

**KORELASI MODEL PEMBELAJARAN MENGGUNAKAN
INDOBLOCKLY TERHADAP PEMAHAMAN MAHASISWA
PADA MATA KULIAH PEMROGRAMAN TERSTRUKTUR**

Skripsi
untuk memenuhi sebagian persyaratan
mencapai derajat Sarjana S-1
Program Studi Teknik Informatika



Disusun Oleh
Rischan Mafrur
09650007

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UIN SUNAN KALIJAGA
YOGYAKARTA
2013**



Universitas Islam Negeri Sunan Kalijaga

FM-UINSK-BM-05-07/R0

PENGESAHAN SKRIPSI/TUGAS AKHIR

Nomor : UIN.02/D.ST/PP.01.1/326/2013

Skrripsi/Tugas Akhir dengan judul : Korelasi Model Pembelajaran Menggunakan Indoblockly Terhadap Pemahaman Mahasiswa Pada Mata Kuliah Pemrograman terstruktur

Yang dipersiapkan dan disusun oleh :
Nama : Rischana Mafrur
NIM : 09650007
Telah dimunaqasyahkan pada : Jum'at. 25 Januari 2013
Nilai Munaqasyah : A
Dan dinyatakan telah diterima oleh Fakultas Sains dan Teknologi UIN Sunan Kalijaga

TIM MUNAQASYAH :

Ketua Sidang

Agung Fatwanto, Ph.D
NIP. 19820511 200604 2 002

Penguji I

Agus Mulyanto, M.Kom
NIP.19710823 199903 1 003

Penguji II

Sumarsono, M.Kom
NIP. 19710209 200501 1 003

Yogyakarta, 4 Februari 2013

UIN Sunan Kalijaga
Fakultas Sains dan Teknologi
Jember



Prof. Drs. H. Akh. Minhaji, M.A, Ph.D
NIP. 19580919 198603 1 002



SURAT PERSETUJUAN SKRIPSI/TUGAS AKHIR

Hal : Permohonan

Lamp : -

Kepada

Yth. Dekan Fakultas Sains dan Teknologi

UIN Sunan Kalijaga Yogyakarta

di Yogyakarta

Assalamu'alaikum wr. wb.

Setelah membaca, meneliti, memberikan petunjuk dan mengoreksi serta mengadakan perbaikan seperlunya, maka kami selaku pembimbing berpendapat bahwa skripsi Saudara:

Nama : Rischana Mafrur

NIM : 09650007

Judul Skripsi :

Pengaruh Model Pembelajaran Menggunakan *IndoBlockly* (Bahasa Pemrograman Visual Block) terhadap Pemahaman Mahasiswa pada Mata Kuliah Pemrograman Terstruktur (Studi pada Mahasiswa Semester I Angkatan 2012/2013 Teknik Informatika UIN Sunan Kalijaga Yogyakarta)

sudah dapat diajukan kembali kepada Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu dalam Teknik Informatika

Dengan ini kami berharap agar skripsi/tugas akhir Saudara tersebut di atas dapat segera dimunaqsyahkan. Atas perhatiannya kami ucapkan terima kasih.

Wassalamu'alaikum wr. wb.

Yogyakarta, 19 Desember 2012

Pembimbing

Agung Fatwanto, Ph.D

NIP: 19770103 200501 1 003

PERNYATAAN KEASLIAN SKRIPSI

Yang bertanda tangan di bawah ini :

Nama : Rischon Mafrur
Nim : 09650007
Program Studi : Teknik Informatika
Fakultas : Sains dan Teknologi

Menyatakan bahwa skripsi dengan judul **Pengaruh Model Pembelajaran Menggunakan *IndoBlockly* (Bahasa Pemrograman Visual Block) terhadap Pemahaman Mahasiswa pada Mata Kuliah Pemrograman Terstruktur (Studi pada Mahasiswa Semester I Angkatan 2012/2013 Teknik Informatika UIN Sunan Kalijaga Yogyakarta)** tidak terdapat pada karya yang pernah diajukan untuk memperoleh gelar sarjana di suatu Perguruan Tinggi, dan sepengetahuan saya tidak terdapat karya atau pendapat yang pernah ditulis oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 12 Desember 2012

