



PENGARUH KEGIATAN *PLUGGED* DAN *UNPLUGGED* TERHADAP BERPIKIR KOMPUTASIONAL MURID SMP

Yohan Adi Setiawan¹; Sariaman Siringo Ringo²

¹ Sekolah Menengah Pertama Negeri 2 Kalitidu, Bojonegoro, Jawa Timur, Indonesia

² Sekolah Menengah Pertama Negeri Satu Atap 1 Pengabuan, Jambi, Indonesia

¹Contributor Email: yohanadisetiawan@gmail.com

Received: June 17, 2024

Accepted: September 18, 2024

Published: November 30, 2024

Article Url: <https://ojsdikdas.kemdikbud.go.id/index.php/didaktika/article/view/1680>

Abstract

Computational thinking has become a major focus in education in many countries, including Indonesia. As a foundation for developing students' computational thinking, informatics learning can be done using unplugged and plugged methods. Students are still at the basic stage of Computational Thinking in Informatics learning. Based on several previous studies obtained conflicting results regarding the influence of plugged and unplugged methods on students' computational thinking. The present study endeavors to assess the impact of both plugged and unplugged activities on the computational thinking proficiencies of junior high school students. This quasi-experimental research enlisted two 7th-grade cohorts comprising a cumulative total of 64 students. Each cohort, comprising 32 students, was assigned to a control group (unplugged activities) and an experimental group (plugged activities). Based on pretest and posttest tests in groups, this research reveals that both unplugged and plugged activities have a big influence on improving junior high school students' computational thinking skills. Meanwhile, the results of the Mann-Whitney test show that there is no difference between the effect of the plugged method and the unplugged method on computational thinking skills. This research indicates that teachers can choose one of the Informatics learning methods, either plugged or unplugged, or combine the two methods.

Keywords: *Plugged Activities; Unplugged Activities; Computational Thinking; Computer Science; Informatics*

Abstrak

Berpikir komputasional menjadi fokus utama dalam pendidikan di banyak negara, termasuk di Indonesia. Pembelajaran Informatika sebagai fondasi pengembangan berpikir komputasional murid dapat dilaksanakan dengan menggunakan metode unplugged dan plugged. Dimana murid masih pada tahap dasar terhadap Berpikir Komputasional dalam pembelajaran Informatika. Berdasarkan beberapa penelitian terdahulu memperoleh hasil yang bertentangan terkait pengaruh metode plugged dan unplugged terhadap keterampilan komputasional murid. Penelitian ini bertujuan menguji pengaruh kegiatan plugged dan unplugged terhadap berpikir komputasional murid SMP. Penelitian kuasi eksperimen ini melibatkan dua kelas 7 dengan total 64 murid sebagai sampel. Selanjutnya, masing-masing kelas terdiri dari 32 murid yang disebut sebagai kelas kontrol (unplugged) dan kelas eksperimen (plugged). Berdasarkan uji pretest dan posttest dalam kelompok, penelitian ini mengungkapkan bahwa baik kegiatan unplugged maupun plugged berpengaruh besar dalam peningkatan keterampilan berpikir komputasional murid SMP. Sementara itu, hasil uji Mann-Whitney menunjukkan bahwa tidak ada perbedaan antara pengaruh metode unplugged dan metode plugged terhadap keterampilan berpikir komputasional. Hasil penelitian ini mengindikasikan bahwa guru dapat memilih salah satu metode pembelajaran Informatika baik unplugged maupun plugged atau mengombinasikan kedua metode tersebut.

Kata Kunci: *Kegiatan Plugged; Kegiatan Unplugged; Berpikir Komputasional; Ilmu Komputer; Informatika*

A. Pendahuluan

Berdasarkan hasil observasi di beberapa Sekolah Menengah Pertama (SMP) di Kabupaten Bojonegoro, rata-rata kegiatan pembelajaran Informatika dilaksanakan dengan menggunakan metode *plugged*. Menurut literatur, metode *unplugged* juga dapat digunakan dalam pembelajaran Informatika. Pemilihan metode pembelajaran ini sangat berpengaruh terhadap keberhasilan pengembangan berpikir komputasional murid. Kegiatan belajar mengajar yang monoton tanpa memperhatikan kesiapan murid untuk menerima pelajaran, memicu timbulnya rasa jenuh dalam proses pembelajaran.

Salah satu keterampilan kognitif yang penting dalam pengembangan generasi saat ini dan akan datang adalah berpikir komputasional (Zapata-Cáceres et al., 2024; Zhang & Nouri, 2019). Wing (2006) bahkan menyatakan bahwa berpikir komputasional sama pentingnya dengan keterampilan

menulis, membaca dan berhitung. Hal ini disebabkan oleh perubahan teknologi yang berkembang dengan pesat pada abad 21, yang akhirnya mengubah keahlian yang diperlukan untuk berbagi jenis pekerjaan (Osman & Lay, 2022). Banyak peneliti dan pendidik telah mengembangkan atau fokus pada berpikir komputasional selama beberapa dekade terakhir (Saxena et al., 2020). Sejalan dengan itu, sejumlah negara telah mulai mengintegrasikan berpikir komputasional ke dalam sistem pendidikan mereka, seperti Amerika, Australia, China, Kanada, Malaysia, dan Swedia (Chongo et al., 2021; Dong et al., 2024; Zhang & Nouri, 2019). Menyikapi hal ini, Indonesia juga telah menetapkan berpikir komputasional sebagai salah satu capaian pembelajaran di kurikulum Merdeka.

Salah satu mata pelajaran baru yang muncul semenjak berlakunya kurikulum Merdeka adalah Informatika. Kata Informatika merupakan padanan kata dari *Computer Science*. Kehadiran Informatika diharapkan dapat menjadi fondasi dalam pengembangan berpikir komputasional sehingga murid dapat menyelesaikan berbagai persoalan di masa depan secara logis, kritis, kreatif dan sistematis. Meskipun berpikir komputasional sangat erat kaitannya dengan dunia komputer, tetapi berpikir komputasional bukanlah keterampilan yang hanya dibutuhkan oleh orang-orang yang ingin menekuni bidang komputer (Li et al., 2020). Hal ini selaras dengan Wing (2006) yang berpendapat bahwa berpikir komputasional merupakan kemampuan dasar yang perlu dimiliki setiap orang.

