

e. Tentukan tanggal, di mana penyumbangnya (peserta didik) paling banyak

Jawab:

Untuk menjawab pertanyaan ini, peserta didik dapat melakukan beberapa cara, antara lain:

- Cara 1

Membuat dulu daftar jumlah peserta didik yang menyumbang per tanggal, lalu catat tanggal yang memiliki jumlah paling banyak

P	Q	R	
Tanggal	Jumlah Siswa		
01-Mar-21	=COUNTIFS(\$B\$2:\$B\$16;P2)		
02-Mar-21	1		
03-Mar-21	1		
04-Mar-21	1		
09-Mar-21	2		
10-Mar-21	2		
11-Mar-21	1		
15-Mar-21	1		
16-Mar-21	2		
17-Mar-21	2		
18-Mar-21	1		

Dari hasil ini terlihat yang memiliki penyumbang paling banyak adalah tanggal 09-Mar-21, 10-Mar-21, 16-Mar-21, dan 17-Mar-21.

- Cara 2

Menggunakan formula MODE, yang akan menghasilkan kolom yang memiliki kemunculan paling banyak. Dalam Ms. Excel versi Microsoft 365, formula ini diganti menjadi MODE.SNGL dan MODE.MULT, dimana yang berbeda adalah nilai yang dihasilkan saja. Arti dari "SNGL" adalah single, dan "MULT" adalah multiple. Contohnya:

Formula =MODE.SNGL(B2:B16) akan menghasilkan 1 nilai yang pertama kali ditemukan paling banyak

Formula =MODE.MULT(B2:B16) akan menghasilkan banyak nilai yang ditemukan paling banyak

Dimana range B2:B16 adalah kolom tanggal.

Contoh hasilnya:

Single	Multiple
09-Mar-21	09-Mar-21
	10-Mar-21
	16-Mar-21
	17-Mar-21

4. Karena kode bertambah terus, seringkali dibutuhkan untuk mengetahui kode “BIS” pada posisi ke berapa [match]?

Jawab :

Posisi ditentukan memakai fungsi match.

	I	J	K	L	M
1					
2		Kode	Bantuan		
3		AMK	air minum kemasan		
4		SBI	susu bayi		
5		SLM	selimut		
6		PKN	pakaian		
7		MIE	mie instan		
8		BIS	biskuit		
9					
10		BIS	6		

5. Entah mengapa, Ketua OSIS paling sering sekali menanyakan isi sel (X,Y) ...
Formula apa yang kalian sarankan untuk menjawab pertanyaan Ketua OSIS tersebut ?

Jawab:

Gunakan langsung tanda sama dengan lalu diikuti dengan kolom dan baris, dimana nilai kolom berupa huruf alfabet dari A, B, C, dan seterusnya dan nilai baris berupa angka 1, 2, 3, dan seterusnya. Contoh:

Kolom	Baris	Formula
5 atau E	8	=E8
2 atau B	4	=B4

H. Metode Pembelajaran Alternatif

Pembelajaran pada bab ini merupakan gabungan dari model aktivitas *plugged* dan *unplugged*. Jika sekolah tidak memiliki sarana dan prasarana yang berkaitan dengan aktivitas dalam unit ini, maka pembelajaran dapat dilakukan dengan mengambil aktivitas yang *unplugged*. Alur untuk *unplugged* dijelaskan pada bagian satu buku guru ini. Materi pengembangan secara *unplugged* dapat berupa penyelesaian persoalan (*problem solving*) soal-soal yang ada pada materi berpikir komputasional.

I. Pengayaan dan Remedial

Guru memberikan pengayaan kepada peserta didik yang kecepatan belajarnya tinggi dengan memberi saran dan tugas tambahan dengan data lain yang dapat diunduh dari situs-situs data pemerintah seperti dari Badan Pusat Statistik (BPS) dengan

alamat bps.go.id.

Aktivitas pembelajaran bagi kelompok peserta didik yang memerlukan remedial bisa dikembangkan dengan melakukan pendampingan bagi peserta didik terkait topik ini. Guru dapat juga memberikan tips-tips khusus untuk memudahkan pemahaman materi.

Aktivitas pembelajaran bisa dikembangkan dengan mempelajari materi dari situs-situs yang memiliki reputasi bagus, seperti:

<https://www.datacamp.com/courses/data-visualization-in-spreadsheets>

<https://www.coursera.org/projects/data-visualization-using-google-sheets>

<https://www.optimizesmart.com/how-to-select-best-excel-charts-for-your-data-analysis-reporting/>

<https://www.academyfinancial.org/resources/Documents/Proceedings/2009/6B-Balik.pdf>

J. Asesmen dan Rubrik Penilaian

Formatif:

Penilaian formatif dilakukan tiap minggu dari aktivitas yang ada, seperti Aktivitas AD-K8-01 sampai Aktivitas AD-K8-07. Rubrik penilaiannya adalah sebagai berikut:

Indikator capaian	Baik Sekali	Baik	Cukup	Kurang
Kemampuan menerapkan fungsi pencarian data (lookup)	≥ 80% fitur yang diminta berhasil diterapkan.	60%-79% fitur yang diminta berhasil diterapkan.	40%-59% fitur yang diminta berhasil diterapkan.	Kurang dari 40% fitur yang diminta berhasil diterapkan.
Kemampuan menerapkan fungsi pencarian data (reference)	≥ 80% fitur yang diminta berhasil diterapkan.	60%-79% fitur yang diminta berhasil diterapkan.	40%-59% fitur yang diminta berhasil diterapkan.	Kurang dari 40% fitur yang diminta berhasil diterapkan.
Kemampuan menerapkan visualisasi data dengan <i>chart</i>	≥ 80% fitur yang diminta berhasil diterapkan.	60%-79% fitur yang diminta berhasil diterapkan.	40%-59% fitur yang diminta berhasil diterapkan.	Kurang dari 40% fitur yang diminta berhasil diterapkan.

Indikator capaian	Baik Sekali	Baik	Cukup	Kurang
Kemampuan melengkapi visualisasi data dengan legend, label dan title pada <i>chart</i>	≥ 80% fitur yang diminta berhasil diterapkan.	60%-79% fitur yang diminta berhasil diterapkan.	40%-59% fitur yang diminta berhasil diterapkan.	Kurang dari 40% fitur yang diminta berhasil diterapkan.
Kemampuan menerapkan peringkasan data dengan pivot tables	≥ 80% fitur yang diminta berhasil diterapkan.	60%-79% fitur yang diminta berhasil diterapkan.	40%-59% fitur yang diminta berhasil diterapkan.	Kurang dari 40% fitur yang diminta berhasil diterapkan.
Kemampuan menerapkan fungsi filter dalam pivot tables	≥ 80% fitur yang diminta berhasil diterapkan.	60%-79% fitur yang diminta berhasil diterapkan.	40%-59% fitur yang diminta berhasil diterapkan.	Kurang dari 40% fitur yang diminta berhasil diterapkan.
Kemampuan menerapkan fungsi perhitungan dalam pivot tables	≥ 80% fitur yang diminta berhasil diterapkan.	60%-79% fitur yang diminta berhasil diterapkan.	40%-59% fitur yang diminta berhasil diterapkan.	Kurang dari 40% fitur yang diminta berhasil diterapkan.
Kemampuan menerapkan tables untuk pengelolaan data	≥ 80% fitur yang diminta berhasil diterapkan.	60%-79% fitur yang diminta berhasil diterapkan.	40%-59% fitur yang diminta berhasil diterapkan.	Kurang dari 40% fitur yang diminta berhasil diterapkan.

Rubrik untuk pembuatan laporan pada aplikasi lembar kerja untuk aktivitas studi kasus analisis data.

Indikator capaian	Baik Sekali	Baik	Cukup	Kurang
Kemampuan menerapkan fungsi pencarian data (lookup)	≥ 80% fitur yang diminta berhasil diterapkan.	60%-79% fitur yang diminta berhasil diterapkan.	40%-59% fitur yang diminta berhasil diterapkan.	Kurang dari 40% fitur yang diminta berhasil diterapkan.

Indikator capaian	Baik Sekali	Baik	Cukup	Kurang
Kemampuan menerapkan peringkasan data dengan pivot tables	≥ 80% fitur yang diminta berhasil diterapkan.	60%-79% fitur yang diminta berhasil diterapkan.	40%-59% fitur yang diminta berhasil diterapkan.	Kurang dari 40% fitur yang diminta berhasil diterapkan.
Kemampuan menerapkan visualisasi data dengan <i>chart</i>	≥ 80% fitur yang diminta berhasil diterapkan.	60%-79% fitur yang diminta berhasil diterapkan.	40%-59% fitur yang diminta berhasil diterapkan.	Kurang dari 40% fitur yang diminta berhasil diterapkan.

Sumatif:

Sumatif dilakukan dengan asesmen melalui soal, seperti contoh pada uji kompetensi.

K. Jawaban Uji Kompetensi

Soal Uraian

1. Uji Kompetensi - Tantangan: Pencarian Data

Soal Uji Kompetensi berupa kasus yang harus diselesaikan oleh peserta didik.

1. Tantangan - Pencarian Data

Buatlah sebuah worksheet yang berisi data nilai Tugas, UTS, dan UAS suatu mata pelajaran seperti pada tabel berikut ini. Hitunglah nilai akhir dengan rumus: $40\% * \text{Tugas} + 30\% * \text{UTS} + 30\% * \text{UAS}$. Untuk menentukan predikat, gunakan fungsi VLOOKUP dengan mengacu tabel Predikat sehingga dihasilkan tampilan seperti pada tabel.

Batas Nilai	Predikat
0	Kurang
65	Cukup
75	Baik
85	Baik Sekali

Nama	Tugas	UTS	UAS	Nilai Akhir	Predikat
	40%	30%	30%		
Vincent	80	75	90	81.50	Baik
Erick	75	80	78	77.40	Baik
Bonita	65	70	75	69.50	Cukup
Imelda	85	90	82	85.60	Baik Sekali
Fikri	72	80	75	75.30	Baik
Mayang	90	85	90	88.50	Baik Sekali
Herlina	60	70	65	64.50	Kurang
Gunawan	80	75	85	80.00	Baik
Karlina	70	65	75	70.00	Cukup
Rahmat	78	80	85	80.70	Baik

Jawaban:

E3 $=\$B\$2*B3+\$C\$2*C3+\$D\$2*D3$

	A	B	C	D	E	F
1	Nama	Tugas	UTS	UAS	Nilai Akhir	Predikat
2		40%	30%	30%		
3	Vincent	80	75	90	81.50	Baik
4	Erick	75	80	78	77.40	Baik
5	Bonita	65	70	75	69.50	Cukup
6	Imelda	85	90	82	85.60	Baik Sekali
7	Fikri	72	80	75	75.30	Baik
8	Mayang	90	85	90	88.50	Baik Sekali
9	Herlina	60	70	65	64.50	Kurang
10	Gunawan	80	75	85	80.00	Baik
11	Karlina	70	65	75	70.00	Cukup
12	Rahmat	78	80	85	80.70	Baik

F3 $=VLOOKUP(E3; \$I\$2: \$J\$5; 2; TRUE)$

	A	B	C	D	E	F
1	Nama	Tugas	UTS	UAS	Nilai Akhir	Predikat
2		40%	30%	30%		
3	Vincent	80	75	90	81.50	Baik
4	Erick	75	80	78	77.40	Baik
5	Bonita	65	70	75	69.50	Cukup
6	Imelda	85	90	82	85.60	Baik Sekali
7	Fikri	72	80	75	75.30	Baik
8	Mayang	90	85	90	88.50	Baik Sekali
9	Herlina	60	70	65	64.50	Kurang
10	Gunawan	80	75	85	80.00	Baik
11	Karlina	70	65	75	70.00	Cukup
12	Rahmat	78	80	85	80.70	Baik

2. Peringkasan dan Visualisasi Data

Soal Uji Kompetensi berupa kasus yang harus diselesaikan oleh peserta didik.

2. Tantangan- Peringkasan dan Visualisasi Data

Buatlah sebuah *worksheet* yang memuat data jumlah kunjungan wisatawan mancanegara pada bulan Januari sampai dengan Mei ke Indonesia menurut pintu masuk udara atau laut untuk data tahun 2018, seperti contoh berikut ini.

Nama Lokasi	Jenis	Provinsi	Bulan	Jumlah
Ngurah Rai	Pintu Udara	Bali	Januari	345,191
Sultan Syarif Kasim II	Pintu Udara	Riau	Januari	2,129
Ahmad Yani	Pintu Udara	Jawa Tengah	Januari	1,680
Soekarno-Hatta	Pintu Udara	DKI Jakarta	Januari	210,494
Tanjung Pinang	Pintu Laut	Riau	Januari	8,753
Tanjung Benoa	Pintu Laut	Bali	Januari	4,626
Tanjung Mas	Pintu Laut	Jawa Tengah	Januari	2,020
Batam	Pintu Laut	Riau	Januari	115,530
Ngurah Rai	Pintu Udara	Bali	Februari	443,805
Sultan Syarif Kasim II	Pintu Udara	Riau	Februari	1,942
Ahmad Yani	Pintu Udara	Jawa Tengah	Februari	1,724
Soekarno-Hatta	Pintu Udara	DKI Jakarta	Februari	201,981
Tanjung Pinang	Pintu Laut	Riau	Februari	10,704
Tanjung Benoa	Pintu Laut	Bali	Februari	6,616
Tanjung Mas	Pintu Laut	Jawa Tengah	Februari	3,191
Batam	Pintu Laut	Riau	Februari	155,671
Ngurah Rai	Pintu Udara	Bali	Maret	484,846
Sultan Syarif Kasim II	Pintu Udara	Riau	Maret	2,510
Ahmad Yani	Pintu Udara	Jawa Tengah	Maret	2,020
Soekarno-Hatta	Pintu Udara	DKI Jakarta	Maret	244,467
Tanjung Pinang	Pintu Laut	Riau	Maret	13,261
Tanjung Benoa	Pintu Laut	Bali	Maret	4,254

Jawaban

- a. PivotTable1 dengan Nama Lokasi sebagai Rows dan bulan sebagai Columns, kemudian Jenis pintu sebagai Filters, dan Jumlah sebagai Values, sehingga hasilnya seperti berikut ini (dalam contoh dipergunakan Jenis pintu: udara).

Sum of Jumlah		Column Labels					
Row Labels	April	Februari	Januari	Maret	Mei	Grand Total	
Ahmad Yani	2232	1724	1680	2020	1612	9268	
Ngurah Rai	516143	443805	345191	484846	526281	2316266	
Soekarno-Hatta	224978	201981	210494	244467	191631	1073551	
Sultan Syarif Kasim II	2044	1942	2129	2510	1666	10291	
Grand Total	745397	649452	559494	733843	721190	3409376	

PivotTable Fields

Choose fields to add to report:

Search

- Nama Lokasi
- Jenis
- Provinsi
- Bulan
- Jumlah

More Tables...

Drag fields between areas below:

Filters

Jenis

Columns

Bulan

Rows

Nama Lokasi

Σ Values

Sum of Jumlah

- b. PivotTable2 dengan Nama Lokasi sebagai Rows dan bulan sebagai Columns, kemudian Provinsi sebagai Filters, dan Jumlah sebagai Values sehingga hasilnya seperti contoh berikut. (dalam contoh provinsi yang digunakan: Riau dan Bali).

Provinsi	(Multiple Items) ✓					
Sum of Jumlah	Column Labels					
Row Labels	April	Februari	Januari	Maret	Mei	Grand Total
Batam	144104	155671	115590	175194	142157	732716
Ngurah Rai	516143	443805	345191	484846	526281	2316266
Sultan Syarif Kasim II	2044	1942	2129	2510	1666	10291
Tanjung Benoa	634	8616	4626	4254	2231	20361
Tanjung Pinang	10449	10704	8753	13261	11268	54435
Grand Total	673374	620738	476289	680065	683603	3134069

PivotTable Fields

Choose fields to add to report: ⚙️

Search 🔍

- Nama Lokasi
- Jenis
- Provinsi
- Bulan
- Jumlah

More Tables...

Drag fields between areas below:

<p style="margin: 0;">▼ Filters</p> <p style="margin: 0;">Provinsi</p>	<p style="margin: 0;">☰ Columns</p> <p style="margin: 0;">Bulan</p>
<p style="margin: 0;">☰ Rows</p> <p style="margin: 0;">Nama Lokasi</p>	<p style="margin: 0;">Σ Values</p> <p style="margin: 0;">Sum of Jumlah</p>

- c. Chart berdasarkan hasil PivotTable1.

- 1) Chart Pie untuk jumlah wisatawan total yang masuk melalui pintu udara.



- 2) Buatlah chart Column untuk jumlah wisatawan dari bulan Januari sampai dengan Mei melalui pintu udara.



3. Uji Kompetensi - Tantangan: Pengelolaan Data

Soal Uji Kompetensi berupa kasus yang harus diselesaikan oleh peserta didik

3. Tantangan - Pengelolaan Data

Program *Wonderful* Indonesia sedang giat memasarkan berbagai daerah pariwisata di Indonesia dan berusaha agar wisatawan mancanegara lebih banyak datang. Kita tertarik untuk mempelajari data wisatawan yang masuk ke Indonesia melalui bandara atau pelabuhan laut.

Rancanglah sebuah lembar kerja yang menampung data nama bandara/pelabuhan, jenisnya (pintu udara/laut), provinsi di mana bandara/pelabuhan laut tersebut berada, dan jumlah wisatawan yang berkunjung per bulan.

Dari lembar kerja tersebut, gunakan pengelolaan data dengan *Tables* dan *Filter* sehingga kita dapat menganalisis data tersebut dengan lebih mudah, misalnya dengan menjawab pertanyaan seperti berikut.

- Bagaimana menampilkan data pada suatu provinsi/bandara/pelabuhan pada bulan-bulan tertentu?
- Bagaimana caranya agar data ditampilkan secara terurut berdasarkan nama provinsi atau berdasarkan jumlah wisatawan yang datang?
- Bagaimana menampilkan jumlah wisatawan yang terbanyak, terkecil, atau rata-rata dari data yang ada?
- Bagaimana menampilkan jumlah wisatawan yang terbanyak, terkecil, atau rata-rata pada setiap bulan?

Guru akan memberikan datanya setelah lembar kerja kalian selesai.

Nama Lokasi	Jenis	Provinsi	Bulan	Jumlah
Ngurah Rai	Pintu Udara	Bali	Januari	345,191
Sultan Syarif Kasim II	Pintu Udara	Riau	Januari	2,123
Ahmad Yani	Pintu Udara	Jawa Tengah	Januari	1,680
Soekarno-Hatta	Pintu Udara	DKI Jakarta	Januari	210,494
Tanjung Pinang	Pintu Laut	Riau	Januari	8,753
Tanjung Benoa	Pintu Laut	Bali	Januari	4,626
Tanjung Mas	Pintu Laut	Jawa Tengah	Januari	2,020
Batam	Pintu Laut	Riau	Januari	115,590
Ngurah Rai	Pintu Udara	Bali	Februari	443,805
Sultan Syarif Kasim II	Pintu Udara	Riau	Februari	1,942
Ahmad Yani	Pintu Udara	Jawa Tengah	Februari	1,724
Soekarno-Hatta	Pintu Udara	DKI Jakarta	Februari	201,301
Tanjung Pinang	Pintu Laut	Riau	Februari	10,704
Tanjung Benoa	Pintu Laut	Bali	Februari	8,616
Tanjung Mas	Pintu Laut	Jawa Tengah	Februari	3,191
Batam	Pintu Laut	Riau	Februari	155,671
Ngurah Rai	Pintu Udara	Bali	Maret	484,846
Sultan Syarif Kasim II	Pintu Udara	Riau	Maret	2,510
Ahmad Yani	Pintu Udara	Jawa Tengah	Maret	2,020
Soekarno-Hatta	Pintu Udara	DKI Jakarta	Maret	244,461

- a. Data didefinisikan dahulu sebagai Tables, kemudian gunakan filter untuk memilih provinsi/bandar/pelabuhan yang akan ditampilkan. Sebagai contoh, berikut ini tampilan untuk Provinsi Batam.

Nama Lokasi	Jenis	Provinsi	Bulan	Jumlah
Batam	Pintu Laut	Riau	Maret	175,194
Batam	Pintu Laut	Riau	Februari	155,671
Batam	Pintu Laut	Riau	April	144,104
Batam	Pintu Laut	Riau	Mei	142,157
Batam	Pintu Laut	Riau	Januari	115,590

- b. Data pada Tables diurutkan berdasarkan jumlah wisatawan dan nama provinsi memakai fitur sorting pada Tables. Tampilan berikut disusunurut berdasarkan nama provinsi (*Ascending*) dan jumlah wisatawan (*Descending*).