Yang Menyatakan

Rischon Mafrur
NIM : 09650007

METERAI
TEMPEL
9B0BCABF2B8951818
6000
DJP

KATA PENGANTAR

Alhamdulillahirabbi'alamin. Puji syukur bagi Allah Subhanahu wa Ta'ala yang telah melimpahkan rahmat-Nya kepada penulis sehingga penulis dapat menyelesaikan skripsi dengan judul Pengaruh Model Pembelajaran Menggunakan *IndoBlockly* (Bahasa Pemrograman Visual Block) terhadap Pemahaman Mahasiswa pada Mata Kuliah Pemrograman Terstruktur (Studi pada Mahasiswa Semester I Angkatan 2012/2013 Teknik Informatika UIN Sunan Kalijaga Yogyakarta) dengan lancar dan tanpa suatu halangan apapun. Sholawat dan Salam senantiasa penulis haturkan kepada junjungan nabi agung, Muhammad Shollallahu'alaihi wa Sallam.

Selanjutnya penulis mengucapkan terima kasih kepada:

1. Ibunda tercinta dan seluruh anggota keluarga tersayang yang senantiasa mendo'akan dan memberikan support.
2. Prof. Drs. H. Akh. Minhaji, M.A.,Ph.D. selaku Dekan Fakultas Sains & Teknologi UIN Sunan Kalijaga.
3. Bapak Agus Mulyanto, M.Kom, selaku Ketua Program Studi Teknik Informatika UIN Sunan Kalijaga.
4. Bapak Agung Fatwanto, S.Si., M.Kom., Ph.D, selaku Pembimbing yang dengan kesabarannya telah membimbing selama ini.
5. Bapak Sumarsono, S.T., M.Kom., yang sudah memberikan ijin penelitian di kelas praktikum Pemrograman Terstruktur Teknik Informatika 2012/2013.
6. Bapak Romi Satrio Wahono yang sudah memberikan berbagai pencerahan mengenai penelitian ini.
7. Mas Adit laboran pendidikan matematika yang sudah memberikan banyak ilmu mengenai statistik.
8. Para Dosen Program Studi Teknik Informatika yang telah memberi bekal ilmu pengetahuan kepada penulis, semoga ilmunya menjadi amal jariyah di dunia hingga akhirat.
9. Angga Maulana, Damar Mustiko Aji, Rosan Qodirin, Ahmad Syaiful, dan Agus Hidayatullah sebagai pengembang *IndoBlockly*.
10. Teman-teman Program Studi Teknik Informatika, khususnya angkatan 2009 yang telah banyak memberi dukungan.
11. Semua pengurus Yayasan Masjid Prayan Raya beserta teman-teman takmir Masjid Prayan Raya yang banyak memberi dukungan.

Penulis merasa masih banyak sekali kekurangan dan kelemahan dalam penelitian ini, oleh karena itu segala kritik dan saran senantiasa penulis harapkan dari para pembaca. Akhir kata, semoga penelitian ini dapat

menjadi panduan serta referensi yang sangat berguna bagi pembaca dan dapat dimanfaatkan sebaik-baiknya.

Yogyakarta,
Yang Menyatakan

Rischan Mafrur
NIM. 09650007

HALAMAN PERSEMBAHAN

Kupersembahkan untuk :

- ❖ *Sembah sujudku kepada Allah Subhanahu wa Ta'ala yang senantiasa melimpahkan rahmat dan hidayah-Nya.*
- ❖ *Sholawat dan Salam kepada junjungan nabi besar Muhammad Shollallahu'alaihi wa Sallam.*
- ❖ *Ibuku,Ibuku,Ibuku dan ayahku tercinta yang tak pernah henti-hentinya berjuang demi aku. Semoga anakmu ini bisa membahagiakan ibu dan bapak, dan mohon doanya wahai ibu dan bapakku semoga anakmu ini bisa memberikan manfaat bagi ibu bapak, keluarga, masyarakat, bangsa dan umat islam ini.*
- ❖ *Adikku tersayang Anjani, jangan nakal yah, belajar yang tekun supaya besok bisa lebih baik dari masmu ini :-D.*
- ❖ *Bapak Agus Mulyanto yang selama keberadaan penulis dalam menuntut ilmu di Jogja ini sudah seperti ayah kedua bagi penulis dan selaku murrabi bagi penulis juga yang selalu sabar membimbing dan memberikan nasehat-nasehat dan saran yang tak ternilai harganya. Semoga Allah selalu melindungi Pak Agus sekeluarga.*
- ❖ *Bapak Agung Fatwanto yang telah banyak sekali memberikan ilmu-ilmu dan diskusi yang bisa menambah pengetahuan penulis. Semoga Allah senantiasa memberikan kemudahan dan petunjuk-Nya untuk Pak Agung dan tak lupa semoga dengan bimbingan Allah saya segera mengikuti jejak Pak Agung.*
- ❖ *Bapak Romi Satrio Wahono, terimakasih atas diskusinya pak dan juga dukungannya terhadap penelitian saya.*
- ❖ *Bapak Sumarsono terimakasih banyak atas ijinnya untuk melakukan penelitian di kelas praktikum Pemrograman Terstruktur pak, dan*

terimakasih banyak atas masukan-masukkannya pak. Semoga Allah melindungi Pak Sumarsono dan keluarga.

- ❖ Mas Adit laboran PMat yang sudah banyak mengajari SPSS, matur suwun lhoo mas, insyaAllah dadi amal jariyah :-D.
- ❖ Bapak Anshori yang walaupun belum pernah mengajar penulis dan hanya ketemu ngobrol sesaat tapi dari motivasi beliau penulis menjadi selalu bersemangat.
- ❖ Ibu Uyun yang selalu asik diajak diskusi mengenai AI ayoo maju AI Indonesia hehe :-D. Semangat terus bu mendidik calon2 peneliti Informatika dari UIN Sunan Kalijaga.
- ❖ Dosen Teknik Informatika: Pak Nurochman terimakasih ilmu JSTnya pak, Pak Aulia doakan biar cepet gak jomblo pak hee :-D, Pak Taufik, Pak Mustakim, Pak Bambang, Pak Didik, Ibu Ulfa, Ibu Ade , terimakasih untuk semua Ilmu yang sudah diajarkan, ilmu ini akan menjadi amal jariyah yang tidak akan pernah putus, Semoga Allah melindungi Bpk ibu Dosen semuanya.
- ❖ Teman-teman terdekatku Fadli (he's no life :-D coding terus ...) dan Krocot/Sholahudin (kembang asem) terimakasih tumpangan kosnya yang hampir setiap hari aku selalu disitu :-D.
- ❖ Teman-teman terdekatku Hafidh "ojo prengas prenges ae fid", pulung aktifis KRPH tetep istiqomah lung, udin "ojo salah ngriting din, saake umimu :-D", Aspar "semoga menjadi ulil albab (orng2 yang berpikir) :-D", Anik "entah sampai sekarang gak tau kamu itu cewe ato cowo nik :-D". Terimakasih teman-teman ayoo ndang nyusul, ndang lulus, wis di usir pak Agus heee :-D.
- ❖ Seluruh teman-teman Teknik Informatika: Kambing, Yosep, Ahdi, Pasa, Estu, Sigit, Oki, Izal, Kusuma, Yanuar, Dimas, Kiki, Ayu, Ratna, Ulin, Delisa, Disa, devi, Amy, Ismi, latip, joko, Lukman dan masih banyak lagi yang tidak bisa kusebutkan satu persatu,

"KELUARGA BESAR TIF 09.. KESUKSESAN ADA DI DEPAN KITA.. SEMANGADD!!!!!!".

- ❖ *Iostream.in crew (informatics research team), salam srigala berkepala 9.*
- ❖ *Mas-mas alumni Teknik Informatika yang menginspirasi, Mas Ganjar, Mas Rifki, Mas Alex, semoga bisa menyusul mas :-D. Mas Fathan yang selalu ngasih nasehat untuk segera nikah dan yang pasti yang sering ngasih siswa untuk private atau proyek juga, Mas Sigit, Mas Sunu, Mas Fendi, Mas Veta, Mas Saiful, Mas Budeng, Warok Ngalek, makasih ilmu-ilmu yang sudah ditularkan mase :-D.*
- ❖ *Teman-teman Laboran Mas Iqbal, Mas Nawir, Mas Rian, dan Mas Yusuf, Terimakasih banyak mas, sudah sering ngrepoti selama ini.*
- ❖ *Seluruh teman-teman pengembang IndoBlockly : Angga, Damar, Rosan, Syaiful dan Agus. Tetep maju IndoBlockly untuk bangsa Indonesia :-D.*
- ❖ *Bapak Nur Mukhlis selaku Ketua Takmir Masjid Prayan Raya yang sudah memberikan ijin dari semester 1 sampai saat ini untuk tinggal di Masjid Prayan Raya, dan juga seluruh warga Prayan Kulon yang sudah saya anggap sebagai keluarga.*
- ❖ *Teman-teman takmir Masjid Prayan Raya : Gembuskun, hendrik, mas ompol dewo, mas hilmy, teman-teman seperjuangan yang berusaha selalu memakmurkan masjid, maaf yah klo tidur di atas subuh sering krinan :-D #krinan koq bendino.*
- ❖ *Teman-teman KKN yang penuh kenangan: Irul, Zaid, Habib, Wahyu, Sulis, Julida, Asti, Iis, Nurika, Ibu Sekar.*
- ❖ *Teman-teman Imagine : Pak Bos Pak Agung, Mbak Esa, Mas Arul, Mas Arfin, Joko, Rio, Gabriel dsb.*
- ❖ *Yang terakhir adalah seseorang yang ada disana untuk menunggu kedatanganku, semoga Allah menyegerakan dan slalu memberikan jalan yang terbaik bagi kita. Allahu A'lam :-D.*

HALAMAN MOTTO

berdiri di depan cermin, dan sy melihat seseorang yang
besok akan memimpin dunia #dWorldConqueror

DAFTAR ISI

Halaman Judul.....	i
Pengesahan Skripsi/Tugas Akhir	ii
Surat Persetujuan Skripsi/Tugas Akhir	iii
Pernyataan Keaslian Skripsi.....	iv
Kata Pengantar	v
Halaman Persembahan	vii
Halaman Motto.....	x
Daftar Isi.....	xi
Daftar Tabel	xv
Daftar Gambar	xvi
Daftar Lampiran	xvii
Abstraksi	xviii
Abstract	xix
BAB I PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
1.6 Keaslian Penelitian.....	6
BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI.....	7
2.1 Tinjauan Pustaka	7

2.2 Landasan Teori.....	13
2.2.1 Pengertian Belajar	13
2.2.2 Media Pembelajaran	14
2.2.3 Proses Pembelajaran.....	15
2.2.3.1 Pengertian Ranah Penilaian Kognitif.....	17
2.2.3.2 Ranah Afektif dan Psikomotorik	20
2.2.4 Belajar Pemrograman (Learning Programming).....	21
2.2.5 Bahasa Pemrograman Visual (Visual Programming Language /VPLs)	24
2.2.5.1 Kelebihan dan Kekurangan Bahasa Pemrograman Visual	25
2.2.5.1.1 Kelebihan VPLs	25
2.2.5.2 Kekurangan VPLs.....	26
2.2.6 IndoBlockly	27
2.2.6.1 Konsep IndoBlockly	27
2.2.6.2 Design Interface IndoBlockly.....	29
2.2.6.3 Contoh Pembuatan Program Sederhana menggunakan IndoBlockly...	33
BAB III METODOLOGI PENELITIAN	36
3.1 Metode Penelitian.....	36
3.2 Populasi dan Sampel	36
3.2.1 Populasi	37
3.2.2 Sampel	37
3.3 Desain Penelitian.....	38
3.4 Alur Penelitian	39
3.5 Prosedur Penelitian.....	39
3.5.1 Persiapan Penelitian	39
3.5.2 Pelaksanaan Penelitian	40

3.5.3 Penyelesaian Penelitian	40
3.6 Instrumen Penelitian.....	41
3.6.1 Seperangkat Soal	41
3.6.1.1 Menentukan Validitas Soal.....	42
3.6.1.2 Menentukan Reliabilitas Soal	44
3.6.1.3 Menentukan Daya Pembeda Soal	45
3.6.1.4 Menentukan Tingkat Kesukaran Soal.....	46
3.6.2 Hasil Uji Instrumen	47
3.6.2.1 Hasil Uji Instrumen Expert.....	48
3.6.2.2 Hasil Uji Instrumen Uji Coba	48
3.7 Teknik Pengumpulan Data.....	51
3.8 Teknik Pengolahan Data	51
3.8.1 Pengolahan Data Kuantitatif	51
3.8.1.1 Analisis Data Tes	51
3.8.1.1.1 Analisis Deskriptif	53
3.8.1.1.2 Analisis Inferensi	53
3.8.1.1.2.1 Uji Normalitas	53
3.8.1.1.2.2 Uji Homogenitas Varians	54
3.8.1.1.2.3 Uji Perbedaan Dua Rata-rata.....	54
3.8.1.1.2.4 Analisis Korelasi Pretes Postes Kelas Ekperimen	55
BAB IV HASIL PENELITIAN DAN PEMBAHASAN.....	57
4.1 Hasil Penelitian dan Pembahasan.....	57
4.1.1 Analisis Deskriptif Data Hasil Pretes dan Postes	58
4.1.2 Analisis Inferensi Data Hasil Pretes dan Postes	60
4.1.2.1 Analisis Data Kemampuan Awal Mahasiswa.....	60

4.1.2.1.1 Uji Normalitas Data Pretes	63
4.1.2.1.2 Uji Perbedaan Dua Rata-rata Data Pretes	64
4.1.2.2 Analisis Data Kemampuan Akhir Mahasiswa.....	66
4.1.2.2.1 Uji Normalitas Data Postes.....	69
4.1.2.2.2 Uji Perbedaan Dua Rata-rata Data postes	70
4.1.2.3 Analisis Data Kualitas Peningkatan Kemampuan Mahasiswa	71
4.1.2.4 Analisis Data Korelasi Pretes Postes Kelas Eksperimen.....	73
BAB V PENUTUP	78
5.1 Kesimpulan	78
5.2 Saran.....	78
DAFTAR PUSTAKA	80
LAMPIRAN	83

DAFTAR TABEL

Tabel 3.1 Desain Kelompok Eksperimen dan Kontrol Pretes-Posttes	39
Tabel 3.2 Klasifikasi Koefisien Validitas	43
Tabel 3.3 Klasifikasi Koefisien Reliabilitas	45
Tabel 3.4 Klasifikasi Koefisien Daya Pembeda.....	46
Tabel 3.5 Klasifikasi Koefisien Indeks Kesukaran	47
Tabel 3.6 Hasil Analisis Instrumen Tes Uji Coba	50
Tabel 3.7 Teknik Pengumpulan Data.....	51
Tabel 3.8 Kategori Gain yang dinormalisasi.....	52
Tabel 3.9 Kriteria Penilaian Korelasi	56
Tabel 4.1 Statistik Deskriptif Data Hasil Pretes dan Postes	58
Tabel 4.2 Data Statistik Skor Pretes Kelas Eksperimen dan Kelas Kontrol.....	61
Tabel 4.3 Hasil Uji Normalitas Data Pretes	64
Tabel 4.4 Hasil Uji Mann-Whitney Data Pretes	65
Tabel 4.5 Statistika Deskriptif Skor Postes Kelas Eksperimen dan Kelas Kontrol...	66
Tabel 4.6 Hasil Uji Normalitas Data Postes.....	70
Tabel 4.7 Hasil Uji Mann Whitney Data Postes	71
Tabel 4.8 Statistik Deskriptif Skor Indeks Gain Tes Kemampuan Mahasiswa Kelas Eksperimen dan Kelas Kontrol	72
Tabel 4.9 Daftar Persentase Kualitas Peningkatan Kemampuan Mahasiswa Kelas Eksperimen dan Kelas Kontrol	72
Tabel 4.10 Hasil Uji Normalitas Data Pretes Postes kelas eksperimen	74
Tabel 4.11 Hasil Uji Korelasi Wilcoxon Data Pretes Postes Kelas Eksperimen.....	75
Tabel 4.12 Koefisien Korelasi antara Pretes dan Postes Kelas Eksperimen	76
Tabel 4.13 Uji Signifikansi Koefisien Korelasi antara Pretes dan Postes Kelas Eksperimen.....	77

DAFTAR GAMBAR

Gambar 2.1 Hirarki Piramida Taksonomi Bloom	20
Gambar 2.2 Konsep IndoBlockly	27
Gambar 2.2 Halaman Indeks aplikasi IndoBlockly	29
Gambar 2.3 Input output sederhana menggunakan IndoBlockly	30
Gambar 2.4 Menu Teks pada IndoBlockly	30
Gambar 2.5 Contoh penggunaan variabel pada IndoBlockly	31
Gambar 2.6 Contoh array di IndoBlockly	31
Gambar 2.7 Penggunaan if pada IndoBlockly	32
Gambar 2.8 Penggunaan for pada IndoBlockly	32
Gambar 2.9 Program sederhana untuk mencari nilai maksimum dari inputan user menggunakan IndoBlockly	33
Gambar 2.10 Aplikasi sederhana perhitungan luas persegi panjang menggunakan IndoBlockly	34
Gambar 2.11 Hasil output running aplikasi sederhana perhitungan luas persegi panjang menggunakan IndoBlockly	34
Gambar 2.12 Hasil konvert code puzzle IndoBlockly ke source code C	35
Gambar 2.13 Hasil output running program dengan C Free.	35
Gambar 3.1 Alur Penelitian	38
Gambar 4.1 Hasil Pretes Kelas Eksperimen dan Kelas Kontrol	62
Gambar 4.2 Q-Q Plot pretes kelas eksperimen	62
Gambar 4.3 Q-Q Plot pretes kelas kontrol	63
Gambar 4.4 Hasil Postes Kelas Eksperimen dan Kelas Kontrol	67
Gambar 4.5 Uji Normalitas dengan Q-Q Plot data postes kelas eksperimen	68
Gambar 4.6 Uji Normalitas dengan Q-Q plot data postes kelas kontrol	69

DAFTAR LAMPIRAN

Lampiran A Pelaksanaan Penelitian.....	83
Lampiran B Analisis Uji Coba Instrumen.....	84
Lampiran C Soal Pretes dan Postes.....	87
Lampiran D Perolehan Data dan Analisis Deskriptif.....	90
Lampiran E Uji Normalitas Data Pretes.....	93
Lampiran F Uji Beda Rata-rata Pretes	94
Lampiran G Uji Normalitas Data Postes.....	95
Lampiran H Uji Beda Rata-rata Postes	96
Lampiran I Perhitungan Indeks Gain	97
Lampiran J Uji Normalitas Data Pretes Postes Kelas Eksperimen	98
Lampiran K Analisis Uji Hipotesis Korelasi dengan Wilcoxon	99
Lampiran L Analisis Korelasi, Uji Signifikansi Korelasi, dan Determinasi.....	100

KORELASI MODEL PEMBELAJARAN MENGGUNAKAN INDOBLOCKLY TERHADAP PEMAHAMAN MAHASISWA PADA MATA KULIAH PEMROGRAMAN TERSTRUKTUR

INTISARI

Kemajuan teknologi dan industri baik di ranah nasional maupun internasional menyebabkan meningkatnya kebutuhan terhadap SDM ahli khususnya programmer ahli. Akan tetapi dari sisi SDM dapat dikatakan sulit untuk menjadi seorang programmer ahli. Dalam penelitian yang dilakukan Wislow disebutkan bahwa butuh waktu 10 tahun bagi seorang programmer pemula untuk menjadi programmer ahli. Kelleher menyebutkan bahwa taksonomi tertinggi dalam *programming environment* adalah *teaching system*. Bagaimana dapat membentuk SDM yang ahli jika sistem pembelajaran tidak mendukung ? Oleh karena itu peneliti mengambil studi di kampus UIN Sunan Kalijaga Yogyakarta pada mata kuliah pemrograman terstruktur tahun 2012/2013. Pembelajaran pada mata kuliah pemrograman terstruktur adalah menggunakan *C Free*. Peneliti mengusulkan penggunaan IndoBlockly yaitu tool pemrograman visual untuk mempermudah pemahaman algoritma. IndoBlockly menggunakan puzzle sebagai media untuk memprogram dan tidak menggunakan text sehingga sangat meminimalisir *error syntax* yaitu masalah yang sering dijumpai oleh programmer pemula.

Penelitian ini lebih ditekankan pada hubungan penggunaan IndoBlockly terhadap hasil belajar mahasiswa pada mata kuliah pemrograman terstruktur. Analisis yang digunakan meliputi analisis deskriptif, perhitungan index gain, analisis inferensi dan analisis korelasi. Data yang kami analisis adalah data nilai mahasiswa hasil dari pretes dan postes. Hasil dari penelitian ini adalah berupa kesimpulan apakah penggunaan IndoBlockly mempunyai pengaruh terhadap hasil pembelajaran mahasiswa pada mata kuliah pemrograman terstruktur.

Analisis uji hitpotesis beda rata-rata untuk data pretes menghasilkan *p-value* $0,749 > 0,05$ sehingga H_0 diterima yaitu tidak ada perbedaan rata-rata pretes kelas eksperimen dan kelas kontrol. Ini menjadi bukti bahwa kelas kontrol dan kelas eksperimen keduanya homogen. Sebaliknya analisis beda rata-rata postes kelas eksperimen dan kelas kontrol menghasilkan *p-value* $0,000 < 0,05$, H_0 ditolak kesimpulannya adalah rata-rata kelas eksperimen dan kelas kontrol berbeda. Perhitungan gain diperoleh rata-rata gain kelas eksperimen sebesar 0,63 adalah gain sedang dan kelas kontrol 0,16 adalah gain rendah, ini menunjukkan bahwa nilai gain kelas eksperimen lebih besar dari kelas kontrol. Analisis korelasi menghasilkan koefisien korelasi sebesar 0,43. Kemudian dilakukan uji signifikansi koefisien korelasi diperoleh *p-value* $0,022 < 0,05$, H_0 ditolak dan disimpulkan bahwa kontribusi variabel independen (X) yaitu penggunaan IndoBlockly terhadap variabel dependen (Y) yaitu hasil belajar mahasiswa adalah signifikan. Hasil perhitungan determinasi yaitu r^2 sebesar 0,1849 atau 18,49 % menunjukkan bahwa IndoBlockly memberikan pengaruh 18,49 % terhadap peningkatan hasil belajar pemrograman terstruktur. Jadi model pembelajaran menggunakan IndoBlockly terbukti lebih baik dibandingkan dengan model pembelajaran konvensional dengan pengaruh sebesar 18,49 % terhadap peningkatan hasil belajar.

Kata Kunci: *IndoBlockly, belajar pemrograman, pemahaman algoritma.*

KORELASI MODEL PEMBELAJARAN MENGGUNAKAN INDOBLOCKLY TERHADAP PEMAHAMAN MAHASISWA PADA MATA KULIAH PEMROGRAMAN TERSTRUKTUR

ABSTRACT

Rapid technological growth of computer technology and industry in the realm nationally and internationally cause increasing demands skilled of human resources expert in particular is expert programmers. However, to be an expert programmer is difficult. In a research Wislow mentioned that takes 10 years for a novice programmers to become expert programmers. Kelleher said that the highest taxonomic programming environment is the teaching system. How to become a expert if the system does not support for learning? Therefore i am a researcher took a study on campus UIN Sunan Kalijaga Yogyakarta on structured programming courses academic year 2012/2013. Structured programming courses in UIN Sunan Kalijaga using C Free (text-based) as a learning medium. Researcher proposed a learning model that uses IndoBlockly, It's visual programming tool that makes it easier to learn algorithms. IndoBlockly use the puzzle as a medium for programming and not use text syntax so it minimizes the problems/ error syntax often encountered by novice programmers.

This study emphasizes the relