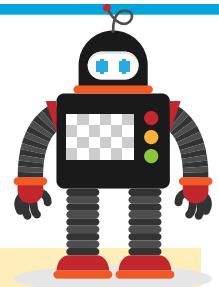
Bayangkan juga kriteria projek berikut:

Tekan butang A untuk menetapkan berapa kali bilangan penggera harus berbunyi. Tekan butang B untuk bunyikan penggera tersebut mengikut bilangan yang ditetapkan serta mengeset semula bilangan tekanan butang. Nombor bilangan tekan akan dipaparkan pada LED



Rajah 26: Atur cara yang tegu dan efisien

Sesuatu atur cara dikira teguh apabila atur cara itu boleh berfungsi dengan pelbagai jenis input. Tak kira kita tekan 1 kali, 10 kali, mahupun 1000 kali, atur cara kita tetap akan menepati kriteria yang ditetapkan iaitu mencatat bilangan kali tekanan butang dan mengeluarkan bilangan bunyi penggera yang sama dengan bilangan kali tekanan butang.



## Cabaran micro:bit G1

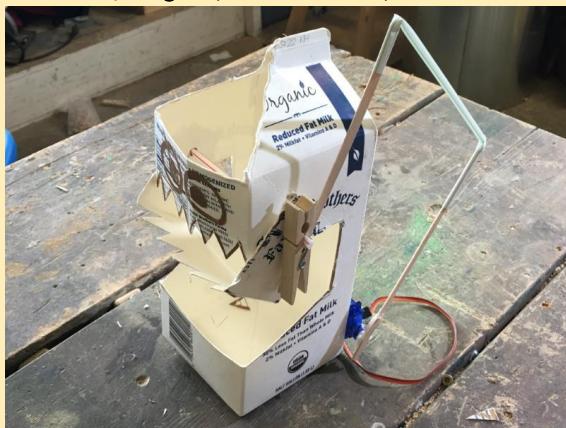
G1-1

Menambahbaik robot yang dihasilkan pada bahagian F dengan menghasilkan "badan" untuk robot anda dengan menggunakan bahan terbuang. Sebelum anda hasilkan badan untuk robot anda, lukiskan lakaran dan mengenal pasti bahan terbuang yang hendak digunakan. Hasilkan carta alir untuk robot anda.

Robot pengesan gempa bumi baru anda boleh:

- a. Mengingati tahap gempa bumi (setiap satu gegaran bersamaan satu tahap)
- b. Apabila butang A ditekan, robot tersebut akan mengeluarkan bunyi penggera berdasarkan berapa tahap gempa bumi dan memaparkan tahap gempa bumi. Contohnya, jika gegaran adalah tahap 3, robot akan keluarkan 3 kali bunyi
- c. Apabila butang B ditekan, tahap gempa bumi akan diset semula.

Contoh badan robot yang diperbuat daripada bahan terbuang (kotak susu)



(Sumber: <https://makecode.microbit.org/projects/milk-carton-robot/make>)

# PENDERIA-PENDERIA LAIN DAN PROJEK DIY



## Objektif Pembelajaran

Darjah 6

7.5.3	Mengenal pasti pemasangan perkakasan pada reka bentuk sebuah robot
7.5.4	Melakar secara kreatif reka bentuk produk yang mempunyai elemen sistem robotik
7.5.5	Menghasilkan reka bentuk produk berdasarkan lakaran yang dibuat
7.5.6	Menghasilkan arur cara dalam bentuk pseudokod atau carta alir berdasarkan produk yang dihasilkan
7.5.7	Membuat simulasi pada arur cara yang telah dibina dan membuat penambahbaikan
7.5.8	Memindahkan arur cara yang telah dibina pada produk yang dihasilkan
7.5.9	Menguji arur cara pada produk yang telah dihasilkan
7.5.10	Membuat rumusan dari projek yang dijalankan

## Penderia sentuh

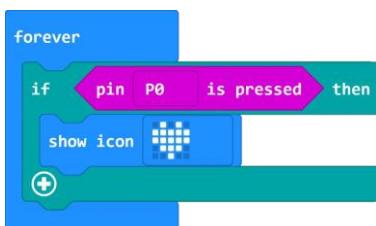
Penderia sentuh boleh dihasilkan dengan menyambungkan dua klip buaya pada micro:bit – satu pada pin dan satu lagi pada pin GND.