Hingga saat ini, para peneliti terus mencoba mendefinisikan berpikir komputasional, sehingga istilah berpikir komputasional memiliki pengertian yang luas. Meskipun demikian, pada umumnya, mereka merujuk pada definisi berpikir komputasional yang diperkenalkan oleh Jeannete M. Wing dalam artikel yang berjudul "*Computational Thinking*". Wing (2006) menyebutkan bahwa berpikir komputasional adalah keterampilan yang menggunakan konsep-konsep dasar ilmu komputer untuk memahami perilaku manusia (*understanding human behavior*), merancang sistem (*designing system*), dan memecahkan masalah (*problem solving*). Selanjutnya, (Wing, 2011) mendefenisikan kembali berpikir komputasional sebagai proses

menciptakan dan menyelesaikan masalah sehingga baik komputer maupun manusia dapat melaksanakannya. Terinspirasi dari Wing, Selby & Woollard (2013) mengusulkan bahwa berpikir komputasional merupakan sebuah pendekatan yang berfokus pada pemecahan masalah dengan melibatkan proses abstraksi (*abstraction*), algoritma (*algorithms*), dekomposisi (*decomposition*), evaluasi (*evaluation*), dan generalisasi (*generalizations*) atau pengenalan pola (*patterns*). Definisi yang disajikan oleh Wing (2011), Wing (2006), dan diperluas oleh Selby & Woollard (2013) memberikan landasan yang kuat untuk pemahaman dan pengembangan konsep berpikir komputasional bagi para pendidik dan peneliti.

Sejumlah penelitian telah dilakukan untuk menguji pengaruh metode atau strategi pembelajaran terhadap berpikir komputasional murid. Pada pembelajaran Informatika, aktivitas pembelajaran yang dilakukan dapat berupa kegiatan belajar dengan menggunakan komputer (*plugged*) dan kegiatan belajar yang tidak memerlukan komputer atau peralatan digital (*unplugged*) (Voon et al., 2023). Beberapa penelitian telah membahas pengaruh kedua metode mengajar ini terhadap berpikir komputasional murid di berbagai jenjang pendidikan, seperti Saxena et al. (2020) di PAUD, Erümit & Sahin (2020) di SD, Chongo et al. (2021) di SMA, Voon et al. (2023) di Universitas. Namun, pada tingkat SMP, masih terdapat kekurangan penelitian yang menyelidiki efektivitas metode *plugged* dan *unplugged* dalam pengembangan berpikir komputasional murid.

Pada metode *plugged*, salah satu media yang sering digunakan adalah Scratch. Scratch adalah bahasa pemrograman visual yang dirancang khusus untuk pemula dalam dunia pemrograman (Dúo-Terrón, 2023). Melalui Scratch, pengguna dapat membuat beragam bentuk proyek, seperti permainan, animasi, dan cerita interaktif (Fagerlund et al., 2021). Scratch bahkan dapat digunakan untuk membuat kuis. Sejumlah penelitian terdahulu telah menguji pengaruh Scratch terhadap berpikir komputasional murid. Piedade & Dorotea (2023) yang melibatkan 189 murid kelas 4 SD sebagai sampel penelitian menemukan bahwa penggunaan Scratch berpengaruh signifikan terhadap berpikir komputasional murid. Penemuan ini sesuai dengan hasil

yang dilaporkan oleh Pérez-Marín et al. (2020) dan Molina-Ayuso et al. (2022) yang melibatkan murid SD dan mahasiswa secara berurutan.

Di sisi lain, penelitian terdahulu juga menunjukkan bahwa metode *unplugged* juga memiliki kelebihan dalam pengembangan berpikir komputasional murid (del Olmo-Muñoz et al., 2020; Merino-Armero et al., 2022). Media pembelajaran yang paling sering digunakan pada metode *unplugged* adalah permainan papan (*board game*) dan permainan kartu (*card game*) (Chen et al., 2023). Selain itu, sejumlah penelitian juga menggunakan Bebras (Delal & Oner, 2020), robot berbasis *unplugged* (Miller et al., 2018), pemrograman visual berbasis kertas (Threekunprapa & Yasri, 2020), dan buku teks (Twigg et al., 2019) sebagai media pembelajaran dalam penerapan metode *unplugged*.

Meskipun demikian, terdapat perdebatan tentang metode mana yang lebih baik. Beberapa penelitian eksperimen yang mencoba membandingkan antara metode *plugged* dan *unplugged* memperoleh hasil yang berbeda. Piedade & Dorotea (2023) dan Sigayret et al. (2022) menemukan bahwa murid yang menggunakan Scratch menunjukkan keterampilan berpikir komputasional yang lebih baik dibandingkan murid yang mengikuti pembelajaran dengan metode *unplugged*. Di sisi lain, Hermans & Aivaloglou (2017) dan (Rodríguez-Martínez et al., 2020) memperoleh bahwa pengaruh metode *plugged* dengan Scratch dan metode *unplugged* terhadap keterampilan berpikir komputasional murid tidak signifikan.

Hasil dari beberapa penelitian tersebut memperlihatkan adanya inkonsistensi pada hasil penelitian yang terkait dengan metode *plugged* dan metode *unplugged*. Penelitian lebih lanjut diperlukan untuk memberikan bukti empiris guna mengatasi kesenjangan hasil penelitian ini. Oleh karena itu, tujuan utama dalam penelitian kuasi eksperimen ini adalah untuk mengetahui bagaimana metode *unplugged* dan *plugged* mempengaruhi keterampilan berpikir komputasional murid SMP.

