

Peningkatan Programming Skill Melalui Pendekatan *Hemispheric Cognitive Style Collaboration (HCSC)* Pada Mata Pelajaran PBO Siswa Kelas XI RPL

Enrico Pariu
PBO,SMKN 2 Poso Pesisir
ikopariur@gmail.com

Abstrak - Berdasarkan pengamatan penulis yang berprofesi sebagai guru terhadap kemampuan siswa memahami alur pemrograman berorientasi object dan minat belajar siswa yang sangat rendah. Pada kegiatan praktikum pemrograman berorientasi objek siswa cenderung hanya mengcopy paste contoh program yang dibuat oleh guru, sehingga ketika dibawa ke kasus atau contoh soal yang berbeda siswa kesulitan mengeluarkan kemampuan nalarnya untuk memecahkan masalah tersebut secara mandiri. Adapun tujuan penelitian ini untuk meningkatkan kemampuan keterampilan siswa membuat program. Model pembelajaran yang tepat sangat diharapkan untuk keefektifan pembelajaran. Metode HCSC berusaha mengoptimalkan fungsi kognitif belahan otak kiri dan kanan dalam mempelajari konsep OOP yang membutuhkan kemampuan kognitif seperti abstraksi, analisis dan pemecahan masalah. Tujuan dari penelitian ini adalah ingin mengkaji metode pembelajaran HCSC dalam meningkatkan keterampilan siswa membuat program pada mata pelajaran PBO Kelas XI RPL SMKNegeri 2 Poso Pesisir. Hasil analisis didapatkan bahwa prestasi belajar siswa mengalami peningkatan dari pra siklus sampai siklus II yaitu, pra siklus (38,46%), siklus I (69,23%) dan pada siklus II (84,61%).

Kata Kunci : pemrograman berorientasi objek, metode HCSC

I. PENDAHULUAN

Dalam riset dibidang ilmu pengetahuan komputer umumnya ditemukan bahwa siswa mengalami kesulitan dalam mempraktekan pemikiran yang abstrak. Fenomena umum yang ditemukan disekolah jurusan komputer adalah kecenderungan siswa untuk fokus langsung pada kode ketimbang disain ketika menulis atau membaca program. Terlebih lagi studi yang belum lama ini dilakukan (I. Hadar 2004) menemukan bahwa proses berpikir (cognitive) juga mempengaruhi dalam desain OOP diujicobakan kepada pengembang perangkat lunak berpengalaman dan berhasil mengidentifikasi fenomena yang sama yaitu kesulitan dalam berpikir secara abstrak (I. Hadar 2007).

Kegiatan pembelajaran merupakan proses pendidikan yang memberikan kesempatan kepada siswa untuk mengembangkan potensi mereka menjadi kemampuan yang semakin lama semakin meningkat dalam sikap, pengetahuan, dan keterampilan yang diperlukan dirinya untuk hidup dan untuk bermasyarakat, berbangsa, serta berkontribusi pada kesejahteraan hidup umat manusia. Oleh karena itu, kegiatan pembelajaran diarahkan untuk memberdayakan semua potensi siswa menjadi kompetensi yang diharapkan. Demikian pula strategi pembelajaran yang akan digunakan harus diarahkan untuk memfasilitasi pencapaian kompetensi yang telah dirancang dalam dokumen kurikulum agar setiap individu mampu menjadi pebelajar mandiri sepanjang hayat dan yang pada gilirannya mereka menjadi komponen penting untuk mewujudkan masyarakat belajar.

Keberhasilan belajar bukan hanya bergantung pada lingkungan atau kondisi belajar dalam hal ini model pembelajaran yang digunakan melainkan juga pada

pengetahuan awal siswa (Suherman, 2003). Vygotsky (dalam Taylor, 1993) menyatakan bahwa siswa dalam mengkonstruksi suatu konsep perlu memperhatikan lingkungan sosial. Jonassen (2010) mengatakan kebanyakan ahli psikologi dan pendidik menyatakan bahwa penyelesaian masalah sebagai hasil pembelajaran sangat penting untuk kehidupan, karena hampir semua orang dalam kehidupan sehari-hari harus berpikir kreatif untuk menyelesaikan masalah.