Apabila kita sentuh pin, badan kita berfungsi sebagai konduktor dan membolehkan arus elektrik mengalir dari pin ke GND (melengkapkan litar). Kita boleh mengeklipkan konduktor-konduktor lain seperti kertas aluminium kepada klip buaya untuk bertindak sebagai penderia sentuh.

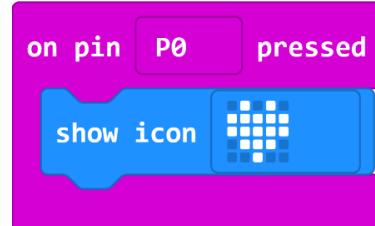


Rajah 27: Sambungan penderia sentuh

Terdapat dua cara untuk menggunakan penderia sentuh. Arur cara untuk membaca nilai pada pin adalah seperti berikut:



Cara pertama menggunakan pengaturcaraan struktur kawalan pilihan tunggal. Syarat struktur kawalan ditetapkan kepada “pin 0 ditekan”.



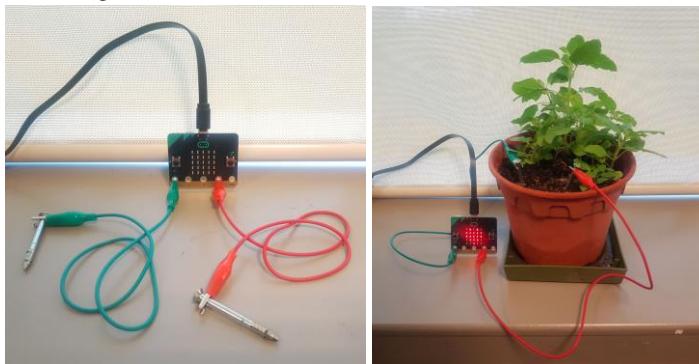
Cara kedua menggunakan pengaturcaraan berasaskan peristiwa di mana peristiwa yang dikesan adalah peristiwa apabila pin ditekan.

Walaupun berbeza cara, tetapi fungsi mereka tetap sama, iaitu micro:bit akan memaparkan bentuk hati apabila kita menyentuh kedua-dua pin dan GND. Tiada satu jawapan yang tunggal apabila menyelesaikan masalah dalam dunia pengaturcaraan. Cara pertama dan kedua dapat memenuhi syarat walaupun menggunakan pendekatan yang berlainan.

## Penderia air / kelembapan tanah

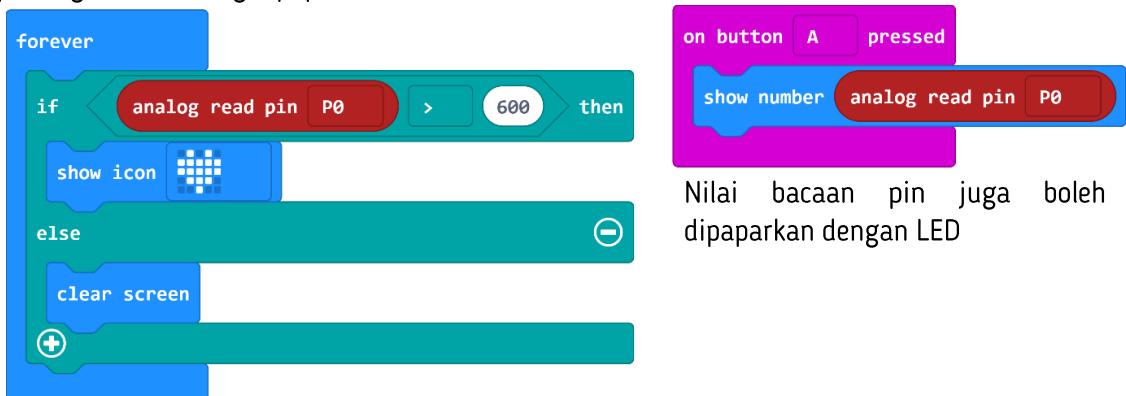
Penderia sentuh juga boleh diubah suai untuk menjadikan penderia kelembapan tanah ataupun penderia air. Prinsip fungsi adalah sama seperti penderia sentuh di mana air bertindak sebagai konduktor di antara dua pin micro:bit.