B. Metode

Penelitian dilaksanakan pada salah satu Sekolah Menengah Pertama (SMP) di Kabupaten Bojonegoro yang telah menerapkan mata

pelajaran informatika sejak Kurikulum 2013 revisi 2018 sebagai *pilot project* pelaksanaan pembelajaran Informatika. Materi pembelajaran yang digunakan adalah elemen Algoritma dan Pemrograman (AP). Kegiatan pembelajaran dilaksanakan oleh seorang guru yang telah mengajar selama 14 tahun. Pengalaman mengajar mulai dari TIK pada Kurikulum 2006, ICT pada Rintisan Sekolah Berstandar International (RSBI), Bimbingan TIK pada Kurikulum 2013, dan Informatika dimulai semenjak dilaksanakannya *pilot project* pada tahun 2018 pada Kurikulum 2013 dan menjadi mata pelajaran wajib pada pelaksanaan Kurikulum Merdeka pada Fase D dan Fase E. Penelitian kuasi eksperimen ini dilaksanakan selama 4 pertemuan pada mata pelajaran Informatika dan menerapkan *nonequivalent control group design* (Sugiyono, 2017). Murid kelas 7 dari dua kelas dilibatkan sebagai sampel penelitian seperti yang ditampilkan pada Tabel 1. Masing-masing kelas tersebut disebut sebagai kelas kontrol dan kelas eksperimen.

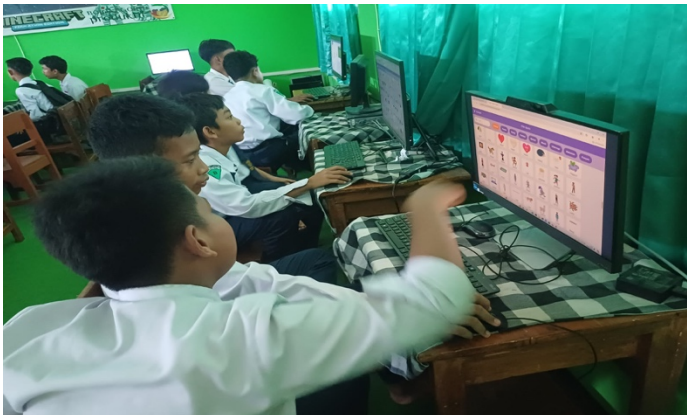
Tabel 1. Profil Murid yang Berpartisipasi dalam Penelitian Ini

Kelas	Laki-Laki	Perempuan	Total
Kontrol	15	17	32
Eksperimen	15	17	32

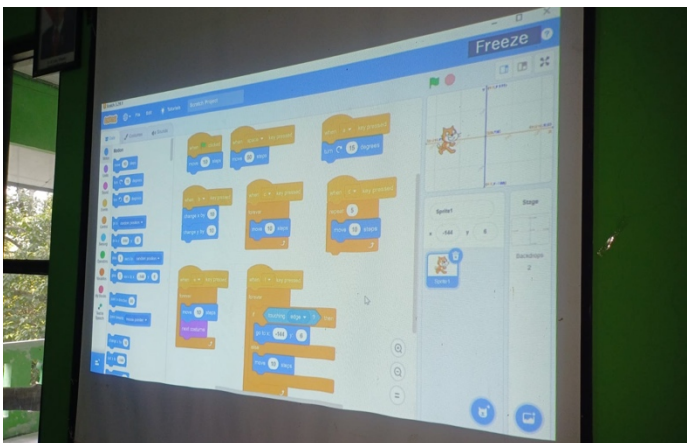
Murid di kelas kontrol melakukan kegiatan *unplugged* (tanpa menggunakan peralatan digital atau komputer). Di sisi lain, murid di kelas eksperimen mengikuti pembelajaran Informatika yang menggunakan metode *plugged* (menggunakan komputer). Pada metode ini, murid menggunakan Scratch. Pemilihan Scratch sebagai media yang digunakan pada kegiatan *plugged* karena Scratch merupakan bahasa pemrograman berbasis visual yang paling sering dan mudah digunakan dibandingkan bahasa pemrograman yang serupa.



Gambar 1. Proses Pembelajaran dengan Menggunakan Metode Unplugged



Gambar 2. Proses Pembelajaran dengan Menggunakan Metode Plugged



Gambar 3. Blok-Blok Kode pada Scratch

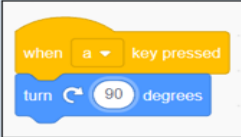
Sebelum dan setelah mengikuti kegiatan pembelajaran Informatika, semua murid di kelas kontrol dan kelas eksperimen mengerjakan soal Algoritma dan Pemrograman (AP). Soal tersebut terdiri atas 10 butir soal pilihan ganda yang disusun berdasarkan empat komponen berpikir komputasional, yaitu abstraksi, algoritma, dekomposisi, dan pengenalan pola. Tabel 2 menampilkan sebaran komponen berpikir komputasional pada masing-masing nomor soal.

Tabel 2. Kisi-Kisi Instrumen Tes Berpikir Komputasional

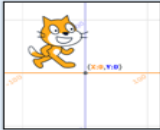
Komponen Berpikir Komputasional	Nomor Soal
Abstraksi	4 dan 10
Algoritma	1, 7, 8, dan 9
Dekomposisi	2 dan 3
Pengenalan Pola	5 dan 6

Gambar 4 menunjukkan salah satu soal yang digunakan untuk mengukur keterampilan berpikir komputasional pada komponen algoritma materi pembelajaran Algoritma dan Pemrograman.

Perhatikan blockly dan posisi sprite "kucing" di bawah ini!




Apabila pemain menekan tombol "a" pada *keyboard* sebanyak tiga kali, maka seperti apakah posisi sprite "kucing"?

a. 

b. 

c. 

d. 

Gambar 4. Sampel Instrumen Tes yang Digunakan

Selanjutnya, hasil *pretest* dan *posttest* digunakan dalam tahap analisis. Proses analisis meliputi analisis deskriptif, statistik inferensial, dan perhitungan *effect size*. Proses pertama dan kedua dilakukan dengan menggunakan aplikasi SPSS, sedangkan proses ketiga dilakukan dengan menggunakan rumus *effect size* ($r = Z/\sqrt{N}$) (Pallant, 2016).

Hasil yang diperoleh dari perhitungan *effect size* dengan menggunakan rumus tersebut dibandingkan dengan kriteria Cohen (1992) yang ditampilkan pada Tabel 3.

Tabel 3. Pengkategorian Effect Size Berdasarkan Kriteria Cohen (1992)

<i>Effect Size</i>	Kategori
0,10	Kecil
0,30	Sedang
0,50	Besar

C. Hasil dan Pembahasan

1. Hasil

Analisis deskriptif dilakukan pada data *pretest* dan *posttest* dari setiap kelas kontrol dan kelas eksperimen (Tabel 4). Jumlah murid tidak mengalami perubahan saat pelaksanaan *pretest* dan *posttest*, yaitu 32 murid.

Tabel 4. Hasil Analisis Deskriptif terhadap Data Pretest Dan Posttest

	N	Minimum	Maximum	Mean	Std. Deviation
<i>Pretest</i> Kelas Kontrol	32	10	70	37,50	16,65
<i>Posttest</i> Kelas Kontrol	32	30	100	77,50	18,83
<i>Pretest</i> Kelas Eksperimen	32	20	60	40,94	9,95
<i>Posttest</i> Kelas Eksperimen	32	20	100	79,37	16,64

Selain itu, data *pretest* dan data *posttest* murid dianalisis berdasarkan komponen berpikir komputasional yang diukur dengan menggunakan instrumen tes. Gambar 5 menunjukkan hasil rata-rata nilai *pretest* dan *posttest* murid pada empat komponen berpikir komputasional yang diukur. Rata-rata nilai *pretest* murid pada kelas kontrol dan kelas eksperimen pada semua komponen berpikir komputasional terlihat hampir sama. Hasil yang sama juga ditunjukkan pada rata-rata nilai *posttest*.



Gambar 5. Hasil Pretest dan Posttest Murid Berdasarkan empat komponen Berpikir Komputasional

Sebelum dilakukan uji inferensial, uji normalitas digunakan untuk menguji persyaratan analisis yang dilakukan dengan menggunakan program SPSS. Semua data hasil *pretest* dari kelas eksperimen dan kontrol tidak berdistribusi normal berdasarkan hasil uji normalitas dengan tes Shapiro-Wilk (Tabel 5). Oleh karena itu, data *pretest* dianalisis dengan uji statistik non parametrik.

Tabel 5. Hasil Tes Shapiro-Wilk pada Data Pretest

Kelas	N	Statistik	Sig.
Kontrol	32	0,89	0,005
Eksperimen	32	0,90	0,007

Uji Wilcoxon Sign Ranks digunakan sebagai tes non parametrik. Hasil analisis ini menunjukkan bahwa murid dalam kelas eksperimen dan kontrol tidak berbeda dalam berpikir komputasional. Dengan demikian, perlakuan dapat dilaksanakan baik di kelas eksperimen maupun di kelas kontrol dalam penerapan kegiatannya.

Tes Shapiro-Wilk digunakan kembali untuk menguji normalitas data *posttest*. Hasil uji ini menunjukkan bahwa data *posttest* dari kelas kontrol dan kelas eksperimen tidak berdistribusi normal (Tabel 6). Oleh karena itu, uji Wilcoxon Sign Ranks digunakan untuk menganalisis perbedaan berpikir komputasional sebelum dan sesudah murid mengikuti pembelajaran Informatika di kelas kontrol dan kelas eksperimen.

Tabel 6. Hasil Tes Shapiro-Wilk Berdasarkan Data *Posttest*

Kelas	N	Statistik	Sig.
Kontrol	32	0,89	0,003
Eksperimen	32	0,73	0,000

Tabel 7 menampilkan hasil tes Wilcoxon Signed Ranks dengan menggunakan data *pretest* dan *posttest* dari kelas kontrol dan kelas eksperimen. Baik di kelas kontrol dan kelas eksperimen, nilai $p = 0,00$ lebih kecil dari 0,05 sehingga H_a diterima dan H_o ditolak. Artinya, terdapat perbedaan antara berpikir komputasional murid saat sebelum dan sesudah mengikuti pembelajaran Informatika baik dengan kegiatan *unplugged* maupun kegiatan *plugged*.

Tabel 7. Hasil tes Wilcoxon Signed Ranks Berdasarkan Data *Pretest* dan *Posttest*

	N	Z	Asymp. Sig. (2-tailed)
<i>Pretest-Posttest</i> Kelas Kontrol	32	-5,24	0,00
<i>Pretest-Posttest</i> Kelas Eksperimen	32	-5,15	0,00

Dengan menggunakan data yang ada pada Tabel 7, *effect size* pada kelas kontrol dan kelas eksperimen diperoleh sebesar 0,65 dan 0,64 secara berurutan. Mengacu pada Tabel 3, dua nilai tersebut mengindikasikan *effect size* yang besar. Dengan demikian, baik kegiatan *plugged* dan kegiatan *unplugged* berpengaruh besar terhadap peningkatan berpikir komputasional murid SMP.

Analisis selanjutnya adalah membandingkan data *posttest* dari kelas kontrol dan kelas eksperimen untuk mengetahui perbedaan berpikir komputasional murid di dua kelas tersebut. Dengan menggunakan uji

Mann-Whitney, nilai p diperoleh sebesar 0,54, sehingga H_a ditolak dan H_o diterima (Tabel 8). Artinya, tidak ada perbedaan berpikir komputasional antara murid di kelas kontrol dan kelas eksperimen. Dengan demikian, kegiatan *plugged* dan kegiatan *unplugged* memiliki pengaruh yang sama besar terhadap berpikir komputasional murid SMP.

Tabel 8. Hasil Uji Mann-Whitney Berdasarkan Data Posttest

	N	Z	Asymp. Sig. (2-tailed)
<i>Posttest</i>			
Kelas Kontrol dan Kelas Eksperimen	64	-0,62	0,54

Terakhir, analisis dilakukan pada setiap komponen berpikir komputasional. Uji *Mann-Whitney* digunakan untuk menganalisis data *posttest*. Tabel 9 menunjukkan bahwa tidak ada perbedaan yang signifikan antara kelas kontrol dan kelas eksperimen pada setiap komponen berpikir komputasional.