Nama Lokasi	Jenis	Provinsi	Bulan	Jumlah
Ngurah Rai	Pintu Udara	Bali	Mei	526,281
Ngurah Rai	Pintu Udara	Bali	April	516,143
Ngurah Rai	Pintu Udara	Bali	Maret	484,846
Ngurah Rai	Pintu Udara	Bali	Februari	443,805
Ngurah Rai	Pintu Udara	Bali	Januari	345,191
Soekarno-Hatta	Pintu Udara	DKI Jakarta	Maret	244,467
Soekarno-Hatta	Pintu Udara	DKI Jakarta	April	224,978
Soekarno-Hatta	Pintu Udara	DKI Jakarta	Januari	210,494
Soekarno-Hatta	Pintu Udara	DKI Jakarta	Februari	201,981
Soekarno-Hatta	Pintu Udara	DKI Jakarta	Mei	191,631
Batam	Pintu Laut	Riau	Maret	175,194
Batam	Pintu Laut	Riau	Februari	155,671
Batam	Pintu Laut	Riau	April	144,104
Batam	Pintu Laut	Riau	Mei	142,157
Batam	Pintu Laut	Riau	Januari	115,590
Tanjung Pinang	Pintu Laut	Riau	Maret	13,261
Tanjung Pinang	Pintu Laut	Riau	Mei	11,268
Tanjung Pinang	Pintu Laut	Riau	Februari	10,704
Tanjung Pinang	Pintu Laut	Riau	April	10,449

- c. Gunakan perhitungan total pada Tables dengan fungsi min untuk jumlah terkecil, max untuk jumlah terbanyak, dan average untuk rata-rata.

Terbanyak	526,281
Terkecil	137
Rata-rata	105,742

Tanjung Mas	Pintu Laut	Jawa Tengah	April	437
Tanjung Mas	Pintu Laut	Jawa Tengah	Mei	137
Total				526,281
				None
				Average
				Count
				Count Numbers
				Max
				Min
				Sum
				StdDev
				Var
				More Functions...

- d. Gunakan filter untuk memilih bulan, kemudian gunakan perhitungan total pada Tables dengan fungsi min untuk jumlah terkecil, max untuk jumlah terbanyak, dan average untuk rata-rata. Salinlah hasil yang diperoleh ke dalam tabel seperti dalam solusi jawaban berikut ini.

Bulan	Terbanyak	Terkecil	Rata-rata
Januari	345,191	1,680	86,310
Februari	443,805	1,724	103,454
Maret	484,846	2,020	116,693
April	516,143	437	112,628
Mei	526,281	137	109,623

L. Interaksi Guru dan Orang Tua/Wali

Orang tua/wali hendaknya selalu aktif dalam mengawasi anaknya ketika melakukan aktivitas dengan komputer. Guru dapat berinteraksi dengan memberikan informasi dan tips bagi orang tua dalam hal penggunaan perkakas aplikasi lembar kerja yang ada sehingga orang tua juga dapat membantu anaknya ketika mengalami kesulitan.

M. Refleksi Guru

No	Aspek	Pertanyaan
1	Tujuan pembelajaran	Apakah tujuan pembelajaran sudah tercapai?
2	Proses kegiatan belajar mengajar	Apakah kegiatan belajar mengajar yang dilakukan sudah berhasil dengan baik?
3	Materi/konten pelajaran	Apakah ketepatan, kedalaman dan keluasan materi yang saya sampaikan sudah cukup untuk mencapai Tujuan Pembelajaran?
4	Kondisi peserta didik	Apakah semua peserta didik dapat mengikuti aktivitas pembelajaran dengan baik?
5	Kesulitan belajar peserta didik	Apakah ada peserta didik yang mengalami kesulitan dalam belajar materi ini?
6	Minat belajar	Apakah ada peserta didik yang memiliki minat belajar lebih dan berkeinginan mengeksplorasi lebih dari materi ini?
7	Efektivitas metode pembelajaran	Seberapa efektifkah metode pembelajaran yang sudah Anda gunakan?
8	Variasi pembelajaran	Apakah saya memiliki ide lain untuk mengembangkan materi ini?

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
REPUBLIK INDONESIA, 2021

Buku Panduan Guru Informatika untuk SMP Kelas VIII

Penulis : Maresha Caroline Wijanto, Natalia, Husnul Hakim, dan Kurniawan Kartawidjaja
ISBN : 978-602-244-719-1

Bab

7

Algoritma dan Pemrograman



Unit pembelajaran ini bertujuan untuk melanjutkan pemrograman visual dengan bahasa Scratch yang sudah dialami peserta didik di kelas VII, dengan lebih menekankan praktik berpikir komputasional. Selain pendalaman aspek berpikir komputasional dalam pemrograman, berdasarkan pemrograman visual menggunakan Scratch, peserta didik diajak untuk praktik pemrograman visual menggunakan bahasa lain, yang di buku ini dipilih Blockly, yaitu sebuah bahasa pemrograman visual yang berbeda, tetapi prinsipnya sama dengan Scratch yang telah dipakai praktik di kelas VII. Kalau dianalogikan dengan menyetir mobil, seseorang yang dapat menyetir mobil merek A pada umumnya juga akan dapat menyetir mobil merek B karena semua mobil prinsipnya sama. Kemampuan transversal antarbahasa pemrograman visual, dengan

menggunakan sebelumnya akan diasah pada algoritma pemrograman kelas VIII. Oleh karena itu, jika peserta didik belum mengalami pelajaran pemrograman kelas VII dengan seutuhnya, guru perlu menyesuaikan materinya kalau hendak memakai buku ini. Peserta didik akan melakukan serangkaian aktivitas yang dirancang dengan kesulitan bertingkat untuk membiasakan diri dengan lingkungan pemrograman visual ini.

A. Tujuan Pembelajaran

Tujuan Pembelajaran untuk elemen Algoritma dan Pemrograman di kelas VIII adalah seperti berikut.

1. Peserta didik mampu memakai fitur bahasa pemrograman visual yang belum dipelajari di kelas VII. Dalam hal ini fitur lanjut dari bahasa pemrograman Scratch:
 - a. Membuat program yang mengandung variabel.
 - b. Membuat custom block yang pada hakikatnya dipakai sebagai prosedur pada Scratch.
2. Peserta didik mampu memprogram dalam bahasa pemrograman visual kedua yang mirip dengan Scratch, yaitu Blockly, dalam sebuah lingkungan pemrograman blok/visual yang dikemas dalam bentuk permainan.
3. Peserta didik mampu membaca dan memahami makna blok penyusun program dalam bahasa Blockly:
 - a. Variabel, input, *output*
 - b. Ekspresi matematika, ekspresi logika dan perhitungannya
 - c. Percabangan
 - d. Pengulangan
4. Peserta didik mampu menyusun kode program Blockly
 - a. Melakukan *drag and drop* blok pemrograman yang tersedia untuk menyusun sebuah program.
 - b. Menjalankan dan melihat hasil eksekusi program yang dibuat.
5. Peserta didik mampu menyelesaikan persoalan dengan menyusun program prosedural dengan bahasa Blockly:
 - a. Membuat spesifikasi input, *output*, proses
 - b. Menganalisis dan mengembangkan solusi

- c. Menyusun kode program yang sesuai :
 - i. Melakukan drag and drop blok pemrograman yang tersedia untuk menyusun sebuah program.
 - ii. Menjalankan dan melihat hasil eksekusi program yang dibuat.
 - iii. Membuat program yang menerima input, dan menyimpannya dalam sebuah variabel.
6. Memahami dan mengenal cara kerja robot “*line follower*” dan mengeksplorasi perilaku robot.

B. Kata Kunci

Pemrograman visual, algoritma; pemrograman; Scratch; Blockly; pemrograman prosedural, variabel, input, *output*, ekspresi aritmetika, ekspresi logika, percabangan; perulangan.

C. Kaitan dengan Bidang Pengetahuan lain

Algoritma dan Pemrograman (AP) berkaitan dengan elemen lain Informatika, yaitu Berpikir Komputasional, Sistem Komputer serta Teknologi Informasi dan Komunikasi. Pengetahuan ini juga digunakan pada informatika dalam pengembangan artefak komputasional. Pengembangan artefak komputasional harus dikembangkan dengan praktik yang berkaitan dengan pengetahuan AP. AP akan membantu peserta didik dalam membentuk pola pikir logika penyelesaian masalah.

D. Organisasi Pembelajaran

Materi	Lama Waktu (JP)	Tujuan Pembelajaran	Aktivitas
Variabel dan Custom Block Scratch	2	A-1	AP-K8-01: Bermain dengan Control, Input dan Variable Aktivitas AP-K8-02: Bermain dengan Custom Block
Program Puzzle Maze dalam Blockly	2	A-2	Aktivitas AP-K8-03: Eksplorasi Maze
Music dalam Blockly	2	A-2	Aktivitas AP-K8-04: Ekplorasi Music

Materi	Lama Waktu (JP)	Tujuan Pembelajaran	Aktivitas
"Sprites" dalam Blockly	2	A-2, A-4	Aktivitas AP-K8-05: Eksplorasi <i>Games Move a sprite</i> Aktivitas AP-K8-06: <i>Customize Games Move a sprite</i>
Pemrograman Prosedural	2	A-3, A-5	Aktivitas AP-K8-07: <i>Hello World</i> Aktivitas AP-K8-08: Hello Namaku
<i>Problem solving</i> dengan solusi Pemrograman Prosedural	2	A-3, A-5	Aktivitas AP-K8-09: Print Pola 1 sampai N Tanda Bintang Aktivitas AP-K8-10: Print Pola N sampai 1 Tanda Bintang Aktivitas AP-K8-11: Print Pola Diamond
Robot <i>Line Follower</i> Ozobot	2	A-6	Aktivitas AP-K8-12-U: Garis Lajur Ozobot Aktivitas AP-K8-13-U: Lajur Warna Ozobot Aktivitas AP-K8-14-U: Kode untuk Mengatur Kecepatan Ozobot

E. Pengalaman Belajar Bermakna, Profil Pelajar Pancasila, Berpikir Komputasional dan Praktik Inti

Pengalaman belajar bermakna	Profil Pelajar Pancasila	Berpikir Komputasional	Praktik Inti
Peserta didik memahami lingkungan <i>tools</i> pemrograman visual.	Mandiri, Bernalar Kritis	Abstraksi	Memahami <i>tools</i>
Peserta didik mengeksplorasi <i>tools</i> pemrograman visual sesuai panduan.	Mandiri, Bernalar Kritis	Abstraksi, Algoritma, Dekomposisi, dan Pengenalan pola	Mengimplementasikan <i>tools</i> yang sesuai
Peserta didik mengeksplorasi penggunaan variabel dan menerima input pada <i>tools</i> pemrograman visual sesuai panduan.	Mandiri, Bernalar Kritis, Kreatif	Abstraksi, Dekomposisi, Algoritma	Mengimplementasi variabel dan input
Peserta didik mengeksplorasi penggunaan percabangan pada <i>tools</i> pemrograman visual sesuai panduan.	Mandiri, Bernalar Kritis, Kreatif	Abstraksi, Dekomposisi, Algoritma	Mengimplementasi percabangan
Peserta didik mengeksplorasi penggunaan perulangan pada <i>tools</i> pemrograman visual sesuai panduan.	Mandiri, Bernalar Kritis, Kreatif	Abstraksi, Dekomposisi, Algoritma	Mengimplementasi perulangan
Peserta didik mengeksplorasi pemrograman visual dalam permainan.	Mandiri, Bernalar Kritis, Kreatif	Abstraksi, Dekomposisi, Algoritma, dan Pengenalan pola	Menyelesaikan kasus pemrograman visual dalam permainan
Peserta didik berkolaborasi dan berdiskusi mengembangkan kasus pemrograman visual.	Gotong Royong, Bernalar Kritis, Kreatif	Abstraksi, Algoritma, Dekomposisi, dan Pengenalan pola	Menyelesaikan kasus pemrograman visual
Peserta didik berkolaborasi dan berdiskusi memanfaatkan robot Ozobot.	Gotong Royong, Bernalar Kritis, Kreatif	Abstraksi, Pengenalan pola	Eksplorasi robot Ozobot

F. Strategi Pembelajaran

Hal paling utama yang perlu diperhatikan oleh guru yang memakai buku Kelas VIII khususnya bab Algoritma dan Pemrograman ini adalah bahwa peserta didik sebaiknya sudah menyelesaikan pembelajaran Algoritma dan Pemrograman kelas VII karena bahasa dan kasus yang dipakai pada buku kelas VIII ini berkesinambungan dengan materi kelas VII. Jika peserta didik anda belum pernah mengerjakan latihan-latihan setara dengan materi yang disajikan di kelas VII, disarankan untuk memakai materi kelas VII.

1. Strategi Pembelajaran Algoritma dan Pemrograman secara *Plugged*

Mata pelajaran Informatika terkait erat dengan materi Algoritma dan Pemrograman yang merupakan salah satu elemen dari Berpikir Komputasional. Selain itu, keterbatasan komputer serta jaringan menyebabkan jam pelajaran lebih banyak bisa diisi dengan konsep informatika yang tidak terlalu membutuhkan biaya. Banyak *platform* perangkat pengembangan dan *tools* berupa perangkat lunak yang tersedia gratis, tetapi tidak demikian dengan perangkat keras dan jaringan komunikasi. Pola berpikir Algoritma dan Pemrograman ini yang membentuk penyelesaian masalah. Aktivitas yang disediakan sudah ada konsep dan contohnya. Guru juga dapat mencoba bersama peserta didik di kelas. Di level sekolah menengah, pemrograman yang digunakan adalah pemrograman visual. Diharapkan, peserta didik dapat belajar pemrograman dengan pendekatan yang menyenangkan, dan menumbuhkan aspek kreatif. Guru juga dapat menjelaskan konsep pemrograman dengan lebih mudah. Bahasa pemrograman yang digunakan mula-mula adalah Scratch, kemudian berdasarkan metapengetahuan pemrograman Scratch, diharapkan peserta didik dengan cepat belajar Blockly.

Selain melakukan aktivitas dengan mengeksplorasi sendiri dibantu dengan langkah jelas dan pertanyaan-pertanyaan yang ada, peserta didik mulai belajar membuat model solusi, yaitu mendefinisikan *Input-Output-Proses* dan menganalisis persoalan. Di sini, aspek Berpikir Komputasional dipraktikkan, terutama melakukan dekomposisi, abstraksi, dan memakai serta membentuk pola solusi berdasarkan solusi yang pernah dikerjakan. Oleh karena itu, rangkaian contoh soal harus memungkinkan peserta didik melakukan konstruksi pola. Rangkaian contoh harus dipilih dengan cermat dengan mempertimbangkan kemiripan dan progression tingkat kesulitan persoalan. Jika guru akan mengganti kasus, dimohon memperhatikan hal ini.

2. Aspek Kreatif Guru

Aspek kreatif yang akan dilakukan guru yang diharapkan ialah untuk mengembangkan kasus-kasus persoalan yang diprogram, yang sesuai dengan konteks kehidupan. Penggantian kasus hendaknya tetap memperhatikan himpunan contoh berkesinambungan yang membentuk suatu pola solusi, dan paradigma pemrograman yang dipilih.

3. Kreatif lewat *Tinkering & Makers*

Telah dijelaskan pada buku kelas VII bahwa pemrograman di tingkat SMP bertujuan untuk membentuk pola pikir CT melalui karya kreatif. Paradigma yang dipilih adalah paradigma pemrograman visual karena memudahkan anak mengomposisi program tanpa disibukkan dengan kesalahan sintaks seperti dalam pemrograman tekstual.

Penyusunan blok kode program visual diharapkan menjadi analogi kegiatan unplugged menyusun balok (atau kartu), dan membangkitkan proses “*tinkering*”, serta membentuk pola pikir (*mindset*) sebagai “*makers*”. Apa itu *makers*? *Makers* yang diusulkan oleh Dale Dougherty adalah pendekatan mengajarkan STEM melalui *Problem Based Learning* (PBL), yang lebih menekankan pengalaman pembelajaran langsung, seringkali kolaboratif, sebagai metode untuk penyelesaian persoalan yang otentik. Pada kegiatan pemrograman visual, peserta didik membangun kode dengan memilih blok, mencoba, mencocokkan, memasang, mengeksekusi untuk menghasilkan karya kreatif yang merupakan tujuan tugas, persoalan yang harus diselesaikan. Pendidikan “*makers*” menekankan pentingnya pengalaman yang digerakkan oleh peserta didik, pembelajaran interdisipliner (untuk mendapatkan berbagai sudut pandang solusi), proses pembelajaran *peer-to-peer*, iterasi, dan gagasan tidak takut gagal untuk maju atau gagasan bahwa pembelajaran berbasis kesalahan sangat penting untuk proses pembelajaran dan kesuksesan sebuah proyek. Guru perlu memberi peserta didik kesempatan untuk ini.

4. Kegiatan Pemrograman

Pemrograman visual di jenjang SMP lebih ditujukan untuk memakai *platform* bahasa pemrogramannya guna menghasilkan program sederhana, di mana proses analisis dan pencarian desain solusi dapat dilakukan menyatu dengan menuliskan kodenya. Menuliskan kode tidak selalu langsung harus benar, tetapi perlu dicoba sampai mencapai tujuan (*debugging*). Inilah perbedaan dengan pemrogram profesional yang harus memikirkan solusi formal dan tidak mencoba-coba. Proses mencoba-coba dan mengamati hasil ini adalah proses *tinkering*, yaitu berpikir yang dituangkan menjadi tindakan yang berefek pada eksekusi objek yang diprogram. Oleh sebab itu, lingkungan pemrograman visual merupakan sebuah lingkungan yang interaktif. Peserta didik mengamati benar atau salahnya sebuah solusi (karya) dengan mengamati gerakan yang mewakili eksekusi program. Kegiatan *tinkering* akan memupuk kemampuan berpikir kreatif jika peserta didik dibiarkan melakukan eksplorasi sendiri, mencari solusi dan bukan hanya disodori oleh kode yang hanya tinggal dijalankan. Kemampuan berpikir kritis akan dibentuk melalui perencanaan variasi-variasi percobaan dengan mengubah-ubah kode, untuk menghasilkan sesuatu yang makin kompleks, dan efisien.

Koding adalah proses kreatif yang dilakukan oleh “pemrogram komputer” (anak-anak) untuk memberitahu komputer cara melakukan suatu tugas. Koding yang dilakukan pada tingkat SMP sebaiknya diisi dengan membuat konten yang mendidik dan menghibur. Karena telah dibuktikan bahwa *game* akan memacu semangat belajar, dan anak gemar bermain, tidak ada salahnya anak belajar melalui bermain *game*, dan bahkan mengajak anak untuk membuat *game* sendiri. Banyak ide untuk membuat *game* yang dapat dicari oleh guru di internet, misalnya hanya dengan *googling* dengan kata kunci “*scratch game project*”.

Pemrograman juga dipakai sebagai wahana melatih *problem solving* dan membangun *engineering mindset*, dimana peserta didik belajar untuk menjadi pengembang artefak komputasional, dengan melakukan tahapan: analisis, desain solusi, implementasi, *debugging* dan *testing*, kemudian mengomunikasikan karyanya. Setiap tahapan menghasilkan suatu artefak. Sebagai contoh, hasil dari analisis dan spesifikasi ialah sketsa solusi dalam bentuk algoritma, *flowchart*, atau artefak desain lainnya, yang dibuat sebelum kode program ditulis. Aspek menyeluruh untuk menghasilkan solusi suatu persoalan yang kompleks dan terencana dalam skala lebih besar akan dipraktikkan di modul Praktik Lintas Bidang (PLB).

5. Platform dan Bahasa yang Dipilih

Dua buah bahasa pemrograman visual yang dipilih untuk SMP adalah Scratch dan Blockly. Scratch sangat mudah dipakai, bahkan jika internet tersedia, tidak perlu menginstal Blockly di komputer lokal. Blockly juga mudah dipakai, ada beberapa pilihan cara untuk akses. Namun sesungguhnya, ada banyak bahasa pemrograman visual lainnya yang juga boleh Anda pilih, jika Anda memang lebih memahaminya. Mengajarkan ke peserta didik lebih dari satu bahasa akan membentuk pola pikir bahwa berkarya lewat memprogram itu tidak terikat kepada satu bahasa saja. Jika sudah menguasai satu bahasa, akan lebih mudah menguasai bahasa lainnya. Ini menakjubkan dan dapat dibuktikan. Selain Scratch dan Blockly, bahasa lain yang dapat diajarkan misalnya: tynker, Alice, squeak, dan masih banyak lagi lainnya.