Klipkan paku pada hujung dua klip buaya dan cucukkan kelip itu ke dalam tanah.



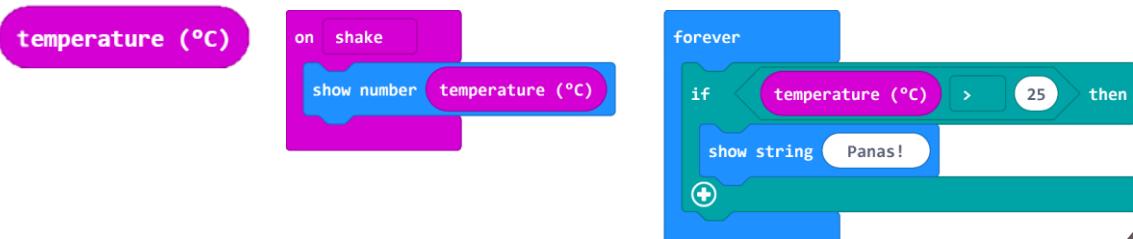
Rajah 28: Penderia air / kelembapan tanah

Untuk membaca maklumat daripada paku, kita boleh menggunakan fungsi “analog read pin” dan mendapatkan maklumat daripada itu. Micro:bit membaca nilai pada pin dan simpankan nilai tersebut dalam pemboleh ubah “analog read pin”. Pemboleh ubah ini boleh digunakan dalam blok perbandingan ataupun digunakan sebagai paparan.



## Penderia suhu

Micro:bit mempunyai penderia suhu yang dibina dalam micro:bit. Suhu dikesan dan disimpan dalam satu pemboleh ubah “temperature” yang boleh diakses dengan menggunakan blok “temperature” yang terdapat di kumpulan blok “Input”.

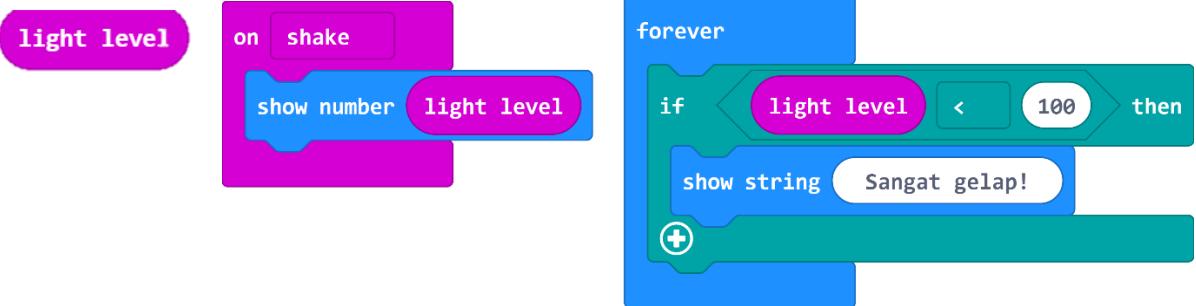


Atur cara di atas akan menunjukkan suhu bilik apabila digoncang dan jika suhu bilik melebihi 25 darjah selsius, micro:bit akan memaparkan “Panas!”

## Penderia cahaya

LED pada micro:bit juga boleh berfungsi sebagai penderia cahaya untuk mengesan keterangan di sekitar micro:bit. Nilai yang dibaca oleh penderia cahaya ini adalah dari 0 (sangat gelap) ke 255 (sangat terang).