Tabel 9. Hasil Uji Mann-Whitney pada Setiap Komponen Berpikir Komputasional

Komponen Berpikir Komputasional	Kelas Kontrol	Kelas Eksperimen	Z	Asymp. Sig. (2-tailed)
	Mean	Mean		
Abstraksi	33,16	31,84	-0,34	0,73
Algoritma	31,55	33,45	-0,43	0,67
Dekomposisi	31,50	33,50	-0,63	0,52
Pengenalan Pola	30,00	35,00	-1,54	0,12

2. Pembahasan

Informatika telah menjadi mata pelajaran wajib bagi murid di jenjang SMP sejak berlakunya kurikulum Merdeka. Melalui pembelajaran Informatika, murid diharapkan dapat mengembangkan keterampilan berpikir komputasional. Secara umum, pembelajaran Informatika dapat dilakukan dengan kegiatan *unplugged* atau kegiatan *plugged*. Peneliti dalam studi ini mengukur seberapa besar pengaruh masing-masing

kegiatan tersebut terhadap berpikir komputasional dan membandingkan berpikir komputasional murid di kelas kontrol dan kelas eksperimen. Hasil analisis pada Tabel 6 dan perhitungan *effect size* membuktikan bahwa baik kegiatan *plugged* dan kegiatan *unplugged* berpengaruh besar terhadap peningkatan berpikir komputasional murid SMP.

Sejumlah penelitian sebelumnya mengungkapkan bahwa kegiatan *unplugged* berpengaruh terhadap peningkatan berpikir komputasional murid, seperti Merino-Armero et al. (2022) dan Tonbuluğlu & Tonbuluğlu (2019). Merino-Armero et al. (2022) memperoleh hasil *effect size* yang besar ($d = 1,31$) berdasarkan kriteria Cohen (1992), seperti yang diperoleh pada penelitian ini. Dalam penelitian tersebut, Merino-Armero et al. (2022) menyimpulkan bahwa penggunaan metode *unplugged* dalam pengembangan berpikir komputasional murid tidak berpengaruh negatif terhadap prestasi akademik murid. Chen et al. (2023) juga mengungkapkan bahwa metode *unplugged* dapat digunakan sebagai salah satu cara dalam upaya pengembangan keterampilan berpikir komputasional murid. Lebih lanjut, metode *unplugged* dapat menjadi solusi bagi sekolah-sekolah yang tidak memiliki fasilitas yang mendukung dalam pembelajaran Informatika, seperti komputer (Huang & Looi, 2020).

Ada banyak bentuk kegiatan yang dapat dilaksanakan saat menggunakan metode *unplugged*, seperti permainan papan (*board game*), permainan kartu (*card game*), robot berbasis *unplugged*, pemrograman visual berbasis kertas, dan buku teks. Pada penelitian ini, peneliti menggunakan blok-blok kode dari Scratch yang dicetak dan dibagikan kepada murid. Peranan guru sangat dibutuhkan pada kegiatan *unplugged* ini karena guru perlu mengonfirmasi benar atau tidak susunan kode yang murid kerjakan. Hal ini sangat berbeda dengan penggunaan Scratch pada metode *plugged*. Ketika selesai mengerjakan suatu proyek melalui Scratch di gawai, murid dapat langsung mengetahui proyek tersebut berhasil atau tidak. Perbedaan pengalaman belajar inilah yang menjadi tantangan bagi murid dan guru dalam menerapkan metode *unplugged*. Guru harus

mampu memberikan konfirmasi benar atau salah susunan kode dari proyek atau pekerjaan murid saat mereka membutuhkannya.

Tantangan yang dihadapi murid dalam kegiatan *unplugged* mendorong murid untuk berdiskusi dengan murid lainnya. Ketika menyusun blok-blok kode, murid akan bertanya kepada teman dalam satu kelompok untuk memastikan pekerjaan *debugging* dalam proses mereka sudah benar atau tidak. Ketika semua anggota kelompok sudah yakin terhadap susunan blok yang dibuat, mereka dapat berkonsultasi kepada guru Informatika untuk mengkonfirmasi kebenaran proyek mereka. Proses ini jarang ditemui pada kegiatan berbasis *plugged*.

Penelitian ini juga memperlihatkan bahwa kegiatan *plugged* berpengaruh dalam meningkatkan keterampilan berpikir komputasional murid SMP. Temuan ini sejalan dengan hasil studi yang didapati oleh Molina-Ayuso et al. (2022), Pérez-Marín et al. (2020), dan Piedade & Dorotea (2023). Penggunaan Scratch dapat berpengaruh dalam pengembangan berpikir komputasional murid karena saat menggunakan Scratch, murid terlibat dengan serangkaian konsep komputasi, seperti *sequences*, *loops*, *events*, *conditionals*, dan *operators* (Brennan & Resnick, 2012).

Brennan & Resnick (2012) menjelaskan bahwa saat menyusun dua atau lebih blok kode pada Scratch, murid harus memperhatikan urutan dari peletakkan blok kode agar blok-blok kode tersebut dapat bekerja sesuai yang diharapkan. Melalui kegiatan ini, murid belajar konsep *sequences*. Ketika murid ingin membuat instruksi yang berulang kali, murid bisa menggunakan blok kode tersebut secara berulang-ulang. Namun, hal ini akan menyebabkan kode blok yang digunakan sangat banyak dan tidak efisien. Oleh karena itu, murid dapat menggunakan konsep *loops* yaitu dengan menggunakan blok ulangi (*repeat*). Dengan penggunaan blok ini, murid dapat melakukan suatu instruksi secara berulang-ulang sebanyak yang diinginkan. Kemudian, murid mempelajari konsep *events* melalui penggunaan blok-blok kode kejadian (*events*). Murid menggunakan blok ini ketika ingin menjalankan suatu instruksi ketika instruksi lain dijalankan. Selanjutnya, murid mempelajari konsep *conditionals*

ketika mereka menggunakan blok-blok kode jika (*if*). Konsep ini digunakan saat murid ingin menjalankan suatu kode ketika suatu kondisi terpenuhi. Murid juga mempelajari konsep operator (*operators*) yang berkaitan dengan penggunaan operator matematika. Semua konsep komputasi yang dipelajari melalui Scratch dijelaskan dengan baik oleh Brennan & Resnick (2012).

