

PENGARUH PENERAPAN ALGORITMA TERHADAP PEMBELAJARAN PEMROGRAMAN KOMPUTER

Allen Marga Retta¹, Asnurul Isroqmi², Tika Dwi Nopriyanti³

Pendidikan Matematika, Universitas PGRI Palembang^{1,2,3}

allenmargaretta1@gmail.com¹

ABSTRAK

Perkembangan teknologi informasi dan komputer saat ini semakin maju, semua bidang ilmu memerlukan keahlian komputer, sehingga bagi mahasiswa pemahaman logika matematika dan pembelajaran komputer sangat penting karena keduanya saling berhubungan. Berdasarkan pengalaman pembelajaran matakuliah komputer pemrograman masih banyak mahasiswa yang belum menguasai matakuliah tersebut. Untuk mengatasi hal tersebut, dilakukan penelitian dengan menerapkan algoritma dalam pembelajaran pemrograman komputer. Penelitian ini merupakan penelitian deskriptif kuantitatif yang bertujuan untuk melihat pengaruh penerapan algoritma terhadap hasil belajar mahasiswa dalam pembelajaran pemrograman komputer. Sampel penelitian yaitu mahasiswa Universitas PGRI Palembang tahun ajar 2017/2018 yang terdiri dari dua kelas (kelas eksperimen dan kelas kontrol). Setelah dilakukan penelitian diperoleh hasil bahwa: 1) Hasil belajar mahasiswa setelah diterapkan algoritma dalam setiap penyelesaian pemrograman lebih baik daripada mahasiswa yang memperoleh pembelajaran konvensional ditinjau dari keseluruhan mahasiswa; 2) Hasil belajar mahasiswa setelah diterapkan algoritma dalam setiap penyelesaian pemrograman lebih baik daripada mahasiswa yang memperoleh pembelajaran konvensional ditinjau dari KAM (tinggi, sedang dan rendah).

Kata kunci : algoritma, pemrograman komputer

ABSTRACT

The development of information technology and computers nowadays is more advanced. All fields of science require computer expertise. Understanding mathematical logic and computer learning is very important for students because the two are interconnected. Based on the experience of lectures in a computer programming course, there are still many students who have not mastered this course. To overcome this problem, research is carried out by applying algorithms in learning computer programming. This research is a quantitative descriptive study that aims to see the effect of applying the algorithm to student learning outcomes in learning computer programming. The research sample were students of the Universitas PGRI Palembang in the academic year 2017/2018 consisting of two classes (experimental class and control class). After the research, the results are obtained that 1) Student learning outcomes after applying the algorithm in each programming completion is better than students who obtain conventional learning in terms of overall students; 2) Student learning outcomes after applying the algorithm in each completion of programming are better than students who obtain

conventional learning in terms of students' initial ability (high, medium and low).

Keywords : algorithm, computer programming

PENDAHULUAN

Perubahan kemajuan teknologi di setiap bidang kehidupan terjadi sangat pesat dalam mengakses pengetahuan dan komunikasi satu sama lain. Dengan menggunakan komputer sebagai salah satu media dalam mengakses pengetahuan dan komunikasi, perlu adanya keahlian komputer yang memadai. Pembelajaran komputer sangat penting diikuti terutama untuk mahasiswa pendidikan matematika yang merupakan pembelajaran yang erat kaitannya dengan kemampuan berpikir logis mahasiswa.

Menurut Munir (2016), kontribusi Matematika dalam Teknologi Informasi dan Komputer adalah Aljabar Boolean, Algoritma, Teori Informasi, Logika Simbolik, Peluang, Kalkulus, Statistika. Keterkaitan antara matematika dan komputer ada dalam dua hal yaitu: 1) Matematika dapat mencari persamaan logika yang rasional yang dapat diterjemahkan ke dalam komputer melalui bahasa pemrograman, 2) Komputer dapat melakukan perhitungan logika rasional matematis secara cepat dan tepat. Keterbatasan komputer dapat diatasi dengan logika matematis, sedangkan persoalan matematis dapat dikomputerisasikan layaknya menghitung banyaknya pasir dalam timbangan.

Program studi pendidikan matematika memiliki mata kuliah yang mempelajari tentang komputer yaitu Algoritma dan Pemrograman. Mata kuliah ini bertujuan untuk memberikan pemahaman tentang konsep pemrograman komputer yang meliputi pembuatan algoritma dan konsep-konsep dasar pemrograman seperti: data dan variabel, ekspresi dan operator, struktur pengambilan keputusan, struktur pengulangan, fungsi, dan array, operasi string dan matematika. Konsep-konsep pemrograman dituangkan ke dalam bahasa pemrograman yang digunakan saat praktek yaitu bahasa pemrograman Just BASIC. Bahasa pemrograman menurut Wahyudi dkk (2013) merupakan suatu himpunan dari aturan sintaks dan semantik yang dipakai untuk mendefinisikan program komputer. Dalam pembelajaran pemrograman komputer, algoritma juga memiliki peranan penting dalam menyelesaikan suatu masalah. Menurut Maulana (2017) algoritma merupakan kumpulan perintah untuk menyelesaikan suatu masalah dimana masalah tersebut diselesaikan dituntut secara sistematis, terstruktur dan logis. Begitu juga dengan pendapat Saniman dan Fathoni (2008) yang menyatakan bahwa algoritma merupakan susunan langkah-langkah sistematis, dan logis dalam pemecahan suatu masalah.

Berdasarkan hasil pengamatan yang dilakukan saat proses pembelajaran mata kuliah Algoritma dan Pemrograman selama ini, serta wawancara informal dengan mahasiswa tahun ajaran 2015/2016 Prodi Pendidikan Matematika Universitas PGRI

Palembang, seringkali didengar keluhan mahasiswa dan mahasiswa terlihat mengalami kendala saat harus menuliskan ide/pemikiran mereka ke dalam kode atau notasi bahasa pemrograman, walaupun mereka sebetulnya sudah sangat memahami persoalan yang harus mereka pecahkan, dan juga memahami kode/notasi atau sintaks dari bahasa pemrograman yang digunakan. Hal ini terjadi karena mahasiswa belum mampu menuangkan logika pemikirannya ke dalam pemrograman, mereka belum mampu membuat langkah-langkah penyelesaiannya yang diurutkan secara logis. Dalam penelitian Isroqmi (2018) menunjukkan bahwa hasil belajar dan respon mahasiswa setelah diterapkan algoritma dikategorikan baik dengan nilai tes akhir sebesar 82,1 dan respon mahasiswa dikategorikan positif. Ditambah pula dengan hasil penelitian yang sama oleh Isroqmi (2013) yang menunjukkan bahwa hasil belajar mahasiswa setelah diterapkan bahasa *Just Basic* pada pembelajaran pemrograman komputer dari tes akhir diperoleh 82,65 dikategorikan baik dan respon atau sikap mahasiswa terhadap penerapan software bahasa pemrograman *Just BASIC* positif.

Dengan latar belakang inilah, maka perlu dilakukan penelitian dengan tujuan untuk mengetahui: 1) Pengaruh penerapan algoritma terhadap hasil belajar mahasiswa ditinjau dari keseluruhan mahasiswa. 2) Pengaruh penerapan algoritma terhadap hasil belajar mahasiswa ditinjau dari Kemampuan Awal Mahasiswa (tinggi, sedang dan rendah).

Penerapan Algoritma

Menurut Barakbah (2013:1), pengertian algoritma sangat lekat dengan kata logika, yaitu kemampuan seorang manusia untuk berpikir dengan akal tentang suatu permasalahan menghasilkan sebuah kebenaran, dibuktikan dan dapat diterima akal, logika seringkali dihubungkan dengan kecerdasan, seseorang yang mampu berlogika dengan baik sering orang menyebutnya sebagai pribadi yang cerdas. Dalam menyelesaikan suatu masalahpun logika mutlak diperlukan. Logika identik dengan masuk akal dan penalaran.

Pertimbangan dalam penerapan algoritma adalah 1) algoritma haruslah benar, artinya algoritma akan memberikan keluaran yang dikehendaki dari sejumlah masukan yang diberikan. Tidak peduli sebegus apapun algoritma, kalau memberikan keluaran yang salah, pastilah algoritma tersebut bukanlah algoritma yang baik, 2) Algoritma yang baik harus mampu memberikan hasil yang sedekat mungkin dengan nilai yang sebenarnya. Kita harus mengetahui seberapa baik hasil yang dicapai oleh algoritma tersebut. Hal ini penting terutama pada algoritma untuk menyelesaikan masalah yang memerlukan aproksimasi hasil (hasil yang hanya berupa pendekatan), 3) Efisiensi algoritma, semisal algoritma itu benar (mendekati kebenaran), tetapi memakan waktu yang lama dalam mendapatkan kebenaran algoritma, untuk apa algoritma tersebut dipakai? Karena inti dari algoritma yang baik adalah mendapatkan jawaban kebenaran (mendekati kebenaran) dengan cepat.

Suatu Algoritma dapat terdiri dari tiga struktur dasar, yaitu runtunan, pemilihan dan pengulangan (Ridho, 2013:5). Ketiga jenis langkah tersebut membentuk konstruksi suatu algoritma. Dimulai dari sebuah runtunan terdiri dari satu atau lebih instruksi. Tiap instruksi dikerjakan secara berurutan sesuai dengan urutan penulisannya, yakni sebuah instruksi dilaksanakan setelah instruksi sebelumnya selesai dikerjakan. Urutan dari instruksi menentukan hasil akhir dari suatu algoritma. Bila urutan penulisan berubah maka mungkin juga hasil akhirnya berubah. Sebagai contoh perhatikan operasi dari aritmatika berikut ini, $(4+3)*7=49$, tetapi bila urutannya diubah maka hasil keluaran akan berbeda menjadi $4+(3*7)=25$.

Selanjutnya struktur dasar pemilihan yaitu instruksi yang dikerjakan dengan kondisi tertentu. Kondisi adalah persyaratan yang dapat bernilai benar atau salah. Satu atau beberapa instruksinya dilaksanakan apabila kondisi bernilai benar, sebaliknya apabila salah maka instruksi tidak akan dilaksanakan. Misalnya dalam penentuan bilangan genap atau ganjil berikut:

1. Masukkan bilangan sebagai sebuah bilangan bulat.
2. Bagi bilangan dengan angka 2, simpan nilai sisa pembagian dalam variabel sisa.
3. Jika nilai sisa sama dengan 0 maka kerjakan langkah 4.
4. Tampilkan "GENAP" ke layar.
5. Jika nilai sisa tidak sama dengan 0 maka kerjakan langkah 6.
6. Tampilkan "GANJIL" ke layar.
7. Selesai.

Terakhir struktur dasar pengulangan yaitu suatu kegiatan mengerjakan sebuah atau sejumlah aksi yang sama sebanyak jumlah yang ditentukan atau sesuai dengan kondisi yang diinginkan Beberapa pernyataan pengulangan di bahasa pemrograman yaitu *for...*, *while ()...*, *do...while ()*, *repeat...until*, *for...down to...do*, *for...to...do* dan lain-lain. Sebagai contoh adalah menampilkan huruf tertentu sebanyak n kali ke layar sebagai berikut:

1. Deklarasikan variabel **huruf** untuk menyimpan karakter yang akan ditampilkan.
2. Deklarasikan variabel **n** untuk menyimpan banyaknya perulangan.
3. Deklarasikan variabel **counter** yang digunakan sebagai counter perulangan yang sudah dilakukan.
4. Masukkan sebuah karakter dan simpan dalam variabel **huruf**.
5. Masukkan banyaknya perulangan yang diinginkan dan simpan dalam variabel **n**.
6. Set nilai **counter** dengan 0.
7. Tampilkan **huruf** ke layar.
8. Lakukan penambahan **counter** dengan 1.
9. Jika nilai **counter** < **n**, kerjakan langkah 6.
10. Jika **nilai counter** = **n** selesai.

Selain tiga struktur dasar yang telah dijelaskan, terdapat tiga cara umum dalam menuliskan algoritma yaitu dengan kalimat deskriptif, *pseudocode*, dan *flowchart*.

Kalimat deskriptif dilakukan dengan cara menuliskan instruksi-instruksi yang harus dilaksanakan dalam bentuk untaian kalimat deskriptif dengan menggunakan bahasa yang jelas. Bahasa yang biasa digunakan adalah bahasa Inggris, namun dapat dimodifikasi dengan bahasa sehari-hari termasuk Bahasa Indonesia. Hal ini disebabkan oleh tidak ada aturan baku dalam menuliskan algoritma dengan notasi deskriptif maka tiap orang dapat membuat aturan penulisan dan notasi algoritma sendiri. Hal ini dapat dimengerti karena teks algoritma tidak sama dengan teks program.

Selanjutnya dengan *pseudocode* merupakan cara penulisan algoritma yang menyerupai bahasa pemrograman tingkat tinggi. *Pseudocode* menggunakan bahasa yang hampir menyerupai bahasa pemrograman. Biasanya *pseudocode* menggunakan bahasa yang mudah dipahami secara universal dan juga lebih ringkas daripada algoritma. *Pseudocode* berisi deskripsi dari algoritma pemrograman komputer yang menggunakan struktur sederhana dari beberapa bahasa pemrograman tetapi bahasa tersebut hanya ditujukan agar dapat dibaca manusia. Sehingga *Pseudocode* tidak dapat dipahami oleh komputer. Supaya notasi *Pseudocode* bisa dipahami oleh komputer maka harus diterjemahkan terlebih dahulu menjadi sintaks bahasa pemrograman komputer tertentu.

Penulisan algoritma terakhir dengan *flowchart* yaitu cara penulisan algoritma dengan menggunakan notasi grafis. *Flowchart* merupakan gambar atau bagan yang memperlihatkan urutan atau langkah-langkah dari suatu program dan hubungan antar proses beserta pernyataannya.

Pembelajaran Pemrograman Komputer

Pemrograman komputer dapat diartikan sebagai proses menulis, menguji dan memperbaiki (*debug*), dan memelihara kode yang membangun sebuah program komputer dimana kode ini ditulis dalam berbagai bahasa pemrograman. Tujuan dari pembelajaran pemrograman komputer adalah untuk memuat suatu program yang dapat melakukan suatu perhitungan atau “pekerjaan” sesuai dengan keinginan si pemrogram. Untuk dapat melakukan pemrograman, diperlukan keterampilan dalam algoritma, logika, bahasa pemrograman, dan di banyak kasus, pengetahuan-pengetahuan lain seperti matematika.

Purnamasari (2005:2) menyebutkan dalam menentukan standar program yang baik dibutuhkan beberapa standar sebagai dasar penilaian, seperti; teknik pemecahan masalah dan penyusunan program.

Dalam standar teknik pemecahan masalah, teknik *Top-Down* merupakan teknik pemecahan masalah yang paling umum digunakan, dimana suatu masalah yang kompleks dibagi-bagi ke dalam beberapa tingkatan kelompok masalah hingga sub bagian yang paling kecil. Setelah itu kemudian disusun langkah-langkah untuk menyelesaikan secara detail. Langkah-langkah yang disusun secara detail sering kali disebut dengan Algoritma.

Proses dari masalah hingga menjadi suatu algoritma disebut tahap pemecahan masalah, sedangkan tahap dari algoritma hingga menjadi suatu solusi disebut dengan tahap implementasi. Solusi yang dimaksud adalah suatu program yang merupakan implementasi dari algoritma yang disusun.

Selanjutnya standar penyusunan program. Terdapat faktor-faktor yang menjadi standar dalam menyusun program, antara lain: (1) Kebenaran logika dan penulisan; (2) Waktu minimum untuk penulisan program; (3) Kecepatan maksimum eksekusi program; dan (4) Ekspresi Penggunaan Memori.