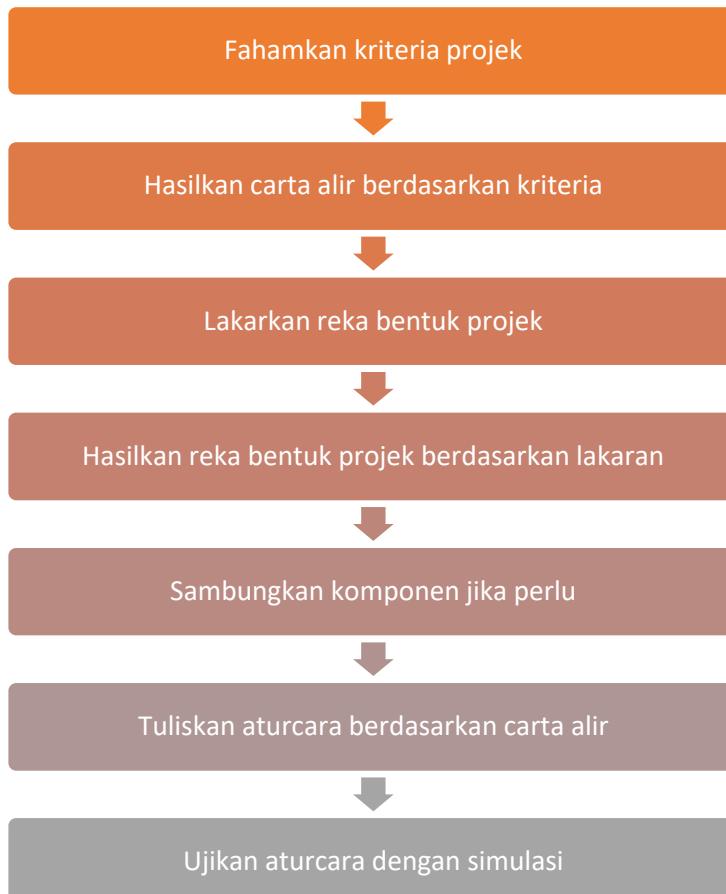
Nilai bacaan ini disimpan di pemboleh ubah “Light level” yang boleh didapatkan di kumpulan blok “Input”.



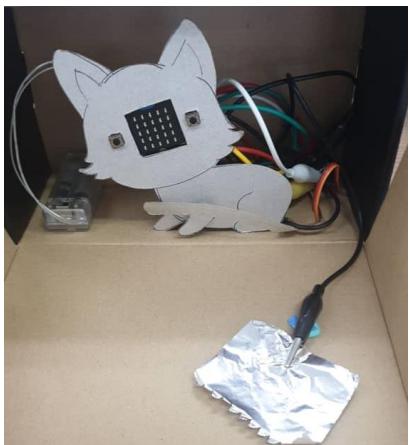
Atur cara di atas akan menunjukkan keterangan bilik apabila digoncang dan jika keterangan bilik / tahap cahaya kurang daripada 100, ia akan memaparkan “Sangat gelap!”.

## Projek DIY

Untuk menghasilkan projek DIY, kita seharusnya meingikut langkah yang tertib dan jelas seperti carta alir di bawah:

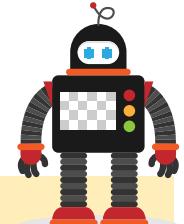


## Contoh projek yang dihasilkan (Robot Kucing):



Berikut adalah spesifikasi kucing yang kami hasilkan:

- Kucing ini boleh menggerakkan tangannya apabila ekor disentuh (penderia sentuh).
- Kucing ini mendapatkan bekalan kuasa daripada bateri
- Apabila butang A ditekan, kucing ini akan mengeluarkan muzik
- Apabila butang B ditekan, kucing ini akan memaparkan tahap cahaya di sekitarnya



## Cabaran micro:bit H1

### H1-1

Hasilkan satu robot haiwan. Haiwan micro:bit anda mesti memenuhi syarat berikut:

- Haiwan anda perlu mempunyai sekurang-kurangnya 2 fitur, iaitu sama ada bunyi, cahaya atau pergerakan.
- Haiwan anda perlu melakukan reaksi yang bersesuaian apabila butang A ditekan.
- Haiwan anda perlu melakukan reaksi yang bersesuaian apabila butang B ditekan.
- Haiwan anda perlu ada lagi satu fitur yang menggunakan penderia iaitu sama ada penderia cahaya, penderia sentuh, penderia kelembapan atau penderia suhu.
- Haiwan anda menggunakan bateri sebagai sumber kuasanya
- Badan haiwan anda diperbuat daripada bahan terbuang

### Perhatian

- 1) Untuk gunakan bateri, masukkan wayar bateri ke dalam bahagian bateri pada micro:bit. (Lihat rajah 19)
- 2) Pin bunyi (Pin 0) dan servo motor menggunakan cip pengira masa yang sama. Jadi apabila muzik dimainkan, motor akan bergerak dan sebaliknya, untuk mengatasi masalah ini, anda perlu meletakkan kesemua kod dalam satu blok kawalan pelbagai pilihan supaya timer itu dapat digunakan untuk satu tujuan sahaja pada masa yang sama. Lihat contoh di bawah:

```

forever
  if pin P1 is pressed then
    digital write pin P0 to 0
    set servo P2 angle to -28 °
    show icon
  else if button A is pressed then
    digital write pin P2 to 0
    show icon
    start melody nyan repeating once
    pause (ms) 6000
  else
    digital write pin P0 to 0
    set servo P2 angle to 58 °
    show leds
  
```



Apabila penderia sentuh mengesan sentuhan, padamkan muzik di pin 0 dengan setkan pin itu ke 0. Kemudian, gerakkan motor

Apabila butang A ditekan, setkan motor di pin 2 ke 0. Kemudian, mainkan muzik

# LAMPIRAN 1: SENARAI SEMAK CABARAN



No.	Kod Cabaran	Hasilkan carta alir	Hasilkan atur cara	Hasilkan projek	Hasilkan robot	Lengkap atau tidak / Catatan
1	Cabaran C1-1	☒	☑	☒	☒	
2	Cabaran C1-2	☒	☑	☒	☒	
3	Cabaran C2-1	☑	☒	☒	☒	
4	Cabaran C2-2	☒	☑	☒	☒	
5	Cabaran C2-3	☒	☑	☒	☒	
6	Cabaran D1-1	☒	☑	☒	☒	
7	Cabaran D1-2	☑	☒	☒	☒	
8	Cabaran D1-3	☑	☑	☑	☒	
9	Cabaran E1-1	☑	☑	☑	☒	
10	Cabaran E1-2	☑	☑	☒	☒	
11	Cabaran E1-3	☒	☑	☑	☒	
12	Cabaran E2-1	☑	☒	☒	☒	
13	Cabaran E2-2	☑	☑	☑	☒	
14	Cabaran E3-1	☑	☑	☑	☒	
15	Cabaran E4-1	☑	☑	☑	☒	
16	Cabaran E4-2	☑	☑	☑	☒	
17	Cabaran E4-3	☑	☑	☑	☒	
18	Cabaran E4-4	☑	☑	☑	☒	
19	Cabaran E4-5	☑	☑	☑	☒	
20	Cabaran E4-6	☑	☑	☑	☒	
21	Cabaran F1-1	☑	☑	☑	☒	
22	Cabaran F2-2	☑	☑	☑	☒	
23	Cabaran G1-1	☑	☑	☑	☑	
24	Cabaran H1-1	☑	☑	☑	☑	

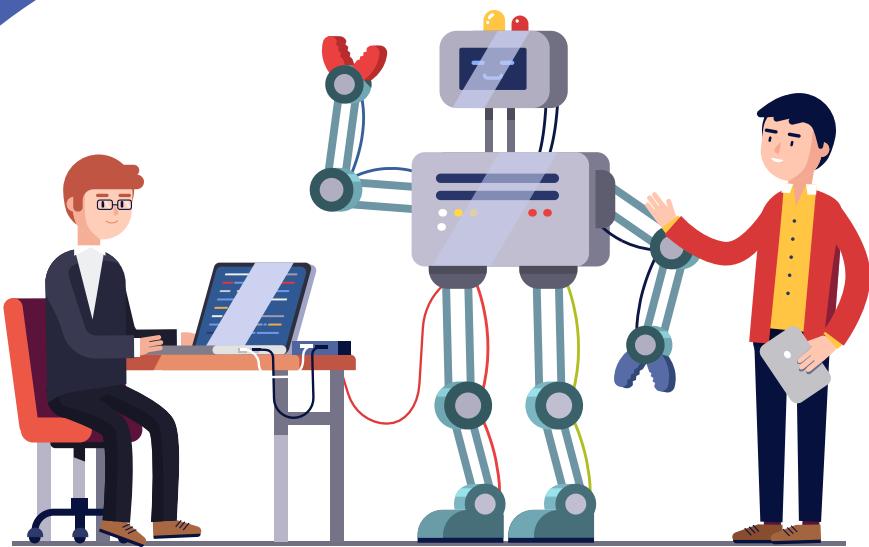
Dapatkan jawapan dengan mengimbas kod QR ini