Penggunaan Scratch pada metode *plugged* dapat menjadi salah satu pilihan terbaik bagi guru dalam mengembangkan keterampilan berpikir komputasional murid. Pada umumnya, Scratch digunakan untuk anak-anak yang ingin belajar pemrograman. Tampilan menarik dan kemudahan dalam penggunaannya menjadi kelebihan dari Scratch (Cárdenas-Cobo et al., 2021). Selain itu, pada website Scratch, ada banyak proyek Scratch yang dapat dilihat untuk sekedar digunakan atau untuk kegiatan belajar. Meskipun demikian, ada beberapa website atau aplikasi pemrograman lain yang dapat digunakan, seperti Blokly, CodingGame, dan Gamestar Mechanic.

Hasil analisis berikutnya pada penelitian ini ditampilkan pada tabel 8. Temuan ini menunjukkan bahwa keterampilan berpikir komputasional murid dipengaruhi secara seimbang baik menggunakan metode *unplugged* maupun *plugged*. Ditinjau dari masing-masing komponen berpikir komputasional, berdasarkan tabel 9, penelitian ini membuktikan bahwa metode *unplugged* dan metode *plugged* sama baiknya dalam meningkatkan keterampilan abstraksi, algoritma, dekomposisi, dan pengenalan pola. Hasil ini selaras dengan hasil penelitian yang diperoleh Hermans & Aivaloglou (2017) dan Rodríguez-Martínez et al. (2020). Hasil penelitian ini mengimplikasikan bahwa guru memiliki fleksibilitas dalam memilih metode pembelajaran yang diinginkan untuk mengembangkan keterampilan berpikir komputasional murid SMP. Guru dapat menggunakan sumber belajar yang telah ada seperti Scratch dan 4Dframe (Herawati, 2022; Suhardi & Utama, 2022) atau mengembangkan sendiri, seperti yang dilakukan oleh Memolo (2022) dan Nasiba (2022).

Pemilihan antara metode *unplugged* atau metode *plugged* perlu dipertimbangkan sebaik mungkin. Metode *unplugged* cocok digunakan

saat guru mengajarkan materi-materi dasar atau memperkenalkan konsep informatika kepada murid-murid yang baru mengenal komputer (Kotsopoulos et al., 2017). Jika guru memaksakan untuk melaksanakan aktivitas *plugged* tanpa mempertimbangkan kesiapan belajar murid, murid berkemungkinan besar akan menghadapi kendala-kendala yang menghambat proses belajar mereka. Beberapa murid mungkin saja memiliki kekhawatiran saat pertama kali mengoperasikan komputer. Hal ini dapat terjadi kepada murid yang baru pertama kali melihat dan/atau menggunakan komputer. Kegiatan *unplugged* menjadi pilihan terbaik untuk mendekatkan murid dengan berbagai kegiatan pengembangan keterampilan berpikir komputasional tanpa menuntut kemampuan dan pengalaman yang diperlukan dalam kegiatan *plugged*. Dengan demikian, melalui kegiatan *unplugged*, murid dapat mempelajari konsep-konsep komputasi secara bebas tanpa perlu adanya persiapan khusus.

Sementara itu, metode *plugged* dibutuhkan untuk memaksimalkan pengembangan berpikir komputasional murid. Pada aktivitas *plugged*, ide-ide yang dimiliki murid dapat diuji langsung dengan menggunakan komputer. Kemudian, penggunaan Scratch memungkinkan murid untuk melihat hasil proyek mereka dalam berbagai bentuk, seperti permainan, animasi, kuis, atau musik. Berbagai bentuk karya yang dihasilkan ini dapat menarik perhatian murid untuk terus menggunakan Scratch. Mereka juga dapat membagikan karya mereka di website Scratch sehingga dapat diakses orang lain. Penggunaan Scratch dapat membuat aktivitas pembelajaran menjadi menyenangkan seperti saat murid sedang bermain. Kelebihan-kelebihan Scratch pada metode *plugged* ini sangat sulit untuk didapatkan pada aktivitas *unplugged*.

Berdasarkan kelebihan dan kekurangan dari metode *unplugged* dan metode *plugged*, Caeli & Yadav (2020) berpendapat bahwa pembelajaran informatika perlu dilaksanakan dengan mengombinasikan penggunaan metode *unplugged* dan *plugged*. Implementasi kedua metode ini dapat

saling mendukung satu sama lain, sehingga memberikan pengalaman belajar yang lebih bervariasi bagi para murid.

D. Penutup

Penelitian ini menggunakan statistik inferensial untuk menganalisis data *pretest* dan *posttest* dari kelas kontrol dan eksperimen. Hasil uji *Wilcoxon Signed Ranks* menunjukkan bahwa terdapat perbedaan keterampilan berpikir komputasional murid sebelum dan setelah mengikuti kegiatan pembelajaran dengan metode *unplugged* dan metode *plugged*. Masing-masing metode tersebut memperoleh hasil uji *effect size* sebesar 0,64 dan 0,65 secara berurutan. Nilai ini menekankan bahwa baik metode *unplugged* dan *plugged* memiliki pengaruh yang besar terhadap peningkatan keterampilan berpikir komputasional murid SMP. Selanjutnya, hasil uji *Mann-Whitney* menunjukkan tidak ada perbedaan yang signifikan antara pengaruh metode *unplugged* dan *plugged* terhadap keterampilan berpikir komputasional murid pada setiap komponennya.

Hasil penelitian ini mengindikasikan bahwa guru dapat memilih antara metode *unplugged* atau *plugged* yang dipandang lebih sesuai dengan ketersediaan fasilitas di sekolah atau aktivitas pembelajaran yang diinginkan. Metode *plugged* dapat diterapkan di sekolah yang memiliki sarana dan prasarana yang memadai, seperti komputer, tetapi metode *unplugged* dapat diterapkan di sekolah yang tidak memilikinya. Penggunaan kedua metode ini secara bergantian juga dapat menjadi pilihan baik bagi guru dalam mengembangkan keterampilan berpikir komputasional murid.