Bahasa Blockly dipakai sebagai jembatan untuk pemrograman tekstual Python. Python saat ini merupakan salah satu bahasa yang banyak dipakai untuk pengajaran tingkat SMP, jika di SD peserta didik sudah mengenal Scratch. Guru perlu mempertimbangkan bahasa Python jika kondisinya memang demikian. Buku ini ditulis dengan asumsi bahwa peserta didik mulai belajar pemrograman di tingkat SMP kelas VII.

6. Contoh Kasus Serial

Bagian penting dari perencanaan pengajaran pemrograman ialah mencari satu seri contoh mengajarkan anak untuk membangun program secara bertahap (*incremental*

development). Konsep tersebut merangkai seluruh konsep pemrograman mulai dari pertemuan pertama sampai dengan pertemuan terakhir, dengan kasus yang mulai sangat sederhana, menengah dan makin kompleks. Kemampuan guru dalam merangkai tugas percobaan memprogram dan menumbuhkan kreativitas peserta didik dalam menghasilkan karya, akan menentukan keberhasilan peserta didik belajar programming. Jadi, pemrograman tidak berorientasi kepada produk, tetapi lebih kepada proses. Rancanglah satu seri latihan yang saling berkaitan, dan ajak peserta didik untuk mengembangkan produk komputasional yang berkembang seiring dengan perkembangan proses pembelajaran. Tentukan semua sekuen pembelajaran agar guru mempunyai alur mengajar yang nyaman bagi anak untuk melakukan perjalanan belajar pemrograman dengan cara yang menyenangkan. Contoh yang diberikan dalam buku ini sekadar agar guru dapat memulai mengajar pemrograman. Selanjutnya, guru dapat menyusun program belajar untuk mencapai tujuan pembelajaran yang sama dengan mengadaptasi sebagian, melakukan re-mixing dengan bahan yang guru dapat di internet, dan melengkapinya sesuai dengan kondisi sekolah serta minat peserta didik.

7. Pemrograman secara *Unplugged*

Bagi guru yang harus mengajar di daerah yang sulit koneksi jaringan dan bahkan tidak ada komputer, belajar pemrograman visual dapat dilakukan tanpa komputer. Peserta didik diminta menyusun program dari kartu, di mana setiap kartu akan merepresentasikan sebuah blok. Sebaiknya, blok ini semirip mungkin dengan blok bahasa Scratch dan Blockly yang dipakai pada buku ini. Guru dapat membuat dan mencetak kartu tersebut, serta mengajak peserta didik menyusun program. Walaupun ada keterbatasan eksekusi, cara ini masih lumayan untuk dilakukan, ketimbang tidak ada praktik pemrograman. Ada sebuah situs yang mendedikasikan untuk pembelajaran pemrograman visual secara unplugged yang dapat menginspirasi guru, yaitu seperti diberikan pada referensi "*Scratch unplugged*".

G. Panduan Pembelajaran

Materi Algoritma dan Pemrograman dilakukan dalam 7 pertemuan.

1. Pertemuan 1: Eksplorasi Lanjutan Scratch (2 jp)

Tujuan Pembelajaran:

Peserta didik mampu memakai fitur bahasa pemrograman visual yang belum dipelajari di kelas VII. Dalam hal ini, fitur lanjut dari bahasa pemrograman Scratch:

- a. membuat program yang mengandung variabel,
- b. membuat custom block yang pada hakekatnya dipakai sebagai prosedur pada Scratch.

Apersepsi

Setelah mempelajari Scratch di kelas VII, peserta didik diajak untuk melakukan eksplorasi lanjutan pada Scratch. Untuk melakukan pengulangan, dapat menggunakan kode Control. Kode ini sendiri ada beberapa jenis, ada yang memiliki batasan dan ada yang tidak terbatas. Peserta didik harus memahami kapan penggunaan yang tepat dari masing-masing kode Control tersebut. Di kelas VIII, juga peserta didik akan belajar tentang menerima input pada program. Input ini disimpan dalam sebuah variabel dan dapat digunakan pada kode program sebagai parameter yang membuat kode program menjadi lebih dinamis. Variabel sendiri dapat dibentuk tanpa melalui proses input dari pengguna. Pada Scratch, peserta didik juga dapat membuat blok sendiri yang fungsinya disesuaikan dengan kebutuhan, yaitu custom block. Apabila ada kode program melakukan hal yang sama berulang kali, simpan dalam custom block dan cukup blok baru ini saja yang dipanggil. Custom block pada Scratch juga dapat dianggap sebagai prosedur dalam pemrograman pada umumnya.

Misalnya, dalam sebuah *games*, pada awalnya, program meminta pengguna untuk memasukkan nama yang dianggap sebagai variabel input. Dalam prosesnya, ada skor yang diperoleh pengguna. Skor ini dianggap sebagai variabel biasa. Dengan meminta pengguna memasukkan namanya, program seakan-akan menjadi ramah dan pengguna merasa mendapat perlakuan personal dan disapa melalui namanya.

Pemanasan

Guru dapat membuat proyek sederhana pada Scratch yang menggunakan kode Control, Input, Variable, dan Custom block. Guru dan peserta didik dapat mencoba mengeksekusi bersama proyek ini. Guru menunjukkan bagaimana perubahan input dapat berpengaruh pada hasil program. Setelah itu, guru menjelaskan skenario aktivitas yang akan dilakukan.

Kebutuhan Sarana dan Prasarana

- Komputer/*Smartphone* yang *ter-install* sistem operasi dan *browser*
- Koneksi internet

Kegiatan Inti

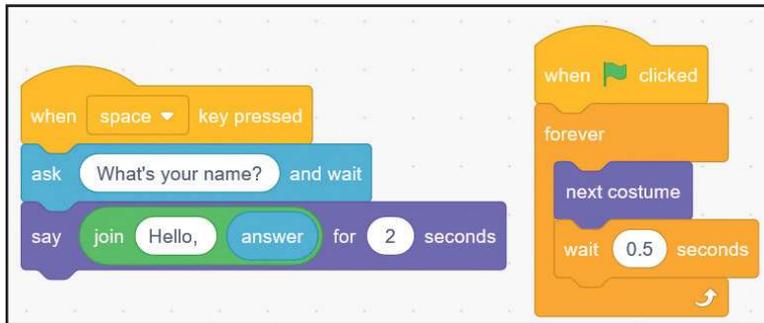
Guru menjelaskan tentang konsep Control, fungsi input dan variable serta custom block atau pada pemrograman dikenal dengan prosedur. Penjelasan dapat dilakukan dengan mencoba kode program yang sudah dibuat terlebih dulu oleh guru sebagai pemanasan. Setelah itu, guru memandu peserta didik untuk mengerjakan aktivitas AP-K8-01 Bermain dengan Control, Input, dan Variable serta AP-K8-02 Bermain dengan Custom block pada Buku Siswa. Waktu keseluruhan sekitar secara

mandiri 80 menit (2 jp). Setelah peserta didik mengerjakan secara mandiri, guru dapat membahas bersama aktivitas tersebut dengan peserta didik di kelas.

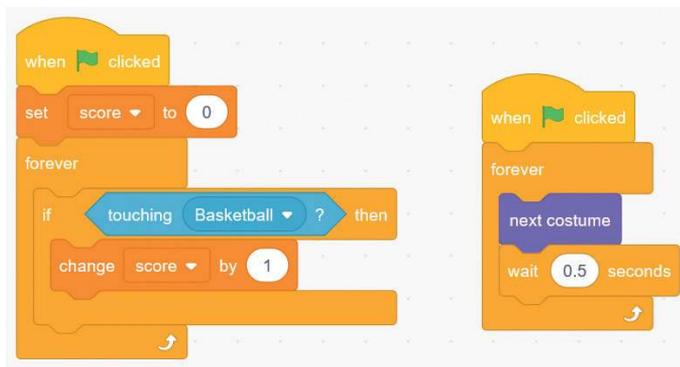
Jawaban Aktivitas AP-K8-01: Bermain dengan Control, Input, dan Variable

Hasil dari Aktivitas AP-K8-01 ialah program lengkap sebagai berikut.

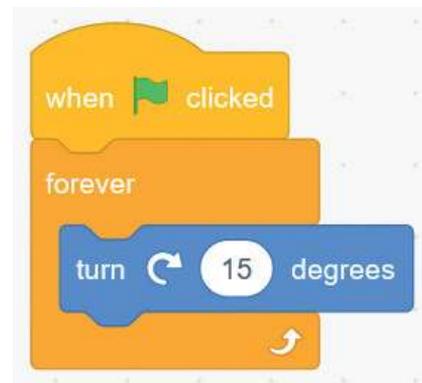
- Kode pada *sprite Cat* si Meong



- Kode pada *sprite Person* si Tika



- Kode pada *sprite Basketball* si Bola



Jika guru membutuhkan penjelasan langkah yang harus dilakukan, berikut ini salah satu urutan langkahnya. Karena ada 3 *sprites*, beberapa langkah yang dapat ditukar urutannya, misalnya menyelesaikan kode *sprite Cat*, *Person*, dan *Basketball*. Ada beberapa pertanyaan sebagai asesmen bagi peserta didik untuk pemahaman lebih lanjut.

Langkah	Tampilan
Persiapan Kasus	
<p>1. Membuat sebuah proyek baru dengan: backdrop <i>Basketball</i>, <i>Sprite Cat</i>, <i>Sprite Person</i>, dan <i>Sprite Basketball</i> serta backdrop <i>Basketball 1</i>.</p> <p>Posisikan <i>Sprite Cat</i> dan <i>Sprite Person</i> pada sisi yang berlawanan dan <i>Sprite Basketball</i> berada di tengahnya.</p>	
<p>2. Buat agar <i>sprite Cat</i> si Meong dan <i>sprite Person</i> si Tika memiliki efek berganti costumes secara terus-menerus.</p> <p>Costumes di sini dapat digambarkan sebagai tampilan yang berbeda untuk 1 <i>sprite</i> tertentu.</p> <p>Manfaatkan kode control <i>forever</i>.</p>	<p>Jawab:</p>  <p>Kode program ini harus ada di <i>sprite Cat</i> si Meong dan <i>sprite Person</i> si Tika.</p>
<p>3. Buatlah juga agar <i>sprite Basketball</i> si Bola berputar secara terus-menerus.</p> <p>Manfaatkan kode control <i>forever</i>.</p>	<p>Jawab:</p> 

Langkah	Tampilan
---------	----------

Bermain dengan Input	
----------------------	--

4. Program meminta input dari pengguna melalui blok ask pada bagian kode Sensing. Pada contoh ini, permintaan input ditiptkan kodenya pada *sprite* Cat karena si Meong yang akan menyapa pengguna dan sapaan muncul setelah pengguna menekan tanda spasi pada keyboard. Input dari pengguna ini juga akan ditampilkan dalam Variabel answer.



5. Kalau pengguna sudah memasukkan input, Variabel answer akan berisi sesuai hasil input.



6. Bagaimana kode program agar si Meong menampilkan kembali hasil input kalian dalam sapaan?

Jawab:



Bermain dengan Variable lain	
------------------------------	--

7. Menyimpan sebuah informasi nilai dalam variable. Untuk menambah variable, memilih bagian kode Variable lalu klik menu Make a Variable, lalu akan muncul dialog box untuk membuat nama variable.



Langkah	Tampilan
<p>Mengatur Variable untuk dikenali di semua <i>Sprite</i> (global variable) atau hanya dikenali di <i>Sprite</i> yang aktif saat Variable dibuat (local variable).</p>	
<p>Contoh: membuat Variable score Apabila di bagian kode Variable, nama Variable di-checklist, Variable tersebut akan muncul di bagian tampilan grid view. Kegunaan dan implementasi dari Variable score ini sama dengan Variable dari hasil Input.</p>	
<p>8. Tambahkan kode berikut pada <i>sprite</i> si Tika. Kode ini akan membuat variable score akan terus bertambah apabila si Bola mengenai si Tika. Apabila di bagian kode variable, nama variable diceklis, variable tersebut akan muncul di bagian tampilan <i>grid view</i>.</p>	

Jawaban Asesmen

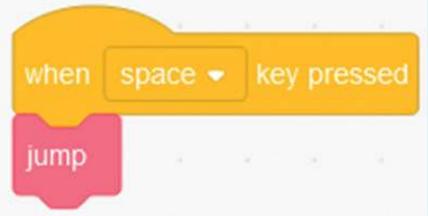
Setelah peserta didik berhasil membuat program yang berjalan dengan baik sesuai skenario, peserta didik diajak untuk menjawab pertanyaan berikut ini untuk memastikan pemahaman peserta didik terhadap pemilihan blok kode.

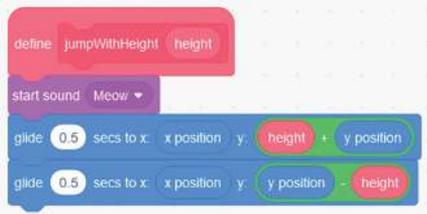
1. Blok kode [wait .. seconds] akan membuat kode dijalankan dengan ada jeda waktu tertentu. Pada kasus ini, jika menggunakan blok [wait .. seconds], akan menyebabkan proses pergantian dari custome 1 ke custome berikutnya seperti lebih pelan. Jika tidak menggunakan blok [wait .. seconds], akan membuat proses pergantian custome menjadi cepat sekali.

2. Makin besar nilai pada blok [wait . . seconds] akan membuat jeda waktu akan makin besar. Dengan demikian, pada kasus ini, proses pergantian custome menjadi makin lambat.
3. Isi nilai score akan terus bertambah. Guru dapat menjelaskan pentingnya inisialisasi nilai sebuah *variable*, dimana hal ini dapat berpengaruh pada hasil akhir program.
4. Perubahan nilai score akan cepat sekali, karena penambahannya per 10 bukan lagi per 1.

Jawaban Aktivitas AP-K8-02: Bermain dengan Custom block

Peserta didik diminta untuk memuat sebuah proyek Scratch baru dan pada kode program si Meong, peserta didik perlu membuat 2 buah Custom Block, yang satu tanpa parameter input dan yang satunya lagi menggunakan parameter input. Kedua Custom Block ini fungsinya mirip, yaitu melakukan pergerakan supaya *sprite* dapat “melompat”. Apa itu melompat? Naik, lalu turun kembali ke posisi semula.

Langkah	Tampilan
Custom Block tanpa Parameter	
<ol style="list-style-type: none"> 1. Buat sebuah blok baru bernama [jump]. Pada blok [jump], tambahkan instruksi pada blok kode agar <i>sprite Cat</i>: <ul style="list-style-type: none"> - Mengeluarkan suara “Meow” - Posisi y akan bertambah 50 - Posisi y kembali lagi ke titik awal Amati hasilnya: kalian akan mendapatkan animasi <i>sprite Cat</i> melakukan lompatan. 	 <p>The image shows the Scratch 'define' block for a custom block named 'jump'. It contains three blocks: 'start sound' set to 'Meow', a 'glide' block for 0.5 seconds to x position and y position + 50, and another 'glide' block for 0.5 seconds to x position and y position - 50.</p>
<ol style="list-style-type: none"> 2. Panggil blok [jump] setiap kali tanda “spasi” pada keyboard ditekan. Note: gunakan blok event. 	 <p>The image shows a Scratch script starting with a 'when space key pressed' event block, followed by a call to the custom block 'jump'.</p>

Langkah	Tampilan
Custom Block dengan Parameter	
<p>3. Buat Custom Block baru, dengan nama [jumpWithHeight] dan beri nama height untuk parameternya.</p> <p>Lakukan:</p> <ul style="list-style-type: none"> - Duplikasi kode dari blok [jump] sebelumnya. - Ganti angka 50 menjadi variable height yang dapat ditarik dari bagian define blok [jumpWithHeight]. 	
<p>4. Jika <i>event</i> ketika tanda panah atas ↑ pada <i>keyboard</i> ditekan, program akan memanggil blok [jumpWithHeight] dengan nilai parameter 100.</p>	

Jawaban Asesment

Setelah peserta didik berhasil membuat program yang berjalan dengan baik sesuai skenario, peserta didik diajak untuk menjawab pertanyaan di Buku Siswa untuk memastikan pemahaman peserta didik terhadap pemilihan blok kode.

1. Efek melompat tidak akan terlalu tinggi, karena berubah dari 100 menjadi 70
2. *Sprite Cat* tidak akan bergerak karena variabel height-nya dianggap 0 sehingga posisi sumbu y akan tetap

Kegiatan Alternatif

Jika sarana komputer/*smartphone* tidak dimiliki, diskusi dapat dilaksanakan dengan mencetak bahan diskusi terlebih dahulu. Fungsi yang ada dapat dicetak sebagai kartu-kartu untuk memudahkan guru dalam menjelaskannya. Guru juga dapat mencari inspirasi melalui Scratch *Unplugged*.

2. Pertemuan 2: Pengantar Blockly Games dan Eksplorasi Puzzle Maze (2 jp)

Tujuan Pembelajaran

Peserta didik mampu memprogram dalam bahasa pemrograman visual kedua yang mirip dengan Scratch, yaitu Blockly, dalam sebuah lingkungan pemrograman blok/visual yang dikemas dalam bentuk permainan Puzzle dan Maze.

Apersepsi

Guru mengajukan beberapa pertanyaan kepada peserta didik: Pernahkah kalian belajar bahasa lain selain bahasa yang digunakan sehari-hari? Di dunia ini ada banyak sekali bahasa dan setiap bahasa memiliki kosakata serta struktur yang berbeda-beda. Bahasa pemrograman juga memiliki banyak jenis. Sebelumnya, kalian belajar bahasa pemrograman Scratch. Pada pertemuan ini, dan berikutnya kalian akan mempelajari bahasa pemrograman Blockly. Seperti belajar bahasa, setelah kalian bisa bahasa Indonesia, belajar bahasa lain tidak sesulit ketika kalian belajar bahasa pertama kali saat bayi 'kan. Dengan menguasai bahasa pemrograman Scratch, kalian akan lebih mudah memahami bahasa pemrograman Blockly.

Pemanasan

Guru mengajak peserta didik untuk membuka link *Blockly Games* dan memperkenalkan lingkungan *Blockly Games* secara umum. Guru bisa menjelaskan secara singkat jenis-jenis permainan yang tersedia. Tabel berikut ini berisi penjelasan mengenai permainan-permainan yang ada dan apa yang dapat dipelajari dari masing-masing permainan. Disarankan, guru sudah mencoba seluruh permainan agar mengerti dan dapat mengembangkan apa yang dapat dipelajari dari masing-masing permainan.

No.	Judul	Deskripsi Ringan	Pelajaran
1	Puzzle	Pemain harus menyusun blok-blok yang berisi keterangan dan foto hewan ke blok hewan yang sesuai.	Peserta didik belajar mengenal blok dan cara memasang blok pada Blockly.
2	Maze	Pemain diminta untuk menggerakkan <i>sprite</i> agar berjalan sampai tujuan.	Peserta didik belajar percabangan dan perulangan melalui permainan ini. Peserta didik juga belajar mengoptimisasi program yang dibuat melalui <i>games</i> ini.

No.	Judul	Deskripsi Ringan	Pelajaran
3	Bird	Pemain diminta untuk menggerakkan bird agar mengambil cacing dan sampai ke sarangnya.	Melalui <i>games</i> ini, peserta didik belajar percabangan. Pada <i>games</i> ini, peserta didik juga belajar mengenai sudut dan pertidaksamaan. Permainan ini dapat direlasikan dengan pelajaran Matematika.
4	Turtle	Pemain diminta untuk menggerakkan turtle untuk menggambar sesuai pola yang disediakan.	Permainan ini cukup mirip dengan permainan Bird. Bedanya, pada permainan ini, peserta didik harus menggunakan perulangan untuk jalannya turtle sehingga peserta didik akan belajar perulangan. Selain perulangan, permainan ini juga dapat direlasikan dengan pelajaran Matematika karena memerlukan kemampuan mengerti sudut dan ukuran.
5	Movie	Pemain diminta untuk membuat <i>sprite</i> sederhana dan menggerakkannya sesuai instruksi.	Pada permainan ini, peserta didik belajar perulangan dengan cara yang berbeda dari sebelumnya. Pada permainan ini, peserta didik juga belajar mengenai bangun datar. Materi ini dapat direlasikan dengan pelajaran Matematika.
6	Music	Pemain diminta untuk membuat musik sederhana berdasarkan not balok yang diberikan.	Pada permainan ini, peserta didik belajar konsep modularisasi program. Selain itu, permainan ini juga dapat direlasikan dengan Seni Musik.

No.	Judul	Deskripsi Ringan	Pelajaran
7	Pond Tutor	Pemain diminta untuk menggerakkan <i>sprite</i> agar menembaiki <i>sprite</i> target hingga darah dari <i>sprite</i> target habis.	Pada permainan ini, peserta didik belajar mengenai percabangan serta perulangan. Pada permainan ini juga, peserta didik untuk mau mencoba berbagai cara untuk memenangkan pertandingan. Peserta didik juga belajar mengenai sudut dan jarak sehingga dapat direlasikan dengan mata pelajaran Matematika.
8	Pond	Permainan ini merupakan lanjutan dari permainan Pond Tutor. Pada permainan ini, <i>sprite</i> pemain memiliki 3 lawan.	Beberapa level pada permainan ini cukup sulit sehingga dirasa kurang cocok untuk kelas VIII. Jika ingin diberikan sebagai latihan, mungkin hanya pada permainan Pond Tutor level 1, 3, 5, dan 7 saja. Guru bisa menyesuaikan dengan waktu dan kemampuan peserta didik.

Isi tabel di atas tidak harus semuanya diberi tahu kepada peserta didik. Guru dapat memilah-milah yang akan dijelaskan sesuai dengan kondisi peserta didik dan waktu yang dimiliki.

Setelah mengenalkan lingkungan Blockly *Games* secara umum, guru mengajak peserta didik untuk memainkan permainan Puzzle bersama-sama sebagai contoh. Pada permainan Puzzle, guru memperkenalkan blok-blok yang tersedia dan bagaimana memasang blok-blok tersebut untuk menyelesaikan permainan. Ajak peserta didik untuk melihat output apa yang diberikan oleh program yang dibuat. Output dari Puzzle adalah text pada dialogue box. Selain text, program juga mengeluarkan output berupa display gambar dan suara. Ingatkan mengenai output yang pernah mereka pelajari di kelas VII.

Kemudian, guru bisa mengajak peserta didik untuk mencoba sendiri permainan Maze sesuai panduan. Guru perlu mengingatkan bahwa tujuan sesi ini adalah belajar pemrograman, jangan hanya bermain!

Kebutuhan Sarana dan Prasarana

- Komputer/*smartphone* yang ter-install sistem operasi dan browser
- Koneksi internet

Kegiatan Inti

Guru dapat memperlihatkan tampilan level 1 dari permainan Maze seperti yang dipandu di Buku Peserta didik. Guru mengarahkan peserta didik untuk melaksanakan Aktivitas AP-K8-03: Eksplorasi Maze. Waktu pengerjaan sekitar 80 menit (2 JP). Guru mengingatkan peserta didik agar tidak hanya sekedar menyelesaikan setiap level, tetapi berusaha memahami kegunaan setiap blok yang ada serta bagaimana jalannya program yang dibuat. Untuk itu, peserta didik perlu diingatkan untuk mencatat hasil eksplorasi mereka pada jurnal.

Saran untuk Guru

Peserta didik tidak diharuskan untuk menyelesaikan semua level, dan tidak harusurut. Guru perlu mempertimbangkan kemampuan peserta didik, dan menentukan level-level yang wajib diselesaikan. Jika memungkinkan, mintalah peserta didik mengerjakan semua level (kecuali level 10). Jika dirasa waktu di dalam kelas tidak cukup, aktivitas dapat dikerjakan sebagai tugas setidaknya sampai level 9. Peserta didik yang berhasil mengerjakan level 10 dapat diberikan *reward* berupa nilai tambahan atau lainnya.

Jawaban Aktivitas AP-K8-03: Eksplorasi Maze

Sebelum mencoba menyelesaikan level-level pada permainan Maze, peserta didik diminta untuk mengenal lingkungan serta fitur yang tersedia pada permainan ini. Untuk itu, diberikan tampilan permainan Maze dan sebuah tabel yang harus diisi. Pada tabel tersebut, peserta didik mengisikan kegunaan dari bagian-bagian yang ditunjukkan oleh setiap nomor pada Gambar (7.1).



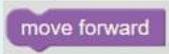
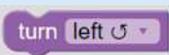
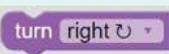
Gambar 7.1 Tampilan Awal Maze

Jawaban dari tabel kegunaan pada tampilan awal Maze.

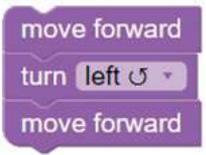
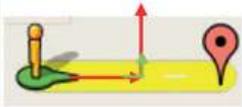
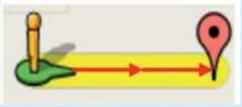
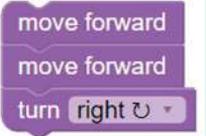
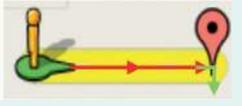
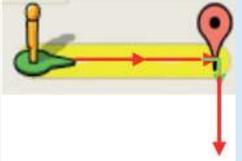
No.	Pelajaran
1	Jawab: Link untuk berpindah ke halaman home dari Blockly <i>Games</i> .
2	Jawab: Link untuk berpindah level. Setiap lingkaran mewakili 1 level dari level 1 (paling kiri) hingga level 10 (paling kanan).
3	Hasil implementasi dari blok-blok kode di bagian 5. Pada bagian ini, terdapat <i>Sprite</i> yang bergerak sesuai dengan kode di bagian 5.
4	Blok-blok kode yang dapat digunakan. Blok kode pada bagian 4 ini dapat di-drag & drop ke bagian 5 untuk menggunakannya.
5	Daftar blok kode yang diimplementasikan pada program.
6	Tombol untuk menjalankan blok kode pada bagian 5. Hasil dari menjalankan kode, diberikan menggunakan gambar di bagian 3. Cobalah untuk menekan tombol ini. Apa perubahan yang terjadi pada bagian 3? Jawab: <i>Sprite</i> yang berada di bagian 3 berjalan sesuai dengan blok yang dipasang di bagian 5.
7	Untuk menghapus blok kode yang sudah terpasang di bagian 5. Cobalah untuk menyeret salah satu blok kode yang sudah terpasang di bagian 5 ke tong sampah ini. Apakah yang terjadi setelah kalian menyeret blok tersebut ke tong sampah ini? Jawab: Blok yang diseret ke tong sampah akan terhapus.
8	Jawab: Daftar <i>Sprite</i> yang dapat digunakan. <i>Sprite</i> yang dipilih akan ditampilkan pada bagian 3. Perubahan <i>Sprite</i> juga akan mengubah <i>background</i> pada bagian 3.

🌱 Level 1

Tabel blok kode yang dapat digunakan:

Blok	Fungsi
	Maju ke depan 1 langkah.
	Jawab: <i>Sprite</i> akan menghadap ke arah kiri.
	Jawab: <i>Sprite</i> akan menghadap ke arah kanan.

Solusi untuk level 1 diberikan oleh tabel di bawah ini.

Blok	Jawaban yang tepat? (Y/N)	Jejak Langkah Sprite	Penjelasan Jalannya Blok
	N		Pada blok yang kedua, <i>Sprite</i> menghadap ke kiri (ditunjukkan oleh panah hijau). Pada langkah ketiga, <i>Sprite</i> bergerak maju dan menabrak jalan.
	Y		2 buah blok [<i>move forward</i>] membuat <i>Sprite</i> bergerak ke depan 2 langkah sehingga <i>Sprite</i> tepat berhenti di posisi yang diinginkan.
	Y		Di langkah ketiga, <i>Sprite</i> menghadap ke kanan, tetapi tidak bergerak maju. Karena <i>Sprite</i> sudah ada di titik target, solusi ini adalah solusi yang valid.
	N		Langkah 1 hingga 3 membuat <i>Sprite</i> berada di titik target, tetapi menghadap ke arah kanan. Dengan langkah keempat, <i>Sprite</i> menabrak jalan.

Selain yang dijelaskan di atas, masih ada solusi-solusi lain untuk permainan level 1. Untuk mengetahui apakah solusi lain yang diajukan peserta didik adalah solusi yang tepat, guru dapat mengevaluasi jejak langkah yang digambarkan peserta didik atau mencoba langsung kode blok yang diberikan oleh peserta didik pada permainan Maze pada Blockly Games.

Setelah menyelesaikan sebuah level, akan tampil *dialogue box* yang berisi kode pada JavaScript yang merupakan padanan dari blok-blok kode. Guru perlu menjelaskan mengenai hubungan kode Javascript tersebut dengan blok yang mereka pakai. Namun, guru perlu menekankan, bahwa untuk kelas VIII, teks kode itu belum menjadi bahan pembelajaran sehingga untuk sementara dapat diabaikan.

Contoh tampilan kode JavaScript untuk level 1 dan blok padananya.

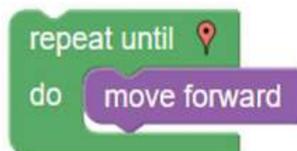


🧩 Level 2

Blok	Jejak Langkah Sprite	Penjelasan Jalannya Blok
		<p>Terdapat 5 buah blok yang menggerakkan <i>Sprite</i>.</p> <ol style="list-style-type: none"> 1. <i>Sprite</i> berjalan maju 2. <i>Sprite</i> menghadap ke kiri 3. <i>Sprite</i> berjalan maju 4. <i>Sprite</i> menghadap ke kanan 5. <i>Sprite</i> berjalan maju

🧩 Level 3

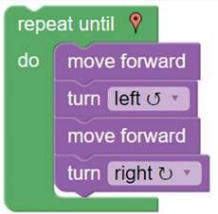
Solusi untuk level 3 sudah diberikan di Buku Siswa. Solusi untuk level 3 adalah sebagai berikut.



Pada level ini, banyaknya blok yang dapat digunakan peserta didik terbatas. Hal ini mengharuskan peserta didik untuk mencari pola dan memanfaatkan perulangan. Kegunaan dari blok [repeat until] adalah menjalankan setiap blok yang dipasang

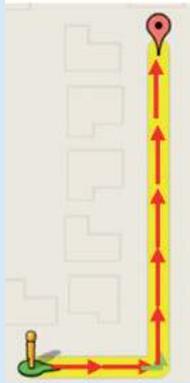
di dalam blok [repeat until] berulang-ulang hingga *Sprite* mencapai target. Guru bisa merelasikan dengan perulangan yang pernah dipelajari di Scratch. Ajaklah peserta didik untuk memahami makna dari perulangan.

🧩 Level 4

Blok	Jejak Langkah Sprite	Penjelasan Jalannya Blok
		<p>Rute pada level ini memiliki pola yang berulang. Setelah <i>Sprite</i> berjalan maju, hadap kiri, berjalan maju, kemudian hadap kanan, <i>Sprite</i> akan menghadapi rute yang sama lagi hingga mencapai titik target. Peserta didik diajak untuk menemukan pola, yang kelak diperlukan untuk mendapatkan abstraksi dari instruksi pengulangan.</p>

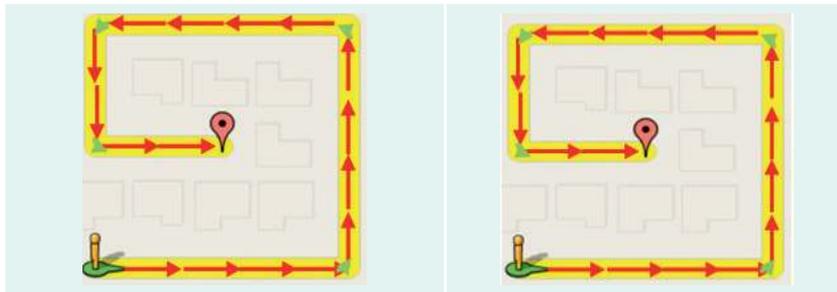
Banyaknya perulangan yang dilakukan adalah sebanyak 4 kali. Penting untuk peserta didik bisa menjawab banyaknya perulangan yang dilakukan karena dengan mengetahuinya, peserta didik belajar untuk paham bagaimana blok [repeat until] bekerja.

🧩 Level 5

Blok	Jejak Langkah Sprite	Penjelasan Jalannya Blok
		<p><i>Sprite</i> berjalan dulu sebanyak 2 langkah ke depan kemudian menghadap ke kiri. Setelah menghadap ke kiri, langkah yang dilakukan <i>Sprite</i> selalu sama, yaitu maju ke depan sehingga dapat menggunakan blok [repeat until] untuk terus bergerak maju hingga mencapai target.</p>

🧩 Level 6

1. Saat blok [if path.. do..] dijalankan, program akan memeriksa apakah kondisi yang ada di dalam if terpenuhi atau tidak. Jika kondisi di dalam if terpenuhi, *Sprite* akan berjalan sesuai blok yang dipasang di dalam [do].
2. Jejak dari eksekusi kedua blok:



Penjelasan bagaimana blok dijalankan.

<p>Dalam setiap perulangan, <i>Sprite</i> selalu:</p> <ol style="list-style-type: none">1. bergerak maju,2. kemudian memeriksa apakah ada jalan di arah kiri. Jika ada jalan di arah kiri, <i>Sprite</i> akan berbelok ke kiri. <p><i>Sprite</i> terus mengulang kedua hal tersebut sampai mencapai titik target.</p>	<p>Dalam setiap perulangan, <i>Sprite</i> selalu:</p> <ol style="list-style-type: none">1. memeriksa apakah ada jalan di arah kiri. Jika ada jalan di arah kiri, <i>Sprite</i> akan berbelok ke kiri;2. setelah itu, <i>Sprite</i> akan bergerak maju 1 langkah. <p><i>Sprite</i> terus mengulang kedua hal tersebut sampai mencapai titik target.</p>
--	---

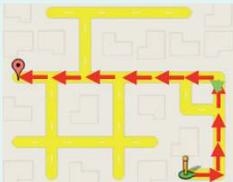
Jawaban Tantangan!

Setelah menggambar jejak *Sprite*, ajak peserta didik untuk melihat bahwa pada setiap jalan lurus, banyak langkah yang dilakukan *Sprite* selalu merupakan bilangan genap (4 langkah atau 2 langkah). Oleh karena itu, sebelum memeriksa apakah harus berbelok ke kiri atau tidak, *Sprite* dapat berjalan sebanyak 2 langkah. Dengan ini, pengecekan **if** jadi berkurang. *Sprite* baru memeriksa kondisi di dalam if, setelah 2 langkah.

Solusi permainan level 6:



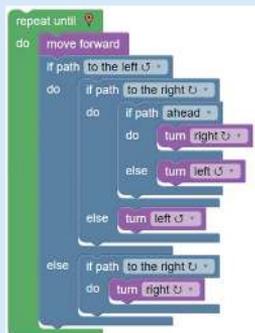
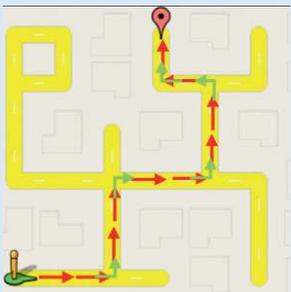
🧩 Level 9

Blok	Jejak Langkah <i>Sprite</i>	Penjelasan Jalannya Blok
 <pre> repeat until [] do if path ahead do move forward else turn left </pre>		<p><i>Sprite</i> akan terus berjalan lurus jika masih ada jalur maju ke depan. Kalau tidak ada jalur maju ke depan, <i>Sprite</i> akan berbelok ke kiri.</p>

Kondisi di dalam if tidak dapat diubah menjadi [to the left] sekalipun blok yang di dalam *do* dan *else* ditukar. Jika kondisi yang diperiksa adalah [to the left], *Sprite* akan berbelok ke kiri di belokan kedua dan kemudian menabrak jalan. Mintalah peserta didik untuk mencoba dan perhatikan apa yang terjadi.

Peserta didik dapat membuat berbagai macam alternatif solusi untuk level 9. Untuk mengevaluasi kebenaran jawaban peserta didik, guru bisa mencoba dengan menjalankan blok-blok tersebut di Blockly *Games* atau mengevaluasi gambar langkah yang dibuat peserta didik.

🧩 Level 10

Blok	Jejak Langkah <i>Sprite</i>	Penjelasan Jalannya Blok
 <pre> repeat until [] do move forward if path to the left do if path to the right do if path ahead do turn right else turn left else turn left else if path to the right do turn right </pre>		<p>Pada level ini, terdapat 4 jenis percabangan yang perlu diatasi.</p> <ol style="list-style-type: none"> 1. Saat ada jalan ke depan dan ke kiri, <i>Sprite</i> selalu memilih untuk berbelok ke kiri. 2. Saat ada jalan ke depan, ke kiri, dan ke kanan, <i>Sprite</i> harus memilih jalan ke kanan. 3. Saat ada jalan ke kiri dan ke kanan, <i>Sprite</i> harus berjalan ke kiri. 4. Saat hanya ada jalan ke kanan, <i>Sprite</i> harus berjalan ke kanan.

Dalam menyelesaikan level ini, peserta didik hanya memiliki blok yang terbatas sehingga peserta didik perlu melakukan optimasi blok yang akan digunakan. Oleh karena itu, peserta didik perlu melihat langkah-langkah apa yang dapat diulang dan kondisi apa yang harus terpenuhi dalam memilih jalur.

Ayo, Pikirkan!

1. Blok [if.. else..] yang tersedia di Blockly *Games* dan Scratch memiliki fungsi dan cara penggunaan yang mirip. Perbedaannya adalah, pada Blockly *Games* kondisi di dalam if terbatas pada menu di *drop down* yang sudah disediakan.
2. Blok [repeat until] pada Blockly *Games* dan Scratch juga memiliki fungsi dan cara yang penggunaan yang mirip. Perbedaannya adalah, pada Blockly *Games* kondisi yang menghentikan perulangannya sudah ditentukan, sedangkan pada Scratch, pengguna dapat menentukan kondisi yang menyebabkan perulangan berhenti.

3. Pertemuan 3: Eksplorasi Blockly Games Music (2 jp)

Tujuan Pembelajaran:

Peserta didik mampu memprogram dalam bahasa pemrograman visual kedua yang mirip dengan Scratch, yaitu Blockly, dalam sebuah lingkungan pemrograman blok/visual yang dikemas dalam bentuk permainan Music untuk kasus kali ini.

Apersepsi

Guru memberikan gambaran kepada peserta didik seperti berikut: Kalian pasti pernah menyanyikan lagu, kan? Pada sebuah lagu biasanya pasti ada bagian yang dinyanyikan berulang kali (misal: bait 1, bait 2, *refrein*, *interlude*). Irama yang sama juga terkadang ditemukan pada beberapa lirik yang berbeda. Coba ambil teks yang isinya lirik lengkap sebuah lagu dan pelajari pola pengulangan pada lagu tersebut! Seringkali untuk menyingkat teks, *refrein* hanya disebut: *refrein* dan orang akan mengerti.

Sekarang, bayangkan teks lagu itu adalah sebuah komposisi blok program. Teks lirik sebuah lagu akan dinyanyikan, sedangkan teks program akan dieksekusi. Pada bagian-bagian blok program yang memiliki fungsi yang sama, dapat diberi nama sehingga ketika bagian tersebut dibutuhkan, penulis program bisa langsung memanggil dengan menggunakan nama yang sudah diberikan.

Pemanasan

Guru dapat memperlihatkan tampilan level 1 dari permainan Music kemudian menjelaskan blok yang ada di permainan level 1. Guru juga bisa menampilkan level 2 dari permainan Music untuk menjelaskan menu Functions dan cara menggunakannya. Guru bisa memberi contoh sederhana menggunakan blok yang tersedia di menu Functions.

Kebutuhan Sarana dan Prasarana

- a. Komputer/*Smartphone* yang terinstal sistem operasi dan browser
- b. Koneksi internet

Kegiatan Inti

Guru mengarahkan peserta didik untuk melaksanakan Aktivitas AP-K8-04: Eksplorasi Music. Waktu pengerjaan sekitar 80 menit (2 JP). Guru mengingatkan peserta didik agar tidak hanya sekedar menyelesaikan setiap level, tetapi berusaha memahami kegunaan setiap blok yang ada serta bagaimana jalannya program yang dibuat.

Tema pada permainan ini dapat dikaitkan dengan pelajaran Musik.

Saran

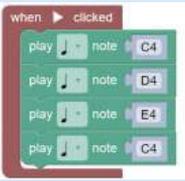
Pada Buku Siswa, diberikan panduan untuk mengerjakan hingga level 5. Level 5 dirasa level minimum yang dikerjakan peserta didik. Jika ada waktu dan kesempatan, guru dipersilakan untuk menambah level yang harus dikerjakan peserta didik. Level-level lain dapat diberikan sebagai tantangan untuk peserta didik.

Jawaban Aktivitas AP-K8-04: Eksplorasi Music

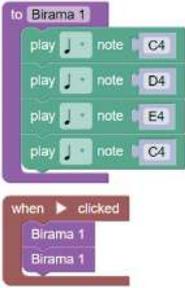
Ada dua konsep yang perlu dijelaskan ke peserta didik terkait eksplorasi musik.

1. Guru perlu menjelaskan bahwa suara yang dikeluarkan oleh komputer dan yang didengar adalah “*output*”. Jadi, *output* pada Blockly tidak hanya berupa teks atau *display* gambar di layar, *output* juga dapat berupa suara. Pada aktivitas kali ini, peserta didik akan membuat program yang *output*-nya adalah suara. Guru bisa mencontohkan menjalankan program yang hanya berisi sebuah blok. Ajak peserta didik untuk memperhatikan bahwa *output* suara yang dimainkan sesuai dengan note yang menjadi input dari blok.
2. Sederet blok dapat dikemas menjadi satu buah blok yang diberi nama, dan nama itu yang akan disebutkan (dipanggil) untuk dieksekusi. Ini akan mendasari pemrograman modular, di mana sebuah program “utama” menjadi lebih ringkas dan mudah dibaca, karena instruksi detail dibungkus menjadi satu blok. Ini abstraksi!

Level 1

Blok	Penjelasan Jalannya Blok
	Ketika tombol Run Program ditekan, program akan memainkan musik C4, D4, E4, dan C4.

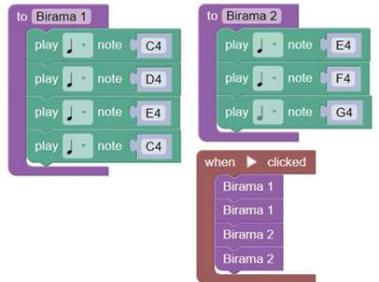
Level 2

Blok	Penjelasan Jalannya Blok
	Kode yang dijalankan adalah kode yang di dalam blok merah. Saat Run Program ditekan, program akan memanggil fungsi Birama 1. Saat memanggil Birama 1, program akan memainkan nada C4, D4, E4, C4. Program akan memanggil fungsi Birama 1 sebanyak 2 kali.

Jawaban pertanyaan:

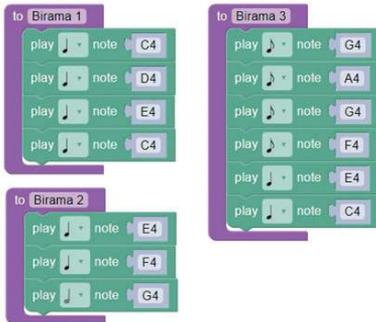
1. Blok Birama 1 dijalankan sebanyak 2 kali karena pada fungsi utama, Birama 1 dipanggil 2 kali.
2. Dengan membuat fungsi, ketika program ingin menjalankan kode yang ada di fungsi tersebut, tidak perlu lagi mengetikkan semua kode, tetapi cukup memanggil nama fungsinya.

Level 3

Blok	Penjelasan Jalannya Blok
	Program akan memanggil fungsi Birama 1 dan menjalankan seluruh blok yang ada di fungsi Birama 1. Kemudian, kembali memanggil fungsi Birama 1 dan setelah itu 2 kali menjalankan fungsi Birama 2.

Level 4

Fungsi-fungsi yang diperlukan untuk menyelesaikan level 4:



Solusi untuk level 4:

Blok	Penjelasan Jalannya Blok
	<p>Program akan memanggil:</p> <ol style="list-style-type: none"> 1. Birama 1 2. Birama 1 3. Birama 2 4. Birama 2 5. Birama 3 6. Birama 3

Level 5

Solusi untuk level 5:

Blok	Penjelasan Jalannya Blok
	<p>Program akan memanggil fungsi:</p> <ol style="list-style-type: none"> 1. Birama 1 2. Birama 1 3. Birama 2 4. Birama 2 5. Birama 3 6. Birama 3 7. Birama 4 8. Birama 4

Setelah peserta didik menyelesaikan level 5, ajaklah peserta didik untuk terus melanjutkan level 6 hingga level 10. Ingatkan mereka untuk menuliskan hasilnya di jurnal mereka. Pada level 10, mereka bisa membuat lagu mereka sendiri. Mintalah mereka mengumpulkan link untuk lagu yang mereka buat.

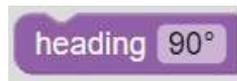
Aktivitas Pengayaan: Permainan Lain pada Blockly Games

Mintalah peserta didik untuk mencoba permainan-permainan lain pada Blockly Games jika masih ada peserta didik waktu di kelas, atau peserta didik tertarik untuk mengerjakan sendiri di luar jam pelajaran. Peserta didik tetap diminta untuk mencatat hasil eksplorasi mereka pada jurnal mereka.

Permainan lainnya diuraikan pada bab ini, memberi peluang guru untuk memilih permainan yang dirasakan lebih cocok untuk disampaikan ke peserta didik.

Bird

Permainan Bird memiliki 10 level. Pada permainan ini, pemain diminta untuk mengatur arah terbang burung agar burung mengambil cacing dan berakhir di sarangnya. Untuk mengatur arah terbang burung, ubah sudut pada blok [heading. . .] seperti pada gambar berikut ini:

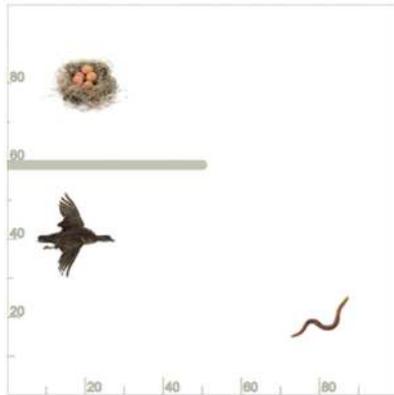


Pada permainan ini, peserta didik belajar konsep percabangan. Pada level 2 hingga 10, peserta didik harus menggunakan blok [if], [else], atau [else if] karena burung tidak terbang dalam sebuah garis lurus saja. Percabangan yang digunakan pada permainan ini lebih rumit daripada permainan Maze. Oleh karena itu, permainan ini dapat menjadi pengantar sebelum peserta didik belajar percabangan pada materi selanjutnya.

Mari melihat level 6 sebagai contoh. Pada level ini, burung harus berganti arah sebanyak 3 kali karena penyebab yang berbeda-beda. Oleh karena itu, dibutuhkan 3 cabang yaitu:

1. Saat belum memakan cacing, burung terbang dalam arah .
2. Setelah bertemu cacing, burung harus berpindah arah. Saat burung terbang pada arah .
3. Setelah , burung terbang dalam arah .

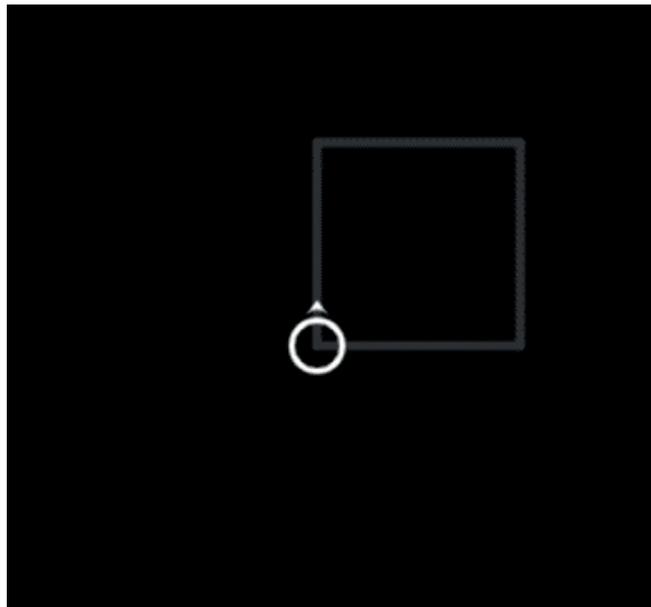
Solusi untuk level 6 adalah sebagai berikut:



```
if does not have worm
do heading 345°
else if y < 80
do heading 90°
else heading 180°
```

Turtle

Permainan Turtle memiliki 10 level. Pada permainan Turtle peserta didik belajar mengenai perulangan. Peserta didik diminta untuk mengarahkan *turtle* hingga menggambar pola yang diinginkan. Mari melihat contoh pada level 1. Pola yang harus digambar oleh *turtle* ditandai oleh garis abu-abu.



Untuk menggambar pola itu, *turtle* dapat melakukan gerakan:

1. maju
2. belok kanan

sebanyak 4 kali. Karena gerakan yang dilakukan sama, peserta didik dapat menggunakan perulangan untuk menyelesaikan level ini. Peserta didik juga dilatih untuk mengeksplorasi ukuran serta sudut yang tepat saat melangkah maju dan berbelok. Solusi untuk level 1 ditunjukkan oleh gambar berikut ini.

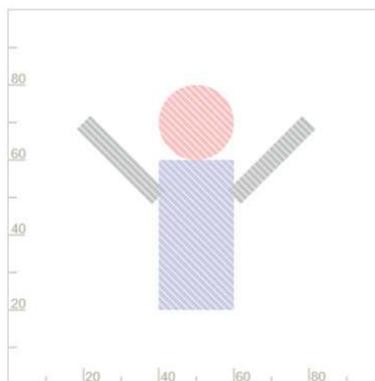


Perulangan yang dibutuhkan akan makin kompleks seiring bertambahnya level. Permainan ini akan menjadi pengantar yang cocok sebelum belajar perulangan pada pertemuan-pertemuan berikutnya. Pada level 10, peserta didik dipersilakan untuk berkreasi membuat gambar dengan menggunakan blok-blok yang diberikan. Tantang peserta didik untuk membuat gambar yang unik dan kirimkan linknya kepada guru.

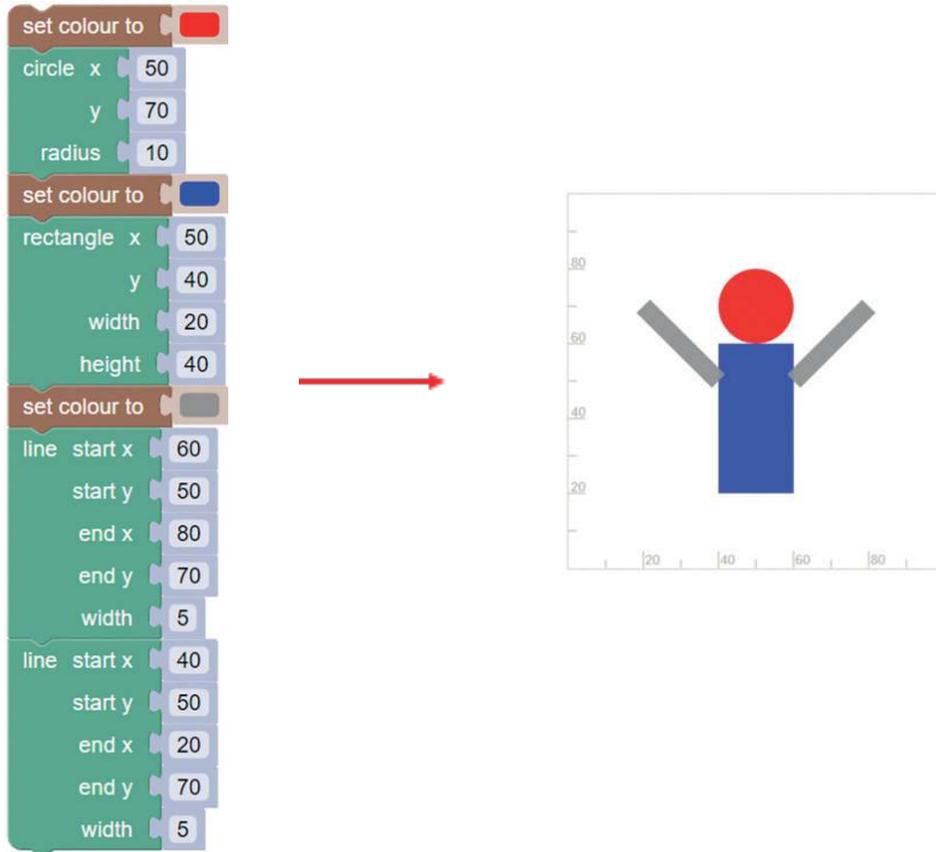
Movie

Pada permainan ini, peserta didik belajar membuat *Sprites* sederhana. Pada level 1, *Sprites* yang dibuat tidak bergerak. Peserta didik diminta untuk menempatkan lingkaran, persegi, serta garis pada posisi yang tepat. Selain posisi, peserta didik juga dapat mengatur ukuran dari setiap bentuk dan juga memberi warna untuk setiap bentuk. Permainan ini berkaitan dengan bentuk/bangun geometris, yang dipelajari oleh peserta didik pada pelajaran Matematika.

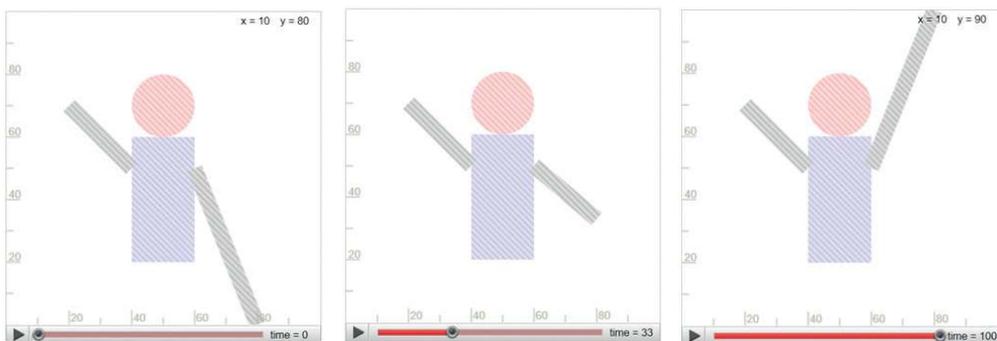
Berikut merupakan contoh soal untuk level 1:



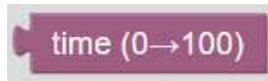
Berikut ini adalah solusi untuk level 1:



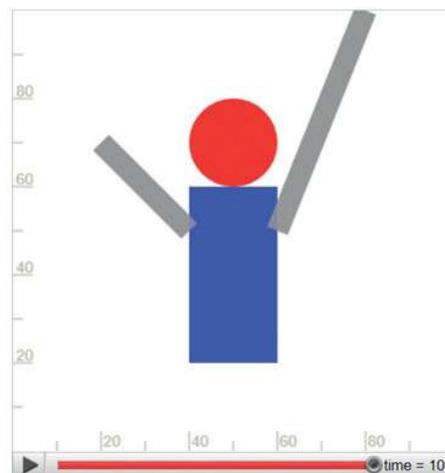
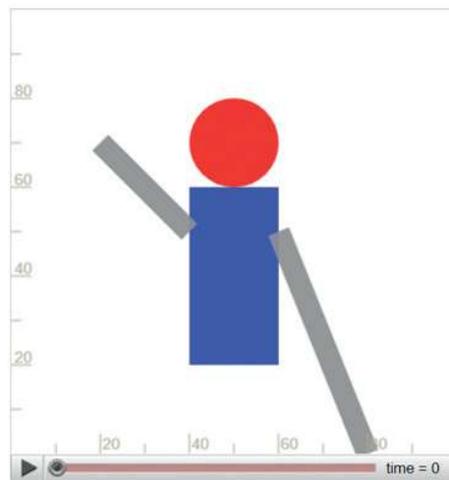
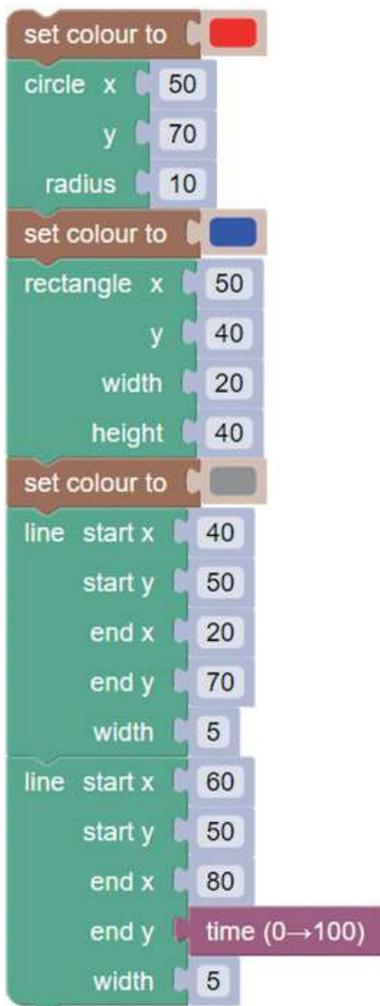
Untuk menentukan posisi x, y, lebar, dan panjang dari sebuah bentuk, peserta didik dapat mengarahkan *cursor* ke benda yang ingin dibuat. Pada level-level berikutnya, ada bagian-bagian yang dapat bergerak. Berikut ini adalah persoalan pada level 2. “Tangan” kanan dari *Sprite* bergerak dari bawah ke atas.



Untuk menggerakkan tangan, ada sebuah blok yang berguna untuk mengubah isi variabel time dari 0 hingga 100. Blok tersebut ditunjukkan pada gambar berikut ini.



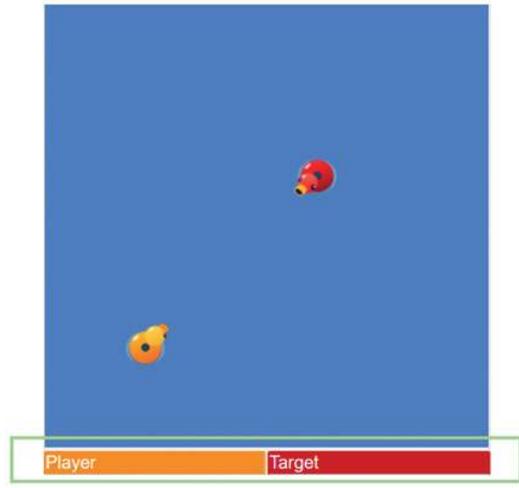
Bagaimana menggunakan blok tersebut untuk menggerakkan tangan *Sprite*? Kalau diperhatikan, yang bergerak hanyalah posisi di sumbu-y dari posisi akhir garis yang menjadi tangan kanan *Sprite*. Posisi sumbu-y dari akhir garis yang menjadi tangan kanan *Sprite* bergerak dari hingga . Oleh karena itu, blok `time` dapat dipasang pada posisi blok `[end y]` seperti pada gambar berikut ini.



Pada level-level berikutnya, pergerakan dari *Sprite* makin bertambah rumit. Kemampuan matematika dan percabangan peserta didik akan diasah dalam permainan ini.

Pond Tutor

Pada permainan Pond Tutor, pemain ditugaskan untuk menembak target hingga darah dari target habis.



Pada gambar di atas, *Sprite* orange ialah pemain, sedangkan *Sprite* merah ialah target. Bar yang ditandai oleh kotak hijau menandakan darah dari target. Mari, mulai dari level 1. Pada level 1, darah target sudah kosong sehingga pemain cukup menembak satu kali mengenai sasaran untuk menyelesaikan level 1.



Untuk menembak, dapat digunakan blok [canon(sudut, jarak)]. Blok canon memiliki 2 parameter, yaitu sudut pemain akan menembak dan jarak peluru yang ditembakkan. Untuk mengetahui jarak yang tepat, pemain perlu mencoba dan memperkirakan

jarak yang tepat. Untuk level 1, sudut dan jarak yang tepat adalah dan 40. Pada level 2, pemain bermain tidak lagi menggunakan blok, tetapi menuliskan perintah yang bersesuaian dengan blok yang digunakan. Perhatikan gambar berikut. Blok yang ada di sebelah kiri bersesuaian dengan perintah yang diberikan di gambar sebelah kanan.