## LAMPIRAN 2: PERBANDINGAN DSKP DAN BUKU



			Bahagian A - Pengenalan kepada mikropengawal dan Pengaturcaraan	Bahagian B - Asas Antara Muka Perisian Pengaturcaraan micro:bit	Bahagian C: Struktur Kawalan Jujukan dengan Paparan Aksona dan Imej	Bahagian D: Pengaturcaraan Berdasarkan Peristiwa dan input pada micro:bit	Bahagian E: Litar dan Komponen Luaran	Bahagian F: Pembelahan Ubah dan Struktur Kawalan Pilihan	Bahagian G: Struktur Kawalan Ulangan	Bahagian H: Pendekar-penderia Lain dan Projek DIY
Darjah 5	7.3.2	Mengenal pasti struktur kawalan pilihan dan ulangan dalam algoritma yang diberikan	<input checked="" type="checkbox"/>							
	7.3.3	Membezakan struktur kawalan jujukan, pilihan dan ulangan melalui algoritma dengan kaedah pseudokod dan carta alir	<input checked="" type="checkbox"/>							
	7.4.1	Mengenal pasti fitur-fitur pada antara muka perisian pengaturcaraan		<input checked="" type="checkbox"/>						
	7.4.2	Mengenal pasti perkakasan yang akan digunakan bersama perisian pengaturcaraan		<input checked="" type="checkbox"/>						
	7.4.3	Menghasilkan carta alir projek reka bentuk yang akan dibuat iaitu yang mengeluarkan cahaya, bunyi dan pergerakan			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
	7.4.4	Membuat sambungan perkakasan projek reka bentuk yang akan dihasilkan			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
	7.4.5	Membina atur cara yang dikehendaki berpandukan carta alir yang dibuat			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
	7.4.6	Membuat simulasi pada atur cara yang telah dibina			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
	7.4.7	Memindahkan atur cara yang telah dibina pada hardware			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
	7.4.8	Menjalankan atur cara pada perkakasan yang mengeluarkan cahaya, bunyi dan pergerakan			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
	7.4.9	Membentangkan atur cara yang telah dihasilkan					<input checked="" type="checkbox"/>			
Darjah 6	7.5.1	Menyatakan elemen sistem robotik						<input checked="" type="checkbox"/>		
	7.5.2	Membincangkan elemen sistem robotik yang terdapat pada sebuah robot						<input checked="" type="checkbox"/>		
	7.5.3	Mengenal pasti pemasangan perkakasan pada reka bentuk sebuah robot								<input checked="" type="checkbox"/>
	7.5.4	Melakar secara kreatif reka bentuk produk yang mempunyai elemen sistem robotik								<input checked="" type="checkbox"/>
	7.5.5	Menghasilkan reka bentuk produk berdasarkan laporan yang dibuat								<input checked="" type="checkbox"/>
	7.5.6	Menghasilkan atur cara dalam bentuk pseudokod atau carta alir berdasarkan produk yang dihasilkan						<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	7.5.7	Membuat simulasi pada atur cara yang telah dibina dan membuat penambahbaikan						<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	7.5.8	Memindahkan atur cara yang telah dibina pada produk yang dihasilkan						<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	7.5.9	Menguji aturcara pada produk yang telah dihasilkan						<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	7.5.10	Membuat rumusan dari projek yang dijalankan								<input checked="" type="checkbox"/>



Modul ini hasil ciptaan



Jika ada sebarang soalan/ pembetulan / cadangan penambahbaikan, sila hubungi

[enquiry@arusacademy.org.my](mailto:enquiry@arusacademy.org.my)

Tempahan kit pembelajaran micro:bit boleh dibuat pada laman web Cytron

[www.cytron.io/micro:bit](http://www.cytron.io/micro:bit) atau hubungi [sales@cytron.io](mailto:sales@cytron.io)  
(Gunakan kod TQCikgu untuk dapatkan RM10 diskaun)

Sertai program Cytron for Cikgu untuk  
dapatkan penghantaran percuma serta  
diskaun pada laman web Cytron



Sertai kumpulan telegram kami untuk  
bantuan pedagogi, teknikal dan  
maklumat terkini mengenai micro:bit



Karya ini dilesenkan di bawah Lesen Atribusi-Perkongsian Serupa  
4.0 Antarabangsa Creative Commons.

Anda bebas untuk kongsi dan adaptasi karya ini dengan syarat memberi atribusi kepada pengarang asal, mengedarkan karya ini di bawah lesen yang sama serta tidak boleh meletakkan sebarang larangan tambahan terhadap karya ini