Penelitian ini memiliki beberapa keterbatasan yang perlu disampaikan. Pertama, durasi pembelajaran Informatika yang diteliti hanya selama 4 pertemuan. Kedua, instrumen tes yang disusun belum menggunakan elemen evaluasi dari keterampilan berpikir komputasional. Ketiga, penelitian ini hanya mengkaji aspek keterampilan berpikir komputasional dan tidak menggali aspek motivasi murid selama mengikuti kegiatan pembelajaran. Studi berikutnya dapat melakukan kajian yang lebih

mendalam dengan mempertimbangkan beberapa keterbatasan pada penelitian ini. Selain itu, penelitian selanjutnya dapat membahas pengaruh pengombinasian antara metode unplugged dan metode plugged terhadap keterampilan berpikir komputasional murid. Peneliti juga dapat memfokuskan kajian pada beberapa aspek komponen berpikir komputasional sehingga dapat menghasilkan pembahasan yang mendalam.

Ucapan Terima Kasih

Kami mengucapkan terima kasih kepada murid yang berpartisipasi dalam penelitian ini, serta kepada bapak kepala sekolah yang memberikan dukungan dalam setiap langkahnya.

Daftar Referensi

- Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. *Proceedings of the 2012 Annual Meeting of the American Educational Research Association*, 1, 25.
- Caeli, E. N., & Yadav, A. (2020). Unplugged Approaches to Computational Thinking: a Historical Perspective. *TechTrends*, 64(1), 29–36. <https://doi.org/10.1007/s11528-019-00410-5>
- Cárdenas-Cobo, J., Puris, A., Novoa-Hernández, P., Parra-Jiménez, Á., Moreno-León, J., & Benavides, D. (2021). Using scratch to improve learning programming in college students: A positive experience from a non-weird country. *Electronics (Switzerland)*, 10(10), 1–15. <https://doi.org/10.3390/electronics10101180>
- Chen, P., Yang, D., Metwally, A. H. S., Lavonen, J., & Wang, X. (2023). Fostering computational thinking through unplugged activities: A systematic literature review and meta-analysis. *International Journal of STEM Education*, 10(1). <https://doi.org/10.1186/s40594-023-00434-7>
- Chongo, S., Osman, K., & Nayan, N. A. (2021). Impact of the Plugged-in and Unplugged Chemistry Computational Thinking Modules on Achievement in Chemistry. *Eurasia Journal of Mathematics, Science and Technology Education*, 17(4), 1–21. <https://doi.org/10.29333/ejmste/10789>
- Cohen, J. (1992). A Power Primer *Psychol Bull* 112:155-159. *Psychological*

Bulletin, 112(1), 155–159.

- del Olmo-Muñoz, J., Cózar-Gutiérrez, R., & González-Calero, J. A. (2020). Computational thinking through unplugged activities in early years of Primary Education. *Computers and Education*, 150(September 2019). <https://doi.org/10.1016/j.compedu.2020.103832>
- Delal, H., & Oner, D. (2020). Developing middle school students' computational thinking skills using unplugged computing activities. *Informatics in Education*, 19(1), 1–13. <https://doi.org/10.15388/INFEDU.2020.01>
- Dong, W., Li, Y., Sun, L., & Liu, Y. (2024). Developing pre-service teachers' computational thinking: a systematic literature review. In *International Journal of Technology and Design Education* (Vol. 34, Issue 1). Springer Netherlands. <https://doi.org/10.1007/s10798-023-09811-3>
- Dúo-Terrón, P. (2023). Analysis of Scratch Software in Scientific Production for 20 Years: Programming in Education to Develop Computational Thinking and STEAM Disciplines. *Education Sciences*, 13(4). <https://doi.org/10.3390/educsci13040404>
- Erümit, A. K., & Sahin, G. (2020). Plugged or Unplugged Teaching: A Case Study of Students' Preferences in the Teaching of Programming. *International Journal of Computer Science Education in Schools*, 4(1), 3–32. <https://doi.org/10.21585/ijcses.v4i1.82>
- Fagerlund, J., Häkkinen, P., Vesisenaho, M., & Viiri, J. (2021). Computational thinking in programming with Scratch in primary schools: A systematic review. *Computer Applications in Engineering Education*, 29(1), 12–28. <https://doi.org/10.1002/cae.22255>
- Herawati, E. (2022). Penerapan STEAM Project Based Learning “Taman Mini Imun Booster Digital Berbasis QR Code Berbantuan 4DFrame.” *Jurnal Didaktika Pendidikan Dasar*, 6(3), 1139–1156. <https://doi.org/10.26811/didaktika.v6i3.679>
- Hermans, F., & Aivaloglou, E. (2017). A controlled experiment comparing plugged first and unplugged first programming lessons. *ACM International Conference Proceeding Series*, 49–56. <https://doi.org/10.1145/3137065.3137072>
- Huang, W., & Looi, C. K. (2020). A critical review of literature on “unplugged” pedagogies in K-12 computer science and computational thinking education. *Computer Science Education*, 00(00), 1–29. <https://doi.org/10.1080/08993408.2020.1789411>

- Kotsopoulos, D., Floyd, L., Khan, S., Namukasa, I. K., Somanath, S., Weber, J., & Yiu, C. (2017). A Pedagogical Framework for Computational Thinking. *Digital Experiences in Mathematics Education*, 3(2), 154–171. <https://doi.org/10.1007/s40751-017-0031-2>
- Li, Y., Schoenfeld, A. H., diSessa, A. A., Graesser, A. C., Benson, L. C., English, L. D., & Duschl, R. A. (2020). Computational Thinking Is More about Thinking than Computing. *Journal for STEM Education Research*, 3(1), 1–18. <https://doi.org/10.1007/s41979-020-00030-2>
- Memolo, T. (2022). Pembelajaran Matematika Berpikir Komputasi Materi Pola Bilangan dengan Media Kalkulator Web Berbasis Javascript. *Jurnal Didaktika Pendidikan Dasar*, 6(3), 815–826. <https://doi.org/10.26811/didaktika.v6i3.854>
- Merino-Armero, J. M., González-Calero, J. A., Cózar-Gutiérrez, R., & Del Olmo-Muñoz, J. (2022). Unplugged Activities in Cross-Curricular Teaching: Effect on Sixth Graders' Computational Thinking and Learning Outcomes. *Multimodal Technologies and Interaction*, 6(2). <https://doi.org/10.3390/mti6020013>
- Miller, B., Kirn, A., Anderson, M., Major, J. C., Feil-Seifer, D., & Jurkiewicz, M. (2018). Unplugged Robotics to Increase K-12 Students' Engineering Interest and Attitudes. *Proceedings - Frontiers in Education Conference, FIE, 2018-Octob*, 1–5. <https://doi.org/10.1109/FIE.2018.8658959>
- Molina-Ayuso, Á., Adamuz-Povedano, N., Bracho-López, R., & Torralbo-Rodríguez, M. (2022). Introduction to Computational Thinking with Scratch for Teacher Training for Spanish Primary School Teachers in Mathematics. *Education Sciences*, 12(12). <https://doi.org/10.3390/educsci12120899>
- Nasiba, U. (2022). Brankas Rahasia: Media Pembelajaran Numerasi Berbasis Berpikir Komputasi untuk Meningkatkan Kemampuan Pemecahan Masalah. *Jurnal Didaktika Pendidikan Dasar*, 6(2), 521–538. <https://doi.org/10.26811/didaktika.v6i2.764>
- Osman, K., & Lay, A. N. (2022). MyKimDG module: an interactive platform towards development of twenty-first century skills and improvement of students' knowledge in chemistry. *Interactive Learning Environments*, 30(8), 1461–1474. <https://doi.org/10.1080/10494820.2020.1729208>
- Pallant, J. (2016). *SPSS Survival Manual - 6th Edition 2017.pdf*. Open University

Press. <https://doi.org/https://doi.org/10.4324/9781003117452>.

- Pérez-Marín, D., Hijón-Neira, R., Bacelo, A., & Pizarro, C. (2020). Can computational thinking be improved by using a methodology based on metaphors and scratch to teach computer programming to children? *Computers in Human Behavior*, 105. <https://doi.org/10.1016/j.chb.2018.12.027>
- Piedade, J., & Dorotea, N. (2023). Effects of Scratch-Based Activities on 4th-Grade Students' Computational Thinking Skills. *Informatics in Education*, 22(3), 499–523. <https://doi.org/10.15388/infedu.2023.19>
- Rodríguez-Martínez, J. A., González-Calero, J. A., & Sáez-López, J. M. (2020). Computational thinking and mathematics using Scratch: an experiment with sixth-grade students. *Interactive Learning Environments*, 28(3), 316–327. <https://doi.org/10.1080/10494820.2019.1612448>
- Saxena, A., Lo, C. K., Hew, K. F., & Wong, G. K. W. (2020). Designing Unplugged and Plugged Activities to Cultivate Computational Thinking: An Exploratory Study in Early Childhood Education. *Asia-Pacific Education Researcher*, 29(1), 55–66. <https://doi.org/10.1007/s40299-019-00478-w>
- Selby, C., & Woollard, J. (2013). Computational Thinking: The Developing Definition. *ITiCSE Conference 2013*, 5–8.
- Sigayret, K., Tricot, A., & Blanc, N. (2022). Unplugged or plugged-in programming learning: A comparative experimental study. In *Computers and Education* (Vol. 184). <https://doi.org/10.1016/j.compedu.2022.104505>
- Sugiyono. (2017). *Metode penelitian pendidikan:(pendekatan kuantitatif, kualitatif dan R & D)*. Alfabeta.
- Suhardi, R. M., & Utama, G. Y. (2022). The Effect of Implementing STEAM and 4Dframe Learning in Developing Students' Computational Thinking Skills. *Southeast Asian Mathematics Education Journal*, 12(1), 51–64. <https://doi.org/10.46517/seamej.v12i1.138>
- Threekunprapa, A., & Yasri, P. (2020). Unplugged coding using flowblocks for promoting computational thinking and programming among secondary school students. *International Journal of Instruction*, 13(3), 207–222. <https://doi.org/10.29333/iji.2020.13314a>
- Tonbuloglu, B., & Tonbuloglu, I. (2019). The effect of unplugged coding

- activities on computational thinking skills of middle school students. *Informatics in Education*, 18(2), 403–426. <https://doi.org/10.15388/infedu.2019.19>
- Twigg, S., Blair, L., & Winter, E. (2019). Using children's literature to introduce computing principles and concepts in primary schools: Work in progress. *ACM International Conference Proceeding Series*, 3–6. <https://doi.org/10.1145/3361721.3362116>
- Voon, X. P., Wong, S. L., Wong, L. H., Khambari, M. N. M., & Syed-Abdullah, S. I. S. (2023). Developing pre-service teachers' computational thinking through experiential learning: hybridisation of plugged and unplugged approaches. *Research and Practice in Technology Enhanced Learning*, 18. <https://doi.org/10.58459/rptel.2023.18006>
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35. <https://doi.org/10.1145/1118178.1118215>
- Wing, J. M. (2011). Research notebook: Computational thinking—What and why? *The Link Magazine*, 6, 20–23.
- Zapata-Cáceres, M., Marcelino, P., El-Hamamsy, L., & Martín-Barroso, E. (2024). A Bebras Computational Thinking (ABC-Thinking) program for primary school: Evaluation using the competent computational thinking test. *Education and Information Technologies*, 0123456789. <https://doi.org/10.1007/s10639-023-12441-w>
- Zhang, L. C., & Nouri, J. (2019). A systematic review of learning computational thinking through Scratch in K-9. *Computers and Education*, 141, 103607. <https://doi.org/10.1016/j.compedu.2019.103607>