Menurut Yendri (2013:4), dalam menulis program bukan hanya merupakan suatu bentuk kreativitas tetapi juga merupakan sebuah penerapan disiplin ilmu secara teliti karena program merupakan implementasi dari suatu algoritma.

Pembuat program atau pemrogram yang baik akan menghasilkan program yang mampu memberikan solusi yang tepat dan benar. Pemrogram dapat digolongkan dalam dua jenjang, yaitu amatir dan profesional. Pemrogram amatir biasanya hanya menghasilkan program untuk memecahkan masalah-masalah yang relatif kecil, sedangkan pemrogram profesional memiliki kemampuan untuk menyusun program untuk memecahkan masalah-masalah yang besar dan rumit

METODE

Penelitian ini merupakan penelitian *Quasi-Experimental*. Perlakuan diberikan kepada mahasiswa Program Studi Pendidikan Matematika yang mengikuti perkuliahan algoritma pemrograman. Populasi pada penelitian ini adalah mahasiswa Program studi Pendidikan Matematika Universitas PGRI Palembang tahun ajaran 2017/2018, sedangkan subyek penelitian ini terdiri dari dua kelas sampel penelitian yaitu kelas 4A dengan jumlah mahasiswa 41 orang dan kelas 4B dengan jumlah mahasiswa 40 orang. Sebelum memulai penelitian dilakukan terlebih dahulu tes kemampuan awal di kedua kelas sampel untuk kemudian dianalisis dengan uji normalitas dan homogenitas. Hal ini dilakukan untuk melihat kesetaraan kemampuan kedua kelas sebelum dilakukan uji statistik. Setelah uji statistik dilakukan dan disimpulkan bahwa kedua kelas memiliki kemampuan yang sama, lalu ditentukan kelas eksperimen dan kelas kontrol secara acak.

HASIL DAN PEMBAHASAN

Pertemuan pertama penelitian ini diawali dengan pemberian soal tes kemampuan awal mahasiswa terdiri dari tujuh soal yang sudah divalidasi oleh dua pakar sebelumnya. Berikut disajikan tabel 1 hasil tes Kemampuan Awal Mahasiswa (KAM):

Tabel 1. Statistik deskriptif data tes KAM berdasarkan kelas sampel penelitian

Kelas Sampel Penelitian	Skor		Rerata	Simpangan Baku	N
	Min	Maks			
A	1	7	4,05	1,7	41
B	1	7	4,3	1,63	40

Berdasarkan tabel 1 di atas terlihat bahwa rerata dan simpangan baku antara kelas A dan kelas B tidak jauh berbeda. Dengan selisih rerata kelas A dengan kelas B hanya 0,25 dan selisih simpangan baku antara kelas A dan B hanya 0,07. Berikut adalah tabel 2 hasil analisis data tes.

Tabel 2. Statistik deskriptif data tes

Kelompok KAM	Penerapan Algoritma				Tidak Menerapkan Algoritma			
	N	Stat	KAM	Hasil belajar	N	Stat	KAM	Hasil belajar
Keseluruhan	41	\bar{x}	4,05	8,3	40	\bar{x}	4,3	7,23
		S	1,7	1,2		S	1,63	1,52
Tinggi	10	\bar{x}	6,3	9,75	10	\bar{x}	6,4	7,51
		S	0,47	0,2		S	0,52	1,63
Sedang	22	\bar{x}	3,95	7,78	23	\bar{x}	4,08	7,07
		S	0,8	1,02		S	0,79	1,62
Rendah	9	\bar{x}	1,56	8,07	7	\bar{x}	1,86	7,31
		S	0,53	0,84		S	0,38	0,95

Berdasarkan tabel 2, terlihat bahwa rerata-rata tes secara keseluruhan maupun berdasarkan tingkat kemampuan (tinggi, sedang, dan rendah) skor mahasiswa di kelas eksperimen atau yang menerapkan algoritma lebih besar dibandingkan dengan kelas kontrol. Hal ini terlihat dari tabel bahwa rerata kelas yang diterapkan algoritma secara keseluruhan sebesar 8,3, sedangkan rerata kelas kontrol yang mahasiswanya tidak diterapkan algoritma dalam pembelajaran hanya memperoleh skor sebesar 7,23. Begitu juga untuk rerata kemampuan tinggi, kelas eksperimen sebesar 9,75 dan kelas kontrol sebesar 7,51.