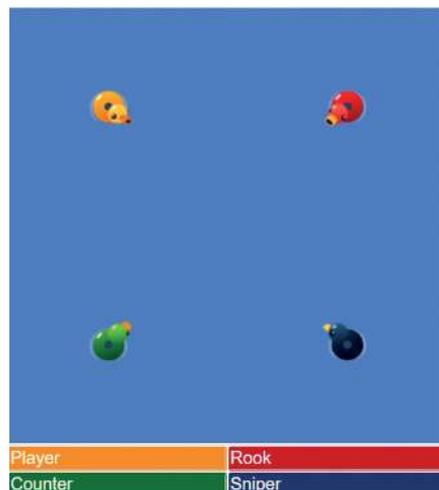
Pada setiap level genap, peserta didik tidak lagi membuat program dengan menggunakan blok, tetapi dengan menggunakan perintah pada JavaScript. Melalui permainan ini, peserta didik mulai belajar bahasa pemrograman lain selain bahasa blok/visual. Permainan ini dirasa kurang cocok untuk diberikan di kelas VIII sehingga bagian ini merupakan *catatan* untuk guru saja.

Seiring bertambah level, makin banyak blok yang dapat digunakan. Pada level 7, *Sprite* pemain harus berenang terlebih dahulu agar dapat menembak target. Pada permainan Pond Tutor, peserta didik berlatih menggunakan blok-blok yang disediakan sebelum bermain permainan Pond.

Pond

Aturan pada permainan Pond sama dengan permainan Pond Tutor. *Sprite* pemain harus bertahan sampai semua *Sprite* target kehabisan darah.

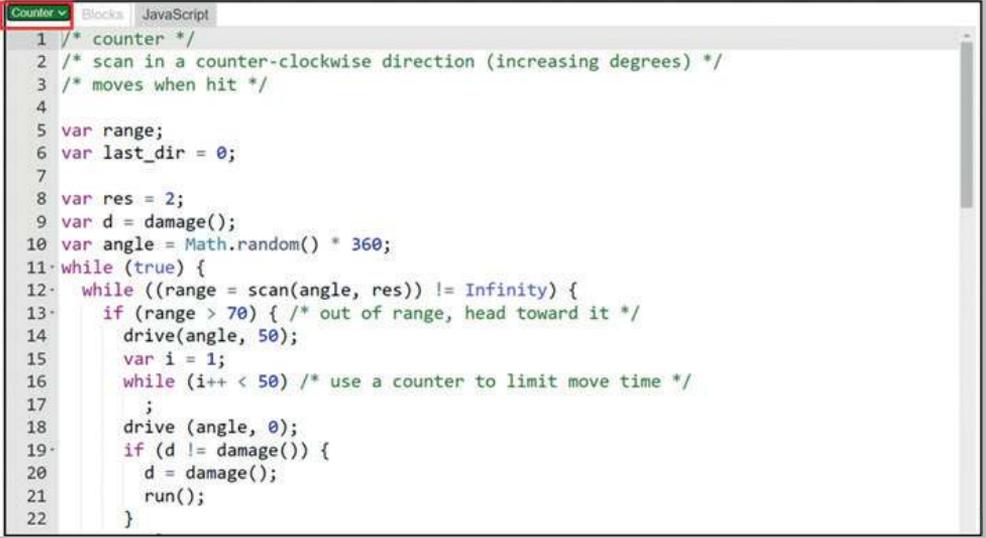
Pada permainan ini, terdapat 3 *Sprite* target, yaitu Rook, Counter, dan Sniper seperti yang ditunjukkan oleh gambar berikut ini.



Rook, Counter, dan Sniper dapat bergerak dan menembak. Peserta didik perlu menyusun strategi bagaimana *Sprite* pemain harus bergerak dan menembak. Untuk mengatur gerak dan tembakan dari *Sprite* pemain, peserta didik dapat menggunakan

blok-blok yang sudah tersedia. Blok yang tersedia pada permainan ini merupakan blok yang digunakan di Pond Tutor dan beberapa blok tambahan. Pada permainan ini, peserta didik perlu mengeksplorasi kegunaan dari setiap blok, kemudian menyusun blok agar *Sprite* pemain menang melawan semua musuhnya.

Peserta didik juga dapat melihat *script* gerakan dari Rook, Counter, dan Sniper dengan memilih pemain yang ingin dilihat pada *dropdown* yang ditunjukkan oleh kotak merah pada gambar berikut ini.



```
1 /* counter */
2 /* scan in a counter-clockwise direction (increasing degrees) */
3 /* moves when hit */
4
5 var range;
6 var last_dir = 0;
7
8 var res = 2;
9 var d = damage();
10 var angle = Math.random() * 360;
11 while (true) {
12   while ((range = scan(angle, res)) != Infinity) {
13     if (range > 70) { /* out of range, head toward it */
14       drive(angle, 50);
15       var i = 1;
16       while (i++ < 50) /* use a counter to limit move time */
17         ;
18       drive (angle, 0);
19       if (d != damage()) {
20         d = damage();
21         run();
22       }
23     }
24   }
25 }
```

Catatan: Permainan ini dirasa kurang cocok untuk kelas VIII sehingga bagian ini juga ditujukan sebagai catatan untuk guru.

4. Pertemuan 4: Eksplorasi *Sprites* dengan Blockly (2 jp)

Pada pertemuan kali ini, peserta didik akan memprogram “*sprite*” Scratch dalam lingkungan Blockly. *Sprite* dalam dunia grafika komputer ialah komponen dalam sebuah program yang merupakan gambar *bitmap* dua dimensi. *Sprite* banyak digunakan untuk menggambarkan objek seperti tokoh karakter atau benda lain. *Sprite* dapat diam atau bergerak dengan memanfaatkan perubahan posisi *sprite* atau gambar lain sehingga seolah-olah terjadi animasi.

Tujuan Pembelajaran:

1. Peserta didik mampu memprogram dalam bahasa pemrograman visual kedua yang mirip dengan Scratch, yaitu Blockly, dalam sebuah lingkungan pemrograman blok/visual yang dikemas dalam bentuk permainan.

2. Peserta didik mampu menyusun kode program Blockly:
 - a. Melakukan *drag and drop* blok pemrograman yang tersedia untuk menyusun sebuah program.
 - b. Menjalankan dan melihat hasil eksekusi program yang dibuat.

Apersepsi

Setelah peserta didik mencoba eksplorasi permainan pada Blockly *Games*, saat ini, peserta didik akan melakukan eksplorasi pada kode program Blockly yang menggunakan *sprite*, mirip seperti Scratch. Peserta didik dikenalkan dengan *tools* yang akan digunakan, yaitu Coding with Chrome. Sesuai namanya, aplikasi ini hanya dapat diunduh menggunakan *browser* Google Chrome. Peserta didik dibantu untuk melakukan proses instalasi dan sesudahnya, peserta didik dapat melakukan eksplorasi mandiri terkait fitur yang tersedia pada *tools* Coding with Chrome, khususnya bagian *Games*.

Pemanasan

Guru dapat menunjukkan secara langsung fitur-fitur yang ada pada *tools* Coding with Chrome. Setelah itu, guru menjelaskan skenario aktivitas yang akan dilakukan.

Kebutuhan Sarana dan Prasarana

- a. Komputer/*Smartphone* yang ter-*install* sistem operasi dan *browser*
- b. Koneksi internet

Kegiatan Inti

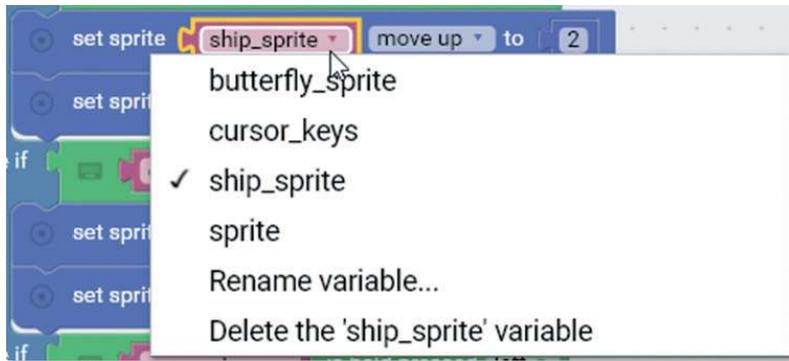
Peserta didik mengerjakan secara mandiri Aktivitas AP-K8-05: Eksplorasi *Games Move a sprite* dan Aktivitas AP-K8-06: *Customize Games Move a sprite* pada Buku Siswa. Setelah peserta didik selesai, guru dapat membahas bersama hasil jawaban dan memberikan umpan balik terkait jawaban.

Jawaban Aktivitas AP-K8-05: Eksplorasi Games Move a sprite

Pada permainan ini, peserta didik diharapkan dapat mengingat kembali pemrograman Scratch, yang selalu dimulai dengan menuliskan kode pada *Sprite*. Guru menjelaskan, bahwa tidak semua pemrograman visual memakai pendekatan seperti itu. Walaupun Blockly mirip dengan Scratch, ternyata ada bedanya, terutama dalam hal penggunaan *Sprite*. *Sprite* bukan objek yang utama dalam Blockly, akan dipahami peserta didik melalui aktivitas ini. Pada Blockly, potongan gambar apa pun yang akan digerakkan di layar, dapat dipakai sebagai padanan “*sprite*” pada Scratch.

Peserta didik perlu diberitahukan bahwa pada Scratch, *drag & drop* blok kode dilakukan untuk setiap *sprite* yang ada. Pada Blockly, blok kode ditempatkan di

tempat yang sama dan pada bloknya dapat memilih kode tersebut akan berdampak pada *sprite* yang mana, sebagai contoh:



Pada blok [set sprite .. moveup to ..], akan muncul pilihan semua *sprite* yang tersedia untuk menentukan *sprite* yang terdampak.

Gambar latar belakang disebut Backdrop pada Scratch, sedangkan pada Blockly, layar belakang tampilan dapat menggunakan gambar maupun warna yang disebut background image atau background color.

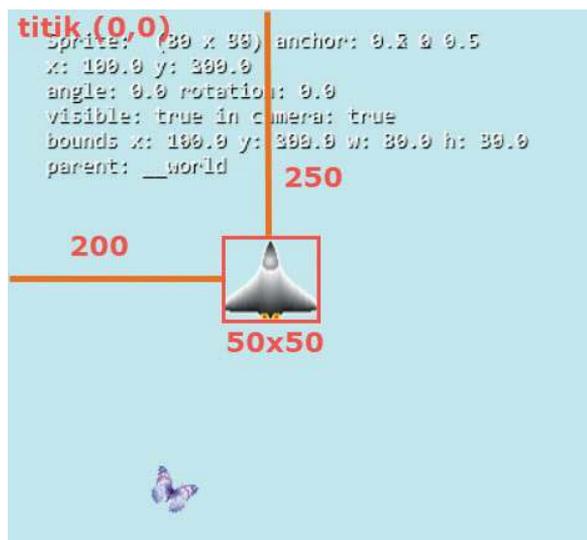
Pertanyaan	Jawaban
1. Apa yang dapat dilakukan oleh program ini?	Dengan menggunakan tanda panah atas, bawah, kanan, kiri pada <i>keyboard</i> , pengguna dapat menggerakkan <i>sprite Ship</i> yang tersedia. Apabila <i>sprite</i> keluar dari salah satu sisi, <i>sprite</i> akan muncul kembali dari sisi seberangnya.
2. Blok kode apa yang digunakan untuk mengatur warna latar pada bagian A?	Blok kode [set background color].
3. Apa yang terjadi apabila pada kode berikut, angka 0.5 diubah menjadi 1?  Bagaimana apabila diubah lagi nilainya menjadi -2?	Jika nilai diubah menjadi 1, posisi <i>sprite Ship</i> akan makin ke kiri atas. Jika <i>sprite Ship</i> digerakkan, tumpuannya juga berubah. Apabila nilai diubah menjadi -2, posisi <i>sprite Ship</i> makin ke kanan bawah. Titik tengah <i>sprite anchor</i> 0.5,
4. Apa efeknya apabila nilai 2 pada blok kode berikut diubah menjadi 7? 	Apabila panah bawah ditekan, <i>sprite</i> akan bergerak lebih jauh ke bawah.

Pertanyaan	Jawaban
5. Apa guna dari blok kode berikut? 	Untuk mengubah arah posisi menghadap dari sebuah <i>sprite</i> . Sudut -90 artinya arah barat.
6. Menurut kalian, apa guna dari blok kode [on preload], [on create], [on input], [on update], dan [on render]?	Karena ini berupa <i>games</i> , perlu ada tahapan-tahapan ini, dimulai dari persiapan sampai final <i>render</i> program.
7. Apabila salah satu blok kode tadi dihapus, apakah program dapat berjalan dengan baik?	Tidak. Bahkan, blok kode tidak dapat saling menempel apabila urutannya berubah.

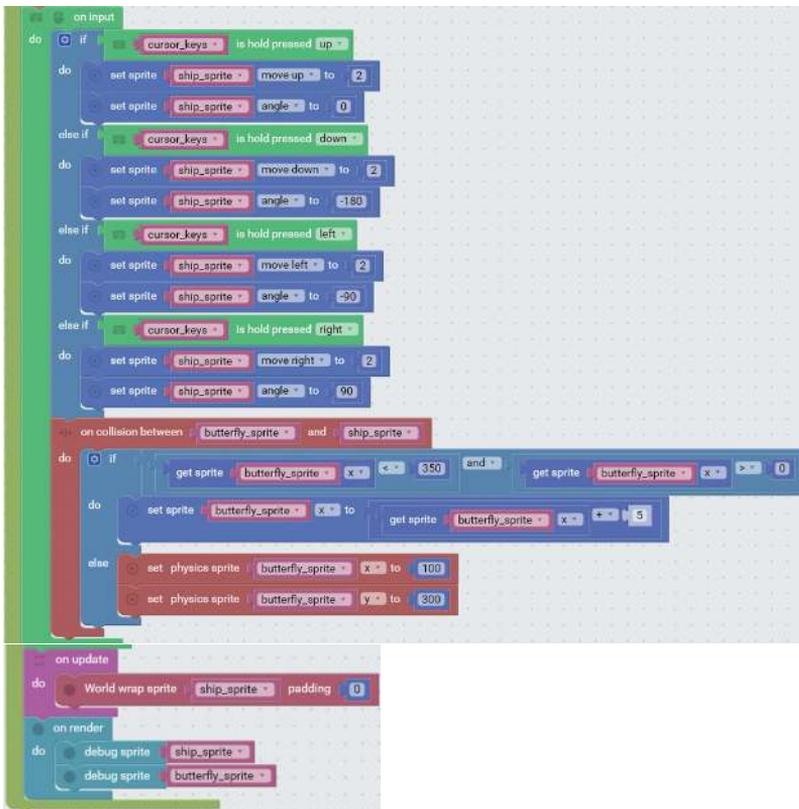
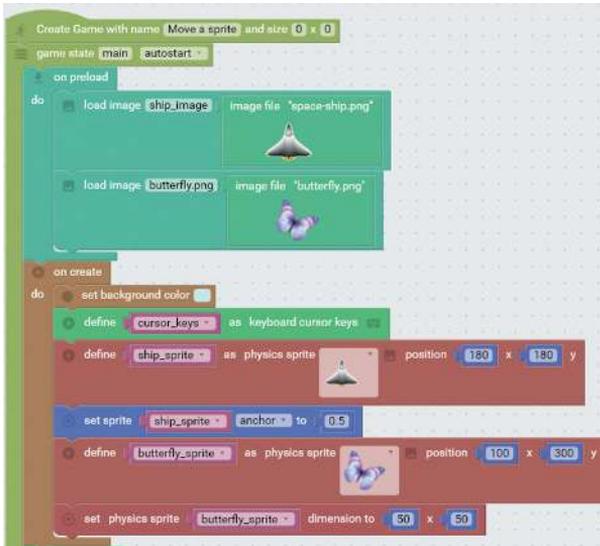
Jawaban Aktivitas AP-K8-06: Customize Games Move a *sprite*

Pada aktivitas ini, peserta didik diajak untuk mencoba eksplorasi lebih lanjut dengan mengostumisasi kode program pada proyek *Move a sprite* yang telah tersedia dengan menambah *sprite* lain dan beberapa aturan lainnya. Tema pembelajaran kali ini ialah Pesawat dan Benda Angkasa Lainnya, peserta didik perlu menambahkan satu *sprite* lain dan mengatur kode lainnya agar sesuai dengan skenario.

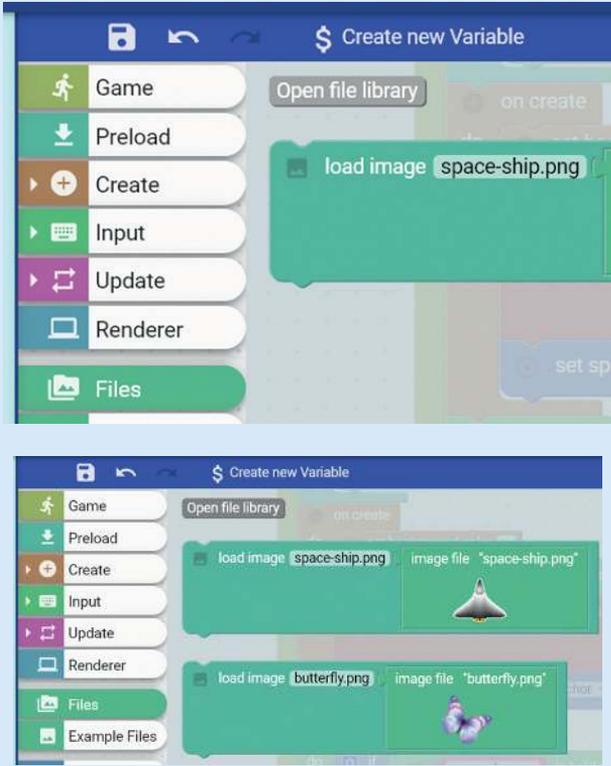
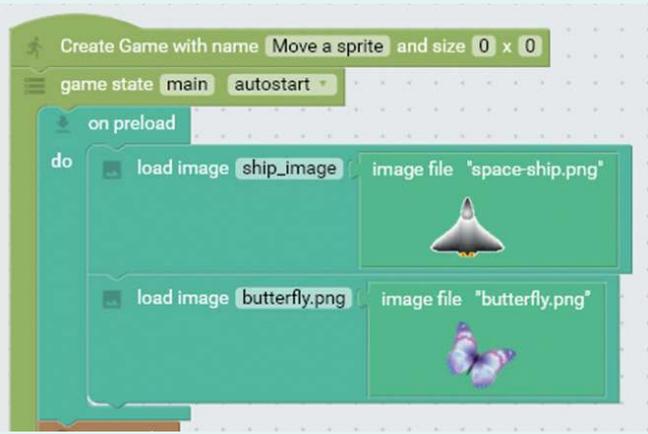
Guru perlu menjelaskan sistem koordinat pada *tools* Coding with Chrome sehingga memudahkan peserta didik dalam mengatur posisi *sprite*. Koordinat (0,0) berada di sudut kiri atas dan ukuran *grid view* pada aplikasi ini adalah 400 x 800 pixel apabila aplikasi dalam kondisi *full screen*. Berikut ilustrasi untuk membantu guru dan peserta didik dapat mengerti *grid view* pada *tools* Coding with Chrome. Posisi *sprite* ditentukan dari titik tengah *sprite* tersebut, seperti contoh berikut.



Keseluruhan jawaban untuk aktivitas ini adalah sebagai berikut.



Jika guru memerlukan tahapan pembentukan solusi tersebut langkah demi langkah seperti yang dituliskan pada Buku Siswa, berikut ini urutan langkahnya.

Langkah	Tampilan
Tambah Sprite Baru – kode on preload	
<p>Unggah dulu gambar yang akan kalian tambahkan.</p> <p>Pilih menu Files, lalu Open file library, unggah gambar baru dan nantinya di tab Images akan muncul gambar yang sudah berhasil diunggah.</p> <p>Lalu, di menu Files, juga akan muncul blok baru.</p>	
<ol style="list-style-type: none"> 1. Dari kode program bagian [on preload], kalian perlu menambahkan gambar baru tadi menggunakan blok [load image]. 	

Langkah	Tampilan
---------	----------

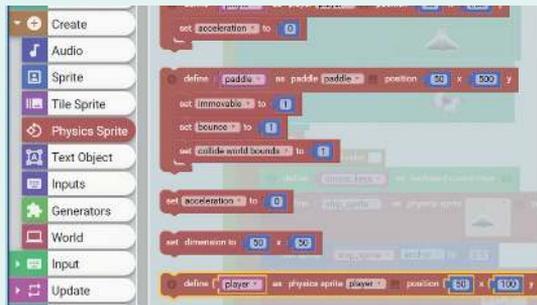
Mempersiapkan Sprite – kode on create	
---------------------------------------	--

2. Setelah *Sprite* baru berhasil ditambahkan, kalian perlu menyiapkan posisi dan ukuran dari *Sprite* di bagian kode *on create*.

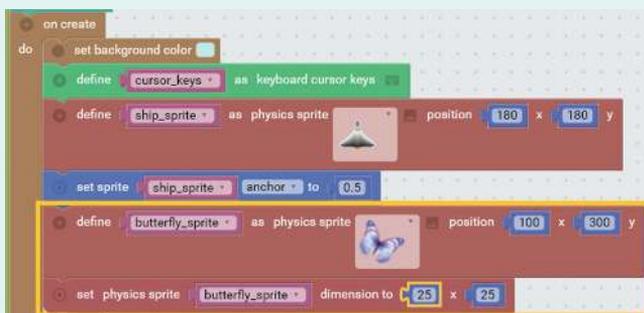
Mengatur ukuran *Sprite*:

Cari blok yang sesuai pada kategori yang sama, yaitu Create – Physics *Sprite*.

Mengatur posisi awal *Sprite*:



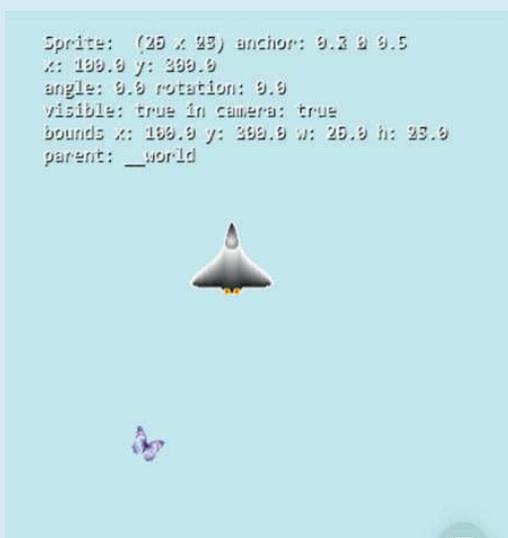
Jawab:



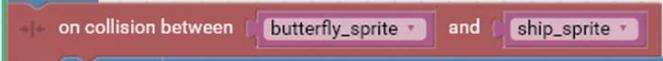
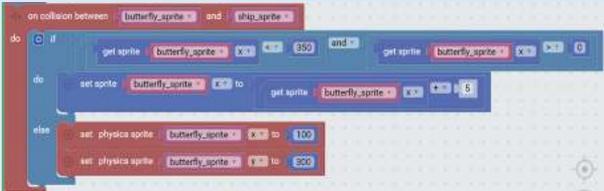
Contoh hasil jadi program yang telah ditambahkan *sprite* X, dengan X gambar *Butterfly*.

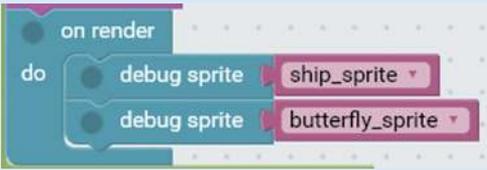
Koordinat *sprite* *Ship* yang digunakan adalah (180, 180).

Koordinat *sprite* *Butterfly* yang digunakan adalah (100, 300) dengan ukuran 25 x 25.



```
Sprite: (25 x 25) anchor: 0.5 0 0.5
x: 100.0 y: 300.0
angle: 0.0 rotation: 0.0
visible: true in camera: true
bounds x: 100.0 y: 300.0 w: 25.0 h: 25.0
parent: __world
```

Langkah	Tampilan
Menjalankan <i>Sprite</i> - kode on input	
<p>3. Kode yang saat ini ada ialah <i>sprite Ship</i> dapat bergerak mengikuti arah panah yang ditekan oleh pengguna, misal panah atas maka <i>sprite</i> akan bergerak ke atas juga.</p> <p>Kode apa yang digunakan untuk mengatur jalannya program apabila <i>sprite Ship</i> dan <i>Butterfly</i> bertabrakan?</p>	<p>Jawab:</p> 
<p>4. Buatlah kode program yang dapat membuat posisi <i>sprite Butterfly</i> akan bergerak secara horizontal ke kanan setiap kali <i>sprite Ship</i> dan <i>sprite Butterfly</i> bertabrakan.</p>	<p>Jawab:</p> 
<p>Buatlah kondisi apabila posisi horizontal <i>sprite Butterfly</i> keluar dari range 0-350, <i>sprite</i> akan kembali ke titik awal <i>sprite</i> dibuat.</p> <p>Gabungkan dengan kode program sebelumnya.</p>	<p>Jawab:</p> 

Langkah	Tampilan
<i>Generating Sprite - kode on render</i>	
Jangan lupa <i>sprite</i> baru harus di- <i>render</i> juga.	<p>Jawab:</p> 

Asesment

Setelah peserta didik berhasil membuat program yang berjalan dengan baik sesuai skenario, peserta didik diajak untuk menjawab pertanyaan di Buku Siswa untuk memastikan pemahaman peserta didik terhadap pemilihan blok kode.

1. Blok yang digunakan untuk mendefinisikan sebuah *sprite* baru memang pasti langsung mengatur posisi *sprite*. Hal ini juga dianggap sebagai inisiasi awal sebuah objek.
2. Apabila gambar yang digunakan berukuran besar, akan membuat hasil jalannya program menjadi kurang tepat. Apabila gambar asli dari *sprite* yang digunakan ukurannya memang kecil, blok pengaturan dimensi ini dihilangkan juga tidak masalah.

5. Pertemuan 5: Pengenalan Pemrograman Prosedural (2 jp)

Guru menjelaskan bahwa dalam pemrograman prosedural, ada kebiasaan, bahwa program pertama yang dibuat untuk mengenal sebuah lingkungan pemrograman, disebut “Hello World”.

Peserta didik akan diajak berkenalan dengan dunia pemrograman dengan menyapa lingkungan pemrograman yang dipakainya melalui “Hello World”.

Tujuan Pembelajaran:

1. Peserta didik mampu membaca dan memahami makna blok penyusun program dalam bahasa Blockly:
 - a. Variabel, input, output
 - b. Ekspresi matematika, ekspresi logika dan perhitungannya
 - c. Percabangan
 - d. Pengulangan

2. Peserta didik mampu menyelesaikan persoalan dengan menyusun program prosedural dengan bahasa Blockly:
 - a. Membuat spesifikasi input, *output*, proses
 - b. Menganalisis dan mengembangkan solusi
 - c. Menyusun kode program yang sesuai :
 - i. Melakukan *drag and drop* blok pemrograman yang tersedia untuk menyusun sebuah program.
 - ii. Menjalankan dan melihat hasil eksekusi program yang dibuat.
 - iii. Membuat program yang menerima input, dan menyimpannya dalam sebuah variabel.

Apersepsi

Saat seseorang berada di tempat baru, tentu biasanya menyapa atau berkenalan dengan lingkungan yang baru tersebut. Di dalam pemrograman, biasanya “menyapa” atau berkenalan dengan lingkungan pemrograman dengan cara menuliskan kode sederhana, yaitu “Hello World”. Mengapa perlu menuliskan kode sederhana ini terlebih dahulu? Karena dengan menulis kode program sederhana ini, pengguna dapat mengetahui apa yang sedang dikerjakan oleh program, melalui *output* yang dihasilkan.

Pada bagian sebelumnya, kalian sudah melihat bahwa *output* dari program adalah sebuah *movement*, *text* dalam *dialogue box* dan *sound*. Pada bagian ini, *output* yang akan dihasilkan adalah berupa teks langsung.

Konsep

Konsep pada Buku Panduan Guru ini sama dengan yang tertulis pada Buku Siswa.

Kebutuhan Sarana dan Prasarana

- a. Komputer/*Smartphone* yang ter-*install* sistem operasi dan *browser*
- b. Koneksi internet

Kegiatan Inti

Kegiatan inti pada pertemuan ini adalah sebagai berikut.

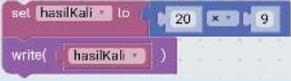
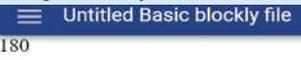
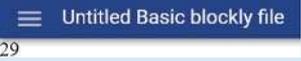
- Guru menjelaskan tiap blok yang ada pada Tabel Contoh Perintah dalam Bahasa Blockly di Buku Siswa.
- Setelah menjelaskan tiap blok tadi, guru akan meminta peserta didik untuk meng-*install* sebuah perangkat lunak untuk membuat program menggunakan Blockly. Perangkat lunak ini bernama **Coding With Chrome**, yang dapat

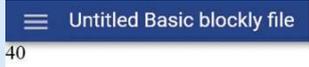
diunduh melalui: <https://chrome.google.com/webstore/detail/coding-with-chrome/becloognjehhioodmnimnehjcbkloed>

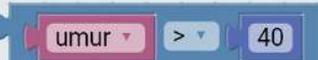
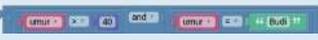
- Peserta didik yang sudah berhasil melakukan instalasi Coding with Chrome dapat diminta untuk mencoba sendiri blok-blok untuk perintah input, *output*, operasi aritmetika, operasi perbandingan, operasi boolean, operasi kondisional, dan operasi perulangan yang ada pada Tabel awal. Peserta didik juga dapat diminta untuk mengganti-ganti isi dari kode program dan mengamati perbedaannya. Guru dapat memandu dengan memberi contoh.
- Kegiatan berikutnya adalah mengerjakan Aktivitas AP-K8-07: *Hello World* dan Aktivitas AP-K8-08: Hello Namaku. Setelah selesai, guru dapat menjelaskan makna dari tiap blok yang digunakan pada kedua aktivitas tersebut.
- Selanjutnya, guru memberikan penjelasan tentang struktur percabangan dalam pemrograman, sesuai dengan penjelasan yang ada pada Buku Siswa.
- Setelah mendengarkan penjelasan dari guru, peserta didik dapat mencoba membuat program dengan perintah operasi boolean dan instruksi kondisional yang ada pada Tabel Contoh Perintah dalam Bahasa Blockly di Buku Siswa. Jawaban untuk Pertanyaan Pada Tabel Contoh Blok pada Blockly

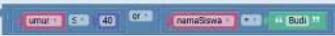
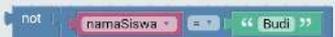
Tabel 1.2 Blok pada Blockly

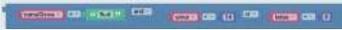
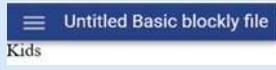
Perintah	Variasi	Contoh Blok Simbol Perintah Blockly	Makna Potongan Kode Program
Input	Input disimpan dalam variabel		Lakukan operasi mengisi variabel namaSiswa dengan "Budi". Nilai namaSiswa: Budi
Output	Output suatu nilai		Tuliskan atau tampilkan nilai 29 ke layar. Tampilan di layar:
	Output nilai variabel		Tuliskan atau tampilkan isi variabel namaSiswa. Tampilan di layar:

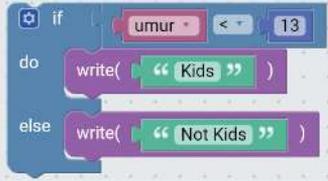
Perintah	Variasi	Contoh Blok Simbol Perintah Blockly	Makna Potongan Kode Program
Operasi aritmetika	Perkalian		<ul style="list-style-type: none"> Lakukan operasi perkalian 20 dengan 9 dan hasil operasi perkaliannya disimpan dalam variabel hasilKali. Tampilkan nilai yang disimpan di hasilKali ke layar. <p>Tampilan di layar:</p> 
	Pembagian		<ul style="list-style-type: none"> Lakukan operasi pembagian 20 dengan 4 dan hasil operasi pembagiannya disimpan dalam variabel hasilBagi. Tampilkan nilai yang disimpan di hasilBagi ke layar. <p>Tampilan di layar:</p> 
	Penjumlahan		<ul style="list-style-type: none"> Lakukan operasi penjumlahan 20 dengan 9 dan hasil operasi penjumlahannya disimpan dalam variabel hasilJumlah. Tampilkan nilai yang disimpan di hasilJumlah ke layar. <p>Tampilan di layar:</p> 

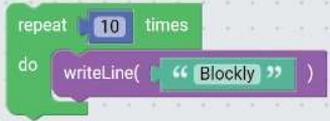
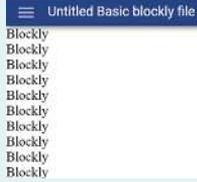
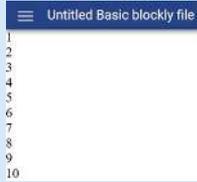
Perintah	Variasi	Contoh Blok Simbol Perintah Blockly	Makna Potongan Kode Program
	Pengurangan		<ul style="list-style-type: none"> Lakukan operasi pengurangan 20 dengan 4 dan hasil operasi penjumlahannya disimpan dalam variabel hasilKurang. Tampilkan nilai yang disimpan di hasilKurang ke layar. <p>Tampilan di layar:</p> 
	Campuran		<ul style="list-style-type: none"> Lakukan operasi dan hasil operasinya disimpan dalam variabel dengan nama hasil. Tampilkan nilai variabel hasil ke layar. <p>Tampilan di layar:</p> 
Operasi perbandingan	Sama dengan		<p>Memeriksa apakah nilai dari variabel umur sama dengan 40 atau tidak.</p> <p>Hasil pemeriksaan:</p> <p>Jika variabel umur diset dengan nilai = 40, hasil pemeriksaan adalah true (benar), selain itu nilainya adalah false (salah).</p> <p>Tampilan di layar:</p> <p>Tidak ada yang ditampilkan ke layar karena tidak ada blok [write] atau [writeLine].</p>

Perintah	Variasi	Contoh Blok Simbol Perintah Blockly	Makna Potongan Kode Program
	Lebih kecil		<p>Memeriksa apakah nilai dari variabel umur kurang dari 40 atau tidak.</p> <p>Hasil pemeriksaan: Jika variabel umur diset dengan nilai kurang dari 40, hasil pemeriksaan adalah true (benar), selain itu, nilainya adalah false (salah).</p> <p>Tampilan di layar: Tidak ada yang ditampilkan ke layar karena tidak ada blok [write] atau [writeLine].</p>
	Lebih besar		<p>Memeriksa apakah nilai dari variabel umur lebih dari 40 atau tidak.</p> <p>Hasil pemeriksaan: Jika variabel umur diset dengan nilai kurang dari 40, hasil pemeriksaan adalah true (benar), selain itu maka nilainya adalah false (salah).</p> <p>Tampilan di layar: Tidak ada yang ditampilkan ke layar karena tidak ada blok [write] atau [writeLine].</p>
Operasi boolean	And		<p>Memeriksa apakah nilai dari variabel umur lebih dari 40 dan apakah nilai dari variabel namaSiswa adalah "Budi".</p> <p>Jika variabel umur bernilai 10, dan namaSiswa = "Budi", keseluruhan blok ini bernilai false(salah).</p>

Perintah	Variasi	Contoh Blok Simbol Perintah Blockly	Makna Potongan Kode Program
			<p>Jika variabel umur bernilai 50 dan namaSiswa = "Budi", keseluruhan blok ini bernilai apa?</p> <p>Keseluruhan blok akan bernilai true (benar).</p> <p>Tampilan di layar: Tidak ada yang ditampilkan ke layar karena tidak ada blok [write] atau [writeLine].</p>
	Or		<p>Memeriksa apakah nilai dari variabel hitung kurang dari atau sama dengan 40, atau nilai dari variabel namaSiswa adalah "Budi".</p> <p>Jika variabel umur bernilai 10, dan namaSiswa = "Anto", keseluruhan blok ini bernilai true(benar).</p> <p>Jika variabel umur bernilai 60, dan namaSiswa = "Budi", keseluruhan blok ini bernilai true(benar).</p> <p>Jika variabel umur bernilai 60 dan namaSiswa = "Anto", keseluruhan blok ini bernilai apa?</p> <p>Keseluruhan blok akan bernilai false (salah).</p> <p>Tampilan di layar: Tidak ada yang ditampilkan ke layar karena tidak ada blok [write] atau [writeLine].</p>
	Not (negasi)		<p>Memeriksa apakah nilai dari variabel namaSiswa bukan "Budi".</p>

Perintah	Variasi	Contoh Blok Simbol Perintah Blockly	Makna Potongan Kode Program
			<p>Jika nilai dari variabel namaSiswa adalah "Anto", keseluruhan blok ini bernilai apa?</p> <p>Keseluruhan blok akan bernilai false (salah).</p> <p>Tampilan di layar: Tidak ada yang ditampilkan ke layar karena tidak ada blok [write] atau [writeLine].</p>
	Campuran		<p>Memeriksa apakah nilai dari variabel namaSiswa adalah "Budi" dan umur kurang dari 14, atau namaSiswa adalah "Budi" dan kelas = 8.</p> <p>Jika nilai namaSiswa adalah Budi, umur = 10, dan kelas = 9, keseluruhan blok ini akan bernilai apa?</p> <p>Keseluruhan blok akan bernilai true (benar).</p> <p>Tampilan di layar: Tidak ada yang ditampilkan ke layar karena tidak ada blok [write] atau [writeLine].</p>
Instruksi kondisional	Satu syarat		<p>Memeriksa apakah nilai dari variabel umur kurang dari 13. Jika benar, akan ditampilkan teks "Kids".</p> <p>Jika nilai dari variabel umur = 10, teks apa yang akan ditampilkan?</p> <p>Teks yang akan ditampilkan adalah "Kids"</p> <p>Tampilan di layar:</p> 

Perintah	Variasi	Contoh Blok Simbol Perintah Blockly	Makna Potongan Kode Program
	Dua cabang		<p>Memeriksa apakah nilai dari variabel umur kurang dari 13. Jika benar, akan ditampilkan teks “kids”. Jika salah, akan ditampilkan teks “Not Kids”.</p> <p>Jika nilai dari variabel umur = 17, teks apa yang akan ditampilkan?</p> <p>Teks yang akan ditampilkan adalah “Not Kids”</p> <p>Tampilan di layar:</p> 
	Tiga cabang		<p>Memeriksa nilai dari variabel umur:</p> <p>Jika nilai dari variabel umur kurang dari 13, maka akan ditampilkan teks “Kids”,</p> <p>Jika nilai dari variabel umur lebih dari atau sama dengan 13 dan kurang dari 20, akan ditampilkan teks “Teenager”.</p> <p>Selain itu, akan ditampilkan teks “adult”.</p> <p>Jika nilai dari variabel umur = 13, teks apa yang akan ditampilkan?</p> <p>Teks yang akan ditampilkan adalah “Teenager”.</p> <p>Tampilan di layar:</p> 

Perintah	Variasi	Contoh Blok Simbol Perintah Blockly	Makna Potongan Kode Program
Instruksi perulangan	Berapa kali diulang		<p>Menampilkan teks “Blockly” berulang-ulang sebanyak sepuluh kali.</p> <p>Tampilan di layar:</p> 
	Perulangan dengan syarat		<p>Keseluruhan blok ini menampilkan nilai dari variabel hitung selama nilainya masih kurang dari atau sama dengan 10. Dengan kata lain, akan ditampilkan angka 1 sampai dengan 10.</p> <p>Tampilan di layar:</p> 

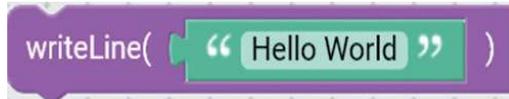
Jawaban Aktivitas AP-K8-07: Hello World

Analisis

Sebelum masuk ke kode program, perlu dibahas dahulu jawaban dari pertanyaan-pertanyaan yang ada pada bagian analisis di Buku Siswa. Jawaban untuk pertanyaan-pertanyaan tersebut adalah sebagai berikut.

1. Pada program ini, tidak dibutuhkan variabel apa pun karena tidak ada nilai yang perlu diganti-ganti.
2. Untuk menampilkan teks ke layar, dapat digunakan blok [write] atau [writeLine]. Perbedaan dari kedua blok tersebut ialah posisi kursor setelah teks ditampilkan. Dengan [writeLine], kursor akan pindah ke baris baru, sehingga teks lain yang akan ditampilkan, akan ada di baris selanjutnya.

Kode Program



```
writeLine( " Hello World " )
```

Kode program di atas digunakan untuk menampilkan teks “Hello World!” ke layar. Sesuai dengan penjelasan sebelumnya pada bagian analisis, ada dua buah blok yang dapat digunakan untuk menampilkan teks ke layar, yaitu blok [write] dan blok [writeLine].

Jawaban Aktivitas AP-K8-08: Hello Namaku

Analisis

Sebelum masuk ke kode program, perlu dibahas dahulu jawaban dari pertanyaan-pertanyaan yang ada pada bagian analisis di Buku Siswa. Jawaban untuk pertanyaan-pertanyaan tersebut adalah sebagai berikut.

1. Pada program ini, ada nilai yang dapat diganti-ganti, yaitu namaSiswa. Oleh karena itu, dibutuhkan variabel di dalam program ini.
2. Variabel yang dibutuhkan dalam program ini ialah variabel untuk menyimpan namaSiswa.



```
set namaSiswa to " Adhi "
```

```
writeLine( namaSiswa )
```

Kode program di atas digunakan untuk menampilkan nilai dari variabel **namaSiswa** ke layar. Sesuai dengan penjelasan sebelumnya pada bagian analisis, variabel ini digunakan untuk menyimpan namaSiswa.

6. Pertemuan 6: Pemrograman Prosedural – Pengulangan (2 jp)

Tujuan Pembelajaran:

1. Siswa mampu membaca dan memahami makna blok penyusun program dalam bahasa Blockly:
 - a. Variabel, input, output
 - b. Ekspresi matematika, ekspresi logika dan perhitungannya
 - c. Percabangan
 - d. Pengulangan

2. Siswa mampu menyelesaikan persoalan dengan menyusun program prosedural dengan bahasa Blockly.
 - a. Membuat spesifikasi input, *output*, proses
 - b. Menganalisis dan mengembangkan solusi
 - c. Menyusun kode program yang sesuai:
 - i. Melakukan *drag and drop* blok pemrograman yang tersedia untuk menyusun sebuah program.
 - ii. Menjalankan dan melihat hasil eksekusi program yang dibuat.
 - iii. Membuat program yang menerima input, dan menyimpannya dalam sebuah variabel.

Apersepsi

Pada pertemuan sebelumnya, peserta didik sudah mempelajari pengulangan melalui permainan maze. Sebagai contoh, untuk mencapai tujuan *Sprite* akan dibuat untuk secara berulang-ulang melakukan hal berikut ini sampai mencapai titik target.

1. Bergerak maju.
2. Kemudian, memeriksa apakah ada jalan di arah kiri. Jika ada jalan di arah kiri, *Sprite* akan berbelok ke kiri.

Pada pertemuan ini, peserta didik akan belajar untuk membuat program yang menggunakan pengulangan, dan menampilkan hasilnya dalam bentuk teks.

Konsep

Terdapat dua jenis pengulangan di dalam pemrograman komputer, yaitu pengulangan sebanyak n kali, dan pengulangan berdasarkan suatu syarat. Pada Blockly, kedua jenis pengulangan tersebut dapat dibuat dengan menggunakan dua buah blok, yaitu blok [repeat n times], dan blok [repeat..while].

Berikut contoh blok [repeat n times]. peserta didik dapat menampilkan teks sebanyak 3 kali dengan cara yang ditunjukkan oleh Gambar 7.2. Angka 3 menunjukkan banyaknya pengulangan yang akan dilakukan. peserta didik bisa mengganti angka ini dengan angka lain yang menunjukkan berapa kali pengulangan akan mereka lakukan. Sementara itu, “print Belajar Blockly” ialah tindakan yang akan dilakukan. Hasil dari proses pengulangan ini ditunjukkan pada Gambar 7.2 (b).



Gambar 7.2: (a) Contoh Penggunaan Blok Repeat n Times (b) Contoh Hasilnya

Kebutuhan Sarana dan Prasarana

- a. Komputer/*Smartphone* yang ter-*install* sistem operasi dan *browser*
- b. Koneksi internet

Kegiatan Inti

Pada pertemuan ini, kegiatan yang dapat dilakukan di kelas adalah sebagai berikut.

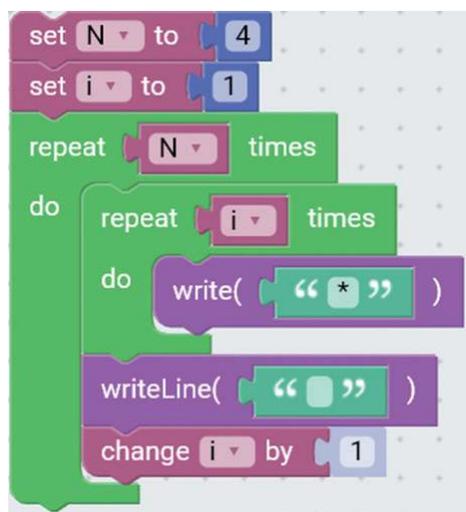
1. Guru menjelaskan contoh kode program untuk perintah pengulangan yang ada pada Tabel Contoh Perintah dalam Bahasa Blockly di Buku Siswa.
2. Siswa diminta untuk mencoba untuk membuat kode program untuk perintah pengulangan yang ada pada tabel yang sama.
3. Setelah itu, peserta didik diarahkan untuk mengerjakan Aktivitas AP-K8-09: Print Pola 1 sampai N Tanda Bintang, dan Aktivitas AP-K8-10: Print Pola N sampai 1 Tanda Bintang.

Jawaban Aktivitas AP-K8-09: Print Pola 1 sampai N Tanda Bintang

Sebelum masuk ke kode program, guru perlu menjelaskan terlebih dahulu *output* untuk $N = 7$ yang ditanyakan pada bagian contoh input dan output. Berikut ini adalah *output* untuk $N = 7$:

```
*
**
***
****
*****
*****
*****
*****
```

Kode program yang lengkap untuk aktivitas ini dapat dilihat pada Gambar 7.3.



Gambar 7.3 Kunci Jawaban untuk Aktivitas AP-K8-09

Sesuai analisis yang sudah diberikan pada Buku Siswa, kode program pada Gambar 7.2 dapat dijelaskan sebagai berikut.

1. Terdapat dua buah variabel dalam program ini, yaitu variabel N dan variabel i. Variabel N adalah variabel yang menjadi input. Nilai 4 diberikan jika akan dibuat pola:

```
*  
**  
***  
****
```

Nilai dari variabel N dapat diganti-ganti sesuai banyaknya baris yang akan ditampilkan dalam pola.

Variabel i menyatakan nomor baris dari pola yang akan ditampilkan.

2. Untuk dapat membuat pola tersebut, kerangka berikut ini harus diikuti.

Mengulang sebanyak N:
 Mengulang print sederet * ke kanan sebanyak i
 Nilai i di-set ke nilai selanjutnya.

Karena itulah, pada kode program, terdapat dua buah pengulangan. Pengulangan yang ada di bagian luar ialah pengulangan sebanyak N baris.

Pada tiap baris, akan dilakukan pengulangan lagi untuk menampilkan tanda * ke kanan sebanyak i buah. Bagian ini ditunjukkan oleh bagian: `repeat i times`, dan `write (“*”)`.

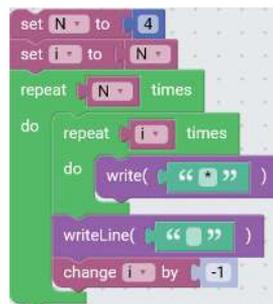
Perubahan nilai i ke nilai selanjutnya ditunjukkan oleh baris terakhir dari program.

Jawaban Aktivitas AP-K8-10: Print Pola N sampai 1 Tanda Bintang

Sebelum masuk ke kode program, guru perlu menjelaskan terlebih dahulu *output* untuk $N = 7$ yang ditanyakan pada bagian contoh input dan *output*. Berikut ini adalah *output* untuk $N = 7$:

```
*****  
*****  
*****  
****  
***  
**  
*  
*
```

Kode program lengkap untuk aktivitas ini ditunjukkan oleh Gambar 7.4.



Gambar 7.4 Kunci Jawaban untuk Aktivitas AP-K8-09

Kode program pada Gambar 7.4 ditulis berdasarkan analisis variabel berikut ini.

1. Banyaknya baris yang harus ditampilkan bervariasi bergantung pada nilai N .
2. Banyaknya * yang harus ditampilkan pada setiap baris dari kiri ke kanan, juga harus bervariasi:
 - a) Baris ke-1 : N buah *
 - b) Baris ke-2 : $N-1$ buah *
 - c) Baris ke-3 : $N-2$ buah *

- d) ...
- e) Baris ke- i : $(N-i-1)$ buah *
- f) ...
- g) Baris ke- $(N-1)$: 2 buah *
- h) Baris ke- N : 1 buah *

Maka, peserta didik perlu mendefinisikan 2 buah variabel, yaitu N yang merupakan banyaknya baris, dan variabel i yang merupakan nomor baris. Variabel N diperoleh melalui input, sedangkan variabel i harus di-*set* nilainya.

Berdasarkan analisis variabel di atas, kerangka blok yang akan peserta didik susun ialah sebagai berikut.

```
Mengulang sebanyak N:
    Mengulang print sederet * ke kanan sebanyak i. Mula-mula, nilai i = N
    Nilai i di-set ke nilai sebelumnya.
```

Jadi, aktivitas ini mengandung pengulangan di dalam sebuah pengulangan.

Catatan penting: karena kode program untuk aktivitas ini mirip dengan kode program pada Aktivitas AP-K8-09, peserta didik disarankan untuk meng-*copy* hasil pekerjaannya, kemudian memodifikasinya dan menyimpannya dalam *file* lain. Akan tetapi, perlu diberitahukan kepada peserta didik bahwa menyontek atau menyalin kode program milik orang lain termasuk dalam tindakan plagiarisme.

Jawaban Aktivitas AP-K8-11: Print Pola Diamond (Materi Pengayaan)

Sebelum masuk ke kode program, guru perlu menjelaskan terlebih dahulu *output* untuk $N = 11$ yang ditanyakan pada bagian contoh input dan *output*. Berikut ini adalah *output* untuk $N = 11$:

```
-----*-----
----***-----
---*****---
--*****--
-*****-
*****
*****-
--*****--
---*****---
----***-----
-----*-----
```

Kode program yang lengkap untuk aktivitas ini ditunjukkan oleh pada Gambar 7.4.

```
set N to 11
set i to 1
set rentang to 0
repeat while i ≤ N
do
  set counter to 1
  repeat while counter ≤ N
  do
    set batasKiri to (N + 1) ÷ 2 - rentang
    set batasKanan to (N + 1) ÷ 2 + rentang
    if counter ≥ batasKiri and counter ≤ batasKanan
    do
      write( "*" )
    else
      write( "-" )
    change counter by 1
  writeLine( "" )
  change i by 1
  if i ≤ (N + 1) ÷ 2
  do
    change rentang by 1
  else
    change rentang by -1
```

Gambar 7.5 Kunci Jawaban untuk Aktivitas AP-K8-11

Kode program pada Gambar 7.5 diperoleh berdasarkan analisis yang terdapat di Buku Siswa halaman 40 (bab 7).

7. Pertemuan 7: Bermain dengan Robot Ozobot (2 jp) – Materi Tambahan

Tujuan Pembelajaran:

Memahami dan mengenal cara kerja robot “*line follower*” dan mengeksplorasi perilaku robot.

Apersepsi

Modul ini adalah modul tambahan apabila sekolah memang memiliki robot Ozobot atau robot *line follower* lainnya serta memiliki waktu yang cukup untuk diberikan pada peserta didik atau menjadi pembelajaran mandiri peserta didik. Modul ini siap pakai apabila sekolah memang memiliki robot Ozobot.

Salah satu alat yang dipakai dalam Revolusi Industri 4.0 adalah robotika. Robotika adalah hasil penelitian disiplin ilmu di bidang pengetahuan komputer dan mekanik. Untuk menguasai mesin robot, manusia dituntut untuk memahami terutama Teknik Informasi dan Komputasi, Teknik Komputer, Mekanika Teknik dan Teknik Elektro. Saat ini, robot dan kecerdasan buatan sudah sangat umum dijumpai sehari-hari. Ozobot adalah mainan kecil berbentuk robot canggih. Ozobot dapat dipakai untuk melatih pemrograman dasar secara menyenangkan. Ozobot dapat membaca garis dan warna. Ozobot juga dapat digunakan di atas tablet digital atau di atas benda fisik seperti kertas. Instruksi dapat diberikan pada Ozobot untuk menyelesaikan perintah-perintah yang dimasukkan ke dalam *memory*-nya.

Pemanasan

Diberikan video simulasi Robot Ozobot.

- <https://ozobot.com/educate>
- <https://ozobot.com/create/challenges>
- <https://ozobot.com/educate/lessons>

Kebutuhan Sarana dan Prasarana

- a. Komputer/*Smartphone* yang ter-*install* sistem operasi dan *browser*
- b. Koneksi internet
- c. Robot Ozobot

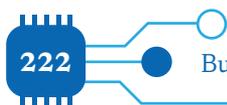
Kegiatan Inti

Apabila sekolah tidak dapat menyediakan robot Ozobot, guru dapat mencoba implementasi secara *unplugged*. peserta didik dibagi menjadi 3 orang per kelompok dan masing-masing akan memiliki peran yang berbeda, yaitu:

- (1) menjadi program yang akan memberi perintah pada robot
- (2) menjadi robot yang akan menerima perintah dari program
- (3) menjadi tester yang akan memastikan apakah robot bergerak sesuai perintah

Kelompok peserta didik ini dapat saling bekerja sama untuk menjalankan aktivitas yang ada. Apabila memungkinkan lembar kerja yang tadinya berupa garis di kertas dapat dilakukan secara nyata dengan membuat pola sejenis di ruang kelas.

Apabila sekolah dapat menyediakan robot Ozobot, hal ini dapat dilakukan langsung menggunakan robotnya. Namun, sebelum menggunakan Robot Ozobot, peserta didik perlu tahu bagaimana menggunakan dan merawat robot ini. Lalu, akan mencoba menjalankan robot Ozobot secara *unplugged*.



Guru perlu menjelaskan juga istilah ini untuk memudahkan dalam penjelasan robot Ozobot.

1. Berpikir sekuensial

Setiap kegiatan rutin yang dilakukan sehari-hari, pasti memiliki rangkaian urutan langkah yang sudah teratur. Kegiatan pertama akan memengaruhi kegiatan kedua, dan selanjutnya akan memengaruhi kegiatan ketiga, dan begitu seterusnya.

Sebagian dari langkah-langkah tersebut tidak bisa ditukar urutannya. Nanti di dalam penulisan *coding* dan pengendalian robot, akan lebih jelas sekuensial ini.

2. Sintaksis

Sintaksis berasal dari bahasa Yunani yang mempunyai arti “menempatkan”. Di sini, peserta didik belajar pengetahuan linguistik dengan cara menempatkan kata-kata agar dapat dimengerti. Di dalam bahasa Indonesia, kata-kata ditempatkan menjadi satu kalimat yang terdiri atas subjek, predikat, dan objek. Contohnya:

- Kucing merah makan tikus.
- Kucing makan tikus merah.
- Tikus merah makan kucing.

Nanti, di dalam penulisan *coding* dan pengendalian robot, peserta didik akan lebih banyak berlatih banyak tentang sintaksis ini.

3. Debugging

Metode yang dilakukan oleh para penulis coding untuk menyempurnakan algoritma pemrograman. Hasil akhir dari *debugging* ialah aplikasi dapat berjalan sesuai dengan harapan.

Sama dengan pembuatan *coding* dan pengendalian robot. Nanti, bisa dilihat apakah ada langkah-langkah yang kurang tepat atau harus ditambahkan. Semua ini dilakukan untuk makin menyempurnakan penulisan dan cara berpikir sekuensial.

Guru mengarahkan peserta didik untuk melaksanakan kegiatan Aktivitas AP-K8-12-U: Garis Lajur Ozobot, Aktivitas AP-K8-13-U: Lajur Warna Ozobot, Aktivitas AP-K8-14-U: Kode untuk mengatur kecepatan Ozobot pada Buku Siswa dengan membagi peserta didik dalam kelompok. Setiap kelompok terdiri atas 4 peserta didik. Waktu pengerjaan sekitar 60 menit. Sesudahnya, guru dapat berdiskusi dengan peserta didik terkait aktivitas yang ada.

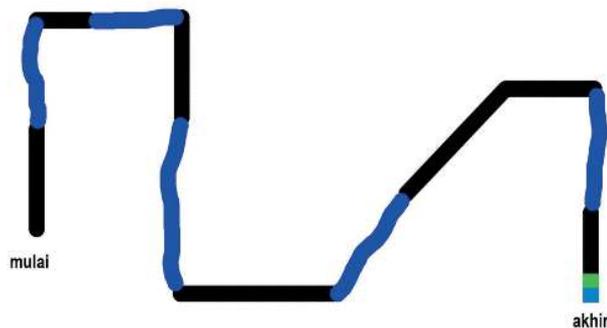
Jawaban Aktivitas AP-K8-14-U: Garis Lajur Ozobot

Latihan-1

1. Sensor baca garis robot Ozobot terdiri atas beberapa mata. Robot Ozobot akan berusaha berjalan tetap berada di jalurnya dengan mengandalkan mata di sisi paling luar. Penyesuaian akan selalu dilakukan sesuai bentuk garis yang dilalui apakah roda kiri atau roda kanan berputar lebih cepat.
2. Sensor mata robot Ozobot tidak dapat membaca garis tajam setipis bolpen. Beberapa sensor mata yang berada di tengah harus bisa membaca garis agar robot Ozobot tahu garisnya bergerak atau berubah ke sisi mana.
3. Jika semua sensor mata robot Ozobot membaca warna yang sama, Robot Ozobot akan menganggap dirinya berdiri dalam bidang luas. Robot tidak akan tahu arah dan posisinya ada di mana.

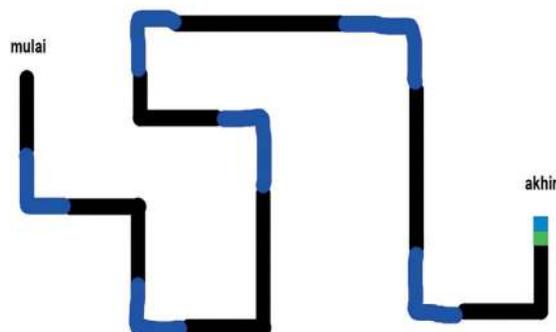
Latihan-2

Garis pelengkap bisa berbagai cara, ini contoh yang dapat digunakan sebagai inspirasi.

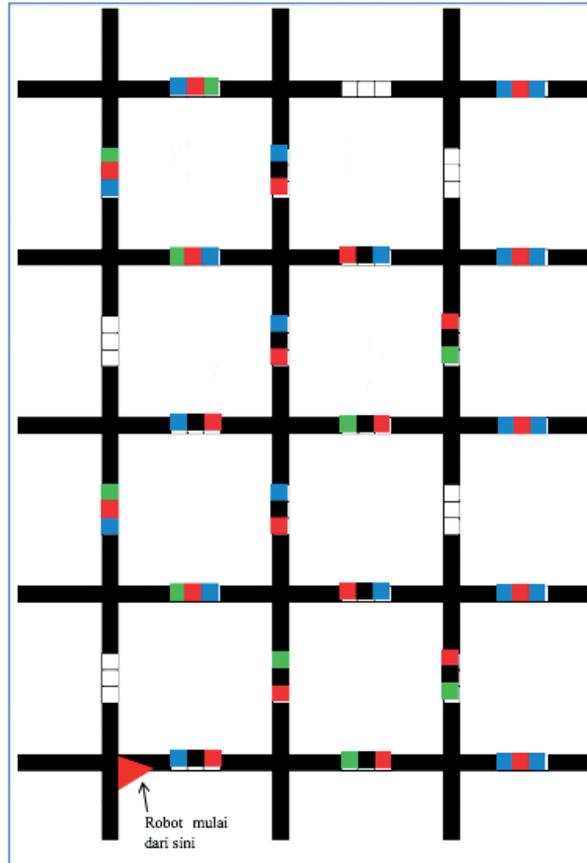


Latihan-3

Garis pelengkap bisa berbagai cara, ini contoh yang dapat digunakan sebagai inspirasi



Latihan-4



Warnai peta di atas, dan usahakan ozobot dapat melalui semua garis tanpa berbelok ke jalan buntu

- Di dalam soal ini, peserta didik ditantang untuk melewati batang garis sebanyak mungkin. Tidak mungkin robot dapat melalui semua batang garis.

H. Metode Pembelajaran Alternatif

Pembelajaran pada bab ini hampir semuanya membutuhkan sarana dan prasarana komputer. Guru dapat berkreasi dengan memanfaatkan materi *unplugged* pada bahan pengayaan. Guru juga dapat membuat blok-blok kode menjadi kartu-kartu yang dapat dimainkan oleh peserta didik. Untuk materi tambahan terkait robot Ozobot dapat menggunakan robot *line follower* lainnya atau melakukan secara *unplugged* seperti yang telah dijelaskan pada panduan pembelajarannya.