Perkembangan bahasa pemrograman semakin meningkat seiring dengan kebutuhan pengembangan sistem yang semakin kompleks. Teknik terstruktur dan prosedural dalam pemrograman dirasakan kurang relevan lagi karena tools untuk analisis dan desain sekarang ini sudah berbasis objek.

Object Oriented Programing (OOP) adalah teknik pemrograman yang sudah berkembang cukup lama Namun industri menemukan bahwa program mengalami kesulitan untuk beralih ke paradigma OOP (White and Sivitanides 2005). Ketika belajar pemrograman berorientasi objek, bahasa ini berisi lebih banyak konsep daripada bahasa structural atau procedural, yang mengakibatkan mempelajari bahasa ini menjadi lebih sulit. Akan tetapi bahasa berorientasi objek memasukan konsep class, yang diharapkan memudahkan dalam mendesain dan menstrukturkan program. (Kaasboll, et al. 2003). Mereka cenderung untuk berkuat pada level terendah dari abstraksi untuk memecahkan masalah. Cakap dalam hal detail tetapi tidak memahami gambaran besarnya (Aharoni 1997).

Permasalahan dalam proses belajar ini perlu segera diatasi. Salah satu upaya yang dilakukan adalah menggunakan pendekatan HCSC (*Hemispheric Cognitive Style Collaboration*) pada siswa Kelas XI RPL SMKN 2 Poso Pesisir pada mata pelajaran PBO (Pemrograman

Berorientasi Objek) semester I tahun pelajaran 2019 / 2020. Dengan pembelajaran tersebut diharapkan siswa lebih aktif dan tertarik mengikuti proses pembelajaran.

II. METODE PENELITIAN

Desain penelitian yang digunakan adalah Penelitian Tindakan Kelas (PTK) yang bertujuan untuk meningkatkan hasil belajar melalui pendekatan HCSC pada mata pelajaran Pemrograman Berorientasi Objek, siswa kelas XI RPL semester 1 SMK Negeri 2 Poso Pesisir. PTK adalah bagaimana guru dapat mengorganisasi kondisi praktek pembelajaran dan belajar dari pengalamannya sendiri, dapat mencobakan gagasan perbaikan dalam praktek pembelajaran mereka, dan melihat pengaruh nyata dari upaya itu (Wiraatmaja, 2006). Subjek penelitian adalah siswa kelas XI RPL semester 1 SMK Negeri 2 Poso Pesisir yang berjumlah 13 siswa. Objek dalam penelitian ini adalah pelaksanaan hasil yang diperoleh dari pembelajaran melalui model pendekatan HCSC pada mata pelajaran Pemrograman Berorientasi Objek siswa kelas XI RPL semester 1 SMK Negeri 2 Poso Pesisir Tahun Pelajaran 2019/2020. Waktu penelitian dimulai dari tahap pra survei hingga dilaksanakannya tindakan adalah 4 pekan. Sebagai tahap awal, dilakukan pra survei pada pekan pertama bulan November 2019 dengan melakukan tes awal materi *Class* dan *Object* XI RPL SMK Negeri 2 Poso Pesisir. Kemudian dilanjutkan dengan pemberian tindakan. Dan penelitian berakhir pada pekan keempat bulan November 2019.

Sebelum mengadakan tindakan penelitian, peneliti mengadakan observasi serta mencari data kemampuan awal penguasaan materi *Class* dan *Object* dari siswa. Model penelitian yang digunakan adalah model kemmis dan taggart, yaitu model spiral, dimana dalam model spiral ini terdiri dari 2 siklus dan dari setiap siklus terdiri dari perencanaan, pelaksanaan tindakan, observasi serta refleksi. Walaupun pada penelitian tindakan kelas ini dilakukan dalam 2 siklus namun bila ternyata dari dua siklus yang direncanakan masih terdapat masalah yang harus dipecahkan maka dapat dilanjutkan dengan siklus berikutnya. Pengumpulan data yang dilakukan dalam penelitian dengan metode Observasi, Tes dan Dokumentasi menggunakan instrument penelitian berupa Lembar Observasi (pengamatan) dan Tes Ketrampilan Pemrograman.

III. HASIL DAN PEMBAHASAN

Hasil Penelitian

1. Pra Tindakan Kelas

Penelitian dilaksanakan di SMK Negeri 2 Poso Pesisir. Sebelum melakukan penelitian, peneliti melakukan persiapan-persiapan yang berkaitan dengan pelaksanaan penelitian. Salah satunya adalah tempat dan waktu penelitian. Pemberian pretest kemampuan penalaran pada tanggal 1 November 2019 dan perlakuan diberikan selama 6 kali pertemuan dan terbagi dalam 2 siklus. Siklus I dilaksanakan pada tanggal 4 - 14 November 2019,

sedangkan siklus II dilaksanakan pada tanggal 20 - 30 November 2019.

Sebelum pelaksanaan tindakan kelas, peneliti memberikan penjelasan bahwa kegiatan pembelajaran yang akan dilakukan adalah kegiatan pembelajaran menggunakan pendekatan HCSC, yang tahapannya terdiri atas *Programming Model (PM)*, *High Level Abstraction (HLA)*, *Middle Level Abstraction (MLA)*, *Detail Implementation (DI)*.

Hasil tes ketrampilan pemrograman awal siswa kelas XI RPL sebelum adanya tindakan berdasarkan pretest yang diberikan adalah sebagai berikut:

Tabel 1. Hasil Tes pada Pra Tindakan / Siklus

No	Kondisi	Frekuensi	Presentase	Rataan tkp
1	Siswa yang memperoleh nilai ≥ 75	5	38,46 %	68,53
2	Siswa yang memperoleh nilai < 75	8	61,53 %	
	Jumlah	13	100 %	

Tabel di atas menunjukkan bahwa ketrampilan pemrograman siswa sebanyak 38,46 % telah tuntas belajar pada Kriteria Ketuntasan Minimal (KKM) yaitu 75. Hal ini berarti masih terdapat 61,53 % siswa belum tuntas belajar. Sedangkan rata-rata tes kemampuan penalaran siswa pada pra penelitian tindakan kelas sebesar 68,53 yang berarti masih dibawah nilai batas KKM.

2. Siklus 1

- a. Perencanaan Tindakan
- b. Pelaksanaan Tindakan
- c. Observasi dan Evaluasi

Berikut detail data observasi proses pembelajaran siswa yang memperlihatkan keterlaksanaannya tahap tahap dalam penerapan metode belajar HCSC :

Tabel 2. Checklist Pengamatan Keterlaksanaan Proses Pembelajaran HCSC siklus

Tahap	Indikator	Pertemuan 1
<i>Programming Model (PM)</i>	Memberikan penjelasan yang komprehensif dan contoh yang relatif dalam pemrograman yang sesungguhnya	✓
	Mengarahkan siswa untuk dapat menerapkan masing - masing konsep pada tempat yang tepat	✓
<i>High Level Abstraction (HLA)</i>	Mengarahkan siswa untuk dapat mendefinisikan kebutuhan sistem, ruang lingkup sistem, object - object yang terlibat	✓

	Membantu siswa mengidentifikasi object dan atribut dengan menggunakan visualisasi berupa gambar, film atau animasi	✓
Middle Level Abstraction (MLA)	Menjelaskan cara mentransformasi visualisasi dari HLA menjadi <i>diagram class</i>	✓
	Membantu siswa mengimplementasikan hal yang abstrak menjadi kode program	✓
Detail Implementation (DI)	Membantu siswa untuk mentransformasikan <i>class diagram</i> pada tahapan MLA menjadi kode	✓
	Mengarahkan siswa agar bisa mengimplementasikan algoritma berdasarkan <i>sequence diagram</i>	✓

Dari tabel di atas diperoleh informasi bahwa proses pembelajaran dengan metode HCSC sudah diterapkan secara keseluruhan pada siklus pertama dan diakhir siklus diberikan tes akhir siklus 1 untuk mengetahui adanya peningkatan ketrampilan pemrograman siswa sebelum diberi perlakuan dan sesudah diberi perlakuan. Adapun hasil tes siswa pada akhir siklus 1 disajikan pada tabel sebagai berikut :

Table 3. Hasil Tes pada Siklus 1

Kondisi	Frekuensi	Presentase	Rataan tkp
Siswa yang memperoleh nilai ≥ 75	9	69,23 %	73,92
Siswa yang memperoleh nilai < 75	4	30,76 %	
JUMLAH	13	100 %	

Tabel di atas memberikan informasi bahwa adanya peningkatan rata-rata hasil belajar siswa sebelum dan sesudah penelitian tindakan kelas menjadi 73,92. Jika dilihat dari indikator hasil pada rata-rata hasil tes ketrampilan pemrograman siswa masih belum mencapai ketuntasan. Hal ini dikarenakan rata-ratanya belum mencapai ketuntasan minimal yaitu 75. Juga pada siklus I masih terdapat 30,76 % siswa yang belum memenuhi batas ketuntasan minimal meski rata-rata siswa yang belum tuntas masih pada taraf interval yang normal. Hal ini menunjukkan hasil yang belum optimal sehingga perlu diadakan kembali pembelajaran pada siklus II.

d. Refleksi

Tujuan refleksi adalah melakukan evaluasi hasil tindakan penelitian yang telah dilakukan pada siklus I.

3. Siklus 2

- a. Perencanaan Tindakan
- b. Pelaksanaan Tindakan
- c. Observasi dan Evaluasi

Berikut detail data observasi proses pembelajaran siswa yang memperlihatkan keterlaksanaannya tahap tahap dalam penerapan metode belajar HCSC :

Table 4. Checklist Pengamatan Keterlaksanaan Proses Pembelajaran HCSC siklus

Tahap	Indikator	Pertemuan 2
Programming Model (PM)	Memberikan penjelasan yang komprehensif dan contoh yang relatif dalam pemrograman yang sesungguhnya	✓
	Mengarahkan siswa untuk dapat menerapkan masing - masing konsep pada tempat yang tepat	✓
High Level Abstraction (HLA)	Mengarahkan siswa untuk dapat mendefinisikan kebutuhan sistem, ruang lingkup sistem, object - object yang terlibat	✓
	Membantu siswa mengidentifikasi object dan atribut dengan menggunakan visualisasi berupa gambar, film atau animasi	✓
Middle Level Abstraction (MLA)	Menjelaskan cara mentransformasi visualisasi dari HLA menjadi <i>diagram class</i>	✓
	Membantu siswa mengimplementasikan hal yang abstrak menjadi kode program	✓
Detail Implementation (DI)	Membantu siswa untuk mentransformasikan <i>class diagram</i> pada tahapan MLA menjadi kode	✓
	Mengarahkan siswa agar bisa mengimplementasikan algoritma berdasarkan <i>sequence diagram</i>	✓

Dari tabel tersebut diperoleh informasi bahwa proses pembelajaran dengan metode HCSC sudah diterapkan secara keseluruhan pada siklus II, setelah diberikan perlakuan berupa proses pembelajaran melalui metode HCSC, kemudian diadakan tes akhir siklus II untuk mengetahui adanya peningkatan rata-rata keterampilan pemrograman siswa dari siklus I dengan siklus II dan banyaknya persentase siswa yang tuntas dalam materi *Class* dan *Object*.

Adapun hasil tes penalaran siswa pada akhir siklus II disajikan pada tabel berikut:

Table 5. Hasil Tes pada Siklus 2

No	Kondisi	Frekuensi	Presentase	Rataan t _{kp}
1	Siswa yang memperoleh nilai ≥ 75	11	84,61 %	80,03
2	Siswa yang memperoleh nilai < 75	2	15,38 %	
	JUMLAH	13	100 %	

Tabel memberikan informasi bahwa adanya peningkatan rata-rata hasil belajar siswa berupa keterampilan pemrograman pada siklus II adalah 80,03. Persentase ketuntasan belajar siswa mencapai 84,61 % dengan jumlah siswa 13 orang dan yang belum tuntas 2 orang. Secara klasikal hasil belajar berupa keterampilan pemrograman siswa pada pembelajaran siklus II sudah mencapai target karena melebihi indikator yang ditetapkan 80%.

d. Refleksi

Data hasil observasi pada siklus II setelah dievaluasi, menunjukkan bahwa selama pembelajaran dengan metode HCSC pada siklus II tidak terdapat kendala yang berarti dan secara keseluruhan pelaksanaan baik dan lancar. Hal ini dapat dilihat dari indikator keberhasilan hasil belajar dalam pembelajaran melalui model HCSC telah tercapai. Penilaian hasil belajar berupa rata-rata keterampilan pemrograman mencapai 80,03, hal ini berarti melebihi KKM yang sudah ditentukan yaitu 75 dengan siswa yang tuntas sebanyak 13 orang (84,61 %).

Dengan demikian peneliti dan observer memutuskan keterampilan pemrograman pada materi *Class* dan *Object* meningkat melalui pembelajaran dengan metode *Hemispheric Cognitive Style Collaboration (HCSC)* sehingga penelitian tindakan kelas diakhiri pada siklus II.

Table 6. Perbandingan Tes Kemampuan Penalaran Siswa pada tiap Siklusnya

Kondisi	Pra Siklus			Siklus I			Siklus II		
	Banyak Siswa	%	Rataan	Banyak Siswa	%	Rataan	Banyak Siswa	%	Rataan
Peserta didik yang memperoleh nilai ≥ 75	5	38,46	68,53	9	69,23	73,92	11	84,61	80,03
Peserta didik yang memperoleh nilai < 75	8	61,53		4	30,76		2	15,38	

Berdasarkan tabel tersebut diperoleh bahwa keterampilan pemrograman siswa mengalami peningkatan. Siswa dinyatakan tuntas apabila mencapai nilai lebih dari ketuntasan Kriteria Minimal (KKM) yaitu 75. Indikator keberhasilan penelitian dapat dilihat dari peningkatan rata - rata keterampilan pemrograman siswa dari tiap siklusnya. Dari tabel diperoleh rata-rata keterampilan pemrograman pada pra siklus adalah 68,53. Setelah adanya perlakuan pada siklus I diperoleh peningkatan rata-rata keterampilan pemrograman siswa menjadi 73,92. Dan meningkat juga setelah adanya

perlakuan pada siklus II, diperoleh rata-rata keterampilan pemrograman meningkat menjadi 80,03.

Selain dilihat dari indikator keberhasilan melalui peningkatan rata-rata keterampilan pemrograman siswa yang menandakan sebagai ketuntasan klasikal, indikator keberhasilan penelitian juga dilihat dari peningkatan ketuntasan secara individual dengan melihat dari banyaknya siswa yang tuntas pada tes keterampilan pemrograman melebihi 80%.

Pembahasan

Penerapan metode HCSC untuk meningkatkan keterampilan pemrograman dilaksanakan di kelas XI RPL SMKN 2 Poso Pesisir yang berjumlah 13 siswa. Metode HCSC dikembangkan dari beberapa metode yaitu; *Object - First Approach* (Kaasboll, Berge, Borge, Fjuk, Holmboe, & Samuelsen, 2003), *Model - First Approach* (Bennedsen & C aspersen, 2004), dan *Iterative Methodology* (Hadar I. , 2007), *Metode Object - First Approach* beranggapan bahwa siswa yang telah mempelajari pemrograman terstruktur atau imperative akan mempengaruhi orientasi pemrograman siswa. Sementara metode *Model - First Approach* menggunakan ilustrasi model dengan bantuan UML untuk memvisualisasikan abstraksi dari kode dalam bentuk class diagram. Sedangkan metode *Iterative* juga menggunakan bantuan model untuk memvisualisasi abstraksi, baru kemudian dari model di-implementasikan kekode yang dilakukan secara berulang-ulang antara model dan kode.

Dalam metode HCSC ini diperhatikan juga sisi *Hemispheric Cognitive Style* yaitu pola yang bersifat khas dari pembelajar ketika merasakan, berpikir, mengingat dan memecahkan masalah baik itu menggunakan belahan otak kiri maupun otak kanan atau keduanya. Metode ini berusaha mengoptimalkan fungsi kognitif belahan otak kiri dan kanan dalam mempelajari konsep OOP yang membutuhkan kemampuan kognitif seperti abstraksi, analisis dan pemecahan masalah, karena dalam proses belajar umumnya seseorang akan mengingat suatu hal terpicu dengan hal-hal yang ada di sekelilingnya baik yang dilihat maupun yang didengar, seperti semacam trigger yang mengaktifkan memori yang menyimpan kenangan terkait.

Hal ini dapat terlihat dari tahapan-tahapan pada metode HCSC yang terdiri dari empat langkah yang berkesinambungan yaitu :

1. *Programming Model (PM)*

Definisi : *Programming Model* adalah konsep dasar dan struktur dari pemrograman yang meliputi; *variable, data type, program execution, debugging, synth, class, function* dan *control flow* yang terdiri dari; *sequence, branching, looping, dan error trapping.*

Pada tahap awal (*Programming Model*) siswa diperkenalkan dengan sintak dasar bahasa java dan struktur program (*sequence, branching, looping*). Tahap ini cukup menyulitkan bagi siswa yang tidak memiliki dasar pemrograman. Biasanya mereka melakukan kesalahan pada pengetikan sintak huruf besar dan huruf kecil, juga keyword

yang cukup banyak harus dihafal. Tahap ini biasanya dapat dilalui dengan mudah bagi tipe pelajar *field independent*

2. High Level Abstraction (HLA)

Definisi : *High Level Abstraction* menggambarkan permasalahan yang sesungguhnya di dunia nyata dimana program akan diimplementasikan. Berdasarkan riset abstraksi adalah konsep yang fundamental dalam pemrograman umumnya dan khususnya pemrograman berorientasi objek (Bennedsen & Caspersen, 2007). Tahap ini memaksimalkan fungsi kognitif dari *imagination, pattern's of behavior* dan *visual memory*.

Pada tahap kedua (*High Level Abstraction*) siswa diperkenalkan dengan konsep OOP dengan bantuan visualisasi pada *real world domain problem*. Contoh yang sederhana siswa diajarkan untuk mengidentifikasi atribut dan metode dari objek nyata misalnya mobil, orang, atau system. Contoh objek tersebut adalah objek yang ada dalam dunia nyata. Dengan mudah siswa dapat mengidentifikasi atribut dan metode yang mungkin mereka miliki. Tahapan ini mengilustrasikan *real world domain* (Kaasboll, Berge, Borge, Fjuk, Holmboe, & Samuelsen, 2003).

3. Middle Level Abstraction (MLA)

Definisi : *Middle Level Abstraction* mentransformasikan high level abstraction menjadi model diagram (UML) yang lebih dekat kepada kode pemrograman. Hal ini diperlukan untuk menjembatani gap antara high level abstraction dan detail implementation. Karena siswa seringkali merasa sulit mengimplementasikan hal yang abstrak menjadi kode atau menggambarkan logis dari abstraksi menjadi algoritma.

Pada tahap ketiga (*Middle Level Abstraction*) siswa dituntun untuk menggambarkan *class diagram* berdasarkan atribut dan metode yang ada pada ilustrasi dengan menggunakan *tools* UML dalam riset ini *tools* yang digunakan adalah Netbeans. *Tools* ini walaupun agak sedikit berat tapi fiturnya lebih banyak.

4. Detail Implementation (DI)

Definisi : Tahapan coding dimana hasil dari tahapan HLA, dan MLA di-implementasikan dalam bentuk kode.

Pada tahap keempat (*Detail Implementation*) siswa diajarkan untuk men-generate kode dengan menggunakan Netbeans yang secara otomatis akan men-generate kode berdasarkan class diagram tersebut. Kemudian pada level detail ini siswa diminta menuliskan kode yang dibutuhkan untuk metode - metode yang ada. Di sini fungsi kognitif belahan otak kiri yang bekerja.

Pembelajaran model HCSC mempunyai keunggulan dalam mengoptimalkan kemampuan kognitive belahan otak kiri dan kanan dalam mempelajari konsep OOP yang membutuhkan kemampuan kognitif seperti abstraksi, analisis dan pemecahan masalah.

IV. KESIMPULAN DAN SARAN

Kesimpulan

Dari uraian sebelumnya ditemukan bahwa programmer dan siswa mengalami kesulitan dalam memahami dan menerapkan konsep OOP dalam pemrograman. Dari hasil riset ditemukan bahwa dalam pembelajaran OOP dipengaruhi *cognitive development* dan *cognitive style* dari siswa Dengan memperhatikan dua hal tersebut (*cognitive style* dan *cognitive development*), metode HCSC terbukti dapat meningkatkan pemahaman siswa terhadap materi OOP dan dapat menerapkannya dalam pemrograman.

Saran

Adapun saran yang diajukan sesuai dengan penelitian yang dilakukan adalah:

1. Dalam mengembangkan teknik pembelajaran perlu diperhatikan kemampuan *cognitive development* dari siswa jangan sampai berada jauh di atasnya karena siswa akan mengalami kegagalan dalam mempelajarinya
2. Selain itu perlu juga diperhatikan *cognitive style* dari siswa apakah siswa *left hemisphere dominant* atau *right hemisphere dominant*. Usahakan untuk mengkolaborasikan kedua belah *hemisphere* untuk proses belajar yang lebih optimal.

DAFTAR PUSTAKA

- [1] Ahmad Abdul Jawwad, M. (2002). *Mengembangkan Inovasi dan Kreativitas Berpikir*. Bandung: Syaamil Cipta Media.
- [2] Arikunto, Suharsimi. (2006). *Prosedur Penelitian; Suatu Pendekatan Praktik*. Jakarta; Rineka Cipta.
- [3] Hadar, Irit. (2007) *An Iterative Methodology for Teaching ObJect Oriented Concepts.*, from <https://infedu.vu.lt/journal/INFEDU/article/547/info>
- [4] Hadar,Irit, (2004) *The Study Of Concept Understanding via Abstract Representation: The Case Of OOP Design* from https://www.academia.edu / 19552912 / An _iterative _methodology _for_teaching_object_oriented_concepts
- [5] Kaasboll,Jens,ct al (2003) *Learning Object - Oriented Programming* from <https://studylib.net/doc/11379681/learning-object-oriented-programming-ola-berge-richard-ed...>
- [6] Tim, Pelatihan, Proyek, PGSM. 1999. *Penelitian Tindakan Kelas (Classroom Action Research)*. Jakarta: DEPDIKBUD, Direktorat Jenderal Penguruan Tinggi.
- [7] White, Garry and Sivitanides, Marcos. (2005) *Cognitive Differences Between Procedural Programming and Object Oriented Programing* from <https://www.proquest.com / docview / 194486087>
- [8] Yoni, Acep. (2010). *Menyusun Penelitian Tindakan Kelas*. Yogyakarta: Familia