Sebelum dilakukan uji statistik, hasil belajar diuji normalitas data dan uji homogenitas data. Berikut disajikan tabel 3 uji normalitas untuk data hasil belajar:

Tabel 3. Hasil uji normalitas data tes akhir

Kelompok KAM	Pembelajaran	N	K-S	Sig	H ₀
Keseluruhan	PA	41	0,103	0,2	Diterima
	PK	40	0,121	0,145	Diterima
Tinggi	PA	10	0,46	0,00	Ditolak
	PK	10	0,166	0,2	Diterima
Sedang	PA	22	0,137	0,2	Diterima
	PK	23	0,142	0,2	Diterima
Rendah	PA	9	0,165	0,2	Diterima
	PK	7	0,168	0,2	Diterima

Rumusan hipotesis untuk uji normalitas:

- H_0 : sampel berasal dari populasi yang berdistribusi normal
 H_1 : sampel berasal dari populasi yang tidak berdistribusi normal

Berdasarkan tabel 3 diperoleh probabilitas untuk kelas yang diterapkan algoritma sebesar 0,2 dan kelas yang tidak diberikan perlakuan nilai probabilitas sebesar 0,145 yang artinya kedua kelas tersebut nilai probabilitas lebih besar daripada taraf signifikansi (*sign*) $\alpha = 0,05$ maka H_0 diterima atau sampel berasal dari populasi yang berdistribusi normal.

Setelah dilakukan uji Normalitas, selanjutnya data dihitung dan analisis dengan Uji Homogenitas. Berikut disajikan tabel 4 Uji Homogenitas varians:

Tabel 4. Hasil uji homogenitas varians

		Levene Statistic	df1	df2	Sig.
NILAI_UAS	Based on Mean	2.608	1	79	.110
	Based on Median	2.013	1	79	.160
	Based on Median and with adjusted df	2.013	1	73.596	.160
	Based on trimmed mean	2.537	1	79	.115

Pada tabel Uji Homogenitas menunjukkan bahwa nilai *sign* 0,110 dan lebih besar daripada taraf signifikansi $\alpha = 0,05$ yang berarti H_0 diterima. Artinya secara keseluruhan kedua kelompok data bervarians homogen.

Dari data penelitian, diperoleh hasil uji t yang disajikan pada tabel 5 berikut:

Tabel 5. Hasil uji perbedaan rerata

Kelompok KAM	Pembelajaran	T	Sig	H_0
Keseluruhan	PA	3,417	0,001	Ditolak
	PK			
Tinggi	PA	4,192	0,001	Ditolak
	PK			
Sedang	PA	1,401	0,168	Diterima
	PK			
Rendah	PA	1,577	0,137	Diterima
	PK			

Berdasarkan tabel 5 dapat disimpulkan bahwa hasil analisis uji t perbedaan rerata kemampuan berpikir logis mahasiswa yang mendapat penerapan algoritma tidak lebih baik daripada mahasiswa yang mendapat pembelajaran konvensional ditinjau dari keseluruhan. Akan tetapi jika ditinjau dari rerata yang terlihat pada tabel 3 hasil belajar kelas yang diberikan penerapan algoritma sebesar 8,3 sedangkan rerata kelas kontrol sebesar 7,23 hal ini menunjukkan bahwa kelas yang menerapkan algoritma lebih tinggi. Berdasarkan hasil analisis uji t perbedaan rerata mahasiswa pada tabel 5

yang mendapat penerapan algoritma tidak lebih baik daripada mahasiswa yang mendapat pembelajaran konvensional ditinjau dari KAM tinggi. Akan tetapi rerata kelas yang terlihat pada tabel 3 yang menerapkan algoritma 9,75 jauh lebih tinggi dibandingkan dengan kelas kontrol yang rerata 7,51. Hal ini menunjukkan bahwa rerata yang menerapkan algoritma lebih baik daripada mahasiswa yang mendapatkan pembelajaran konvensional. Berdasarkan hasil analisis uji t perbedaan rerata mahasiswa yang mendapat penerapan algoritma lebih baik daripada mahasiswa yang mendapat pembelajaran konvensional ditinjau dari KAM sedang dan rendah.

SIMPULAN DAN SARAN

Berdasarkan penelitian yang dilakukan maka dapat disimpulkan bahwa 1) Hasil belajar mahasiswa setelah diterapkan algoritma dalam setiap penyelesaian pemrograman lebih baik daripada mahasiswa yang memperoleh pembelajaran konvensional ditinjau dari keseluruhan mahasiswa. 2) Hasil belajar mahasiswa setelah diterapkan algoritma dalam setiap penyelesaian pemrograman lebih baik daripada mahasiswa yang memperoleh pembelajaran konvensional ditinjau dari KAM (tinggi, sedang dan rendah).

Adapun saran dari hasil penelitian ini adalah pendidik dan calon pendidik dalam mengajar hendaknya memperhatikan self esteem siswa karena berpengaruh pada kemampuan komunikasi matematis siswa.

DAFTAR PUSTAKA

- Barakbah, Ali Ridho, dkk. 2013. *Logika dan Algoritma*. Program Studi Teknik Informatika Departemen Teknik Informatika dan Komputer Politeknik Elektronika Negeri Surabaya.
- Isroqmi, Asnurul. 2013. Penerapan Bahasa Just Basic Pada Pembelajaran Pemrograman Komputer di Program Studi Pendidikan Matematika Universitas PGRI Palembang. *Jurnal Ripteksi Kependidikan PGRI*, Vol. I(1) : 1-10.
- Isroqmi, Asnurul. 2018. Kemampuan Mahasiswa Memahami Logika Pemrograman Komputer Melalui Algoritma. *Nabla Dewantara: Jurnal Pendidikan Matematika*, Vol. 2(2) : 59-74.
- Maulana, Gun Gun. 2017. Pembelajaran Dasar Algoritma Dan Pemrograman Menggunakan El-Goritma Berbasis Web. *Jurnal Teknik Mesin (JTM)*, Vol. 06, Edisi Spesial.
- Munir. 2016. *Kontribusi Matematika dalam TIK*. [Online]. Tersedia : http://file.upi.edu/Direktori/FPMIPA/PRODI_ILMU_KOMPUTER/196603252001121-MUNIR/Presentasi_TIK/Kontribusi_Matematika_dalam_TIK.pdf. [22 September 2016].
- Purnamasari, Detty. 2005. *Materi Pengajaran: Algoritma dan Pemrograman*. [Online]. Tersedia : <http://detty.staff.gunadarma.ac.id/Downloads/files/4659/Pendahulu-an.pdf>. [21 September 2016].

- Ridho, dkk. 2013. *Logika dan Algoritma*. [Online]. Tersedia: <http://entin.lecturer.pens.ac.id/Logika%20Algoritma/Buku%20Logika%20Algoritma.pdf>. [23 Mei 2017].
- Saniman dan Fathoni, M. 2008. Pengantar Algoritma dan Pemrograman. *Jurnal SAINTIKOM*, Vol. 4(1) : 120-133. [Online]. Tersedia : <https://prpm.trigunadharna.ac.id/public/fileJurnal/volArsip/247AKATA%20PENGANTAR-4-1.pdf>
- Wahyudi, J. dan Berlian, R. 2013. Instruksi Bahasa Pemrograman ADT (Abstract Data Type) Pada Virus dan Loop Batch. *Jurnal Media Infotama*, Vol.9(2) : 64-77. [Online]. Tersedia : <https://jurnal.unived.ac.id/index.php/jmi/issue/view/4>
- Yendri, Dodon. 2013. *Bahan Ajar Algoritma dan Pemrograman I*. [Online]. Tersedia pada: http://fti.unand.ac.id/images/MATERIKULIAH/DODONYENDRI/3_Pemrograman.pdf. [23 September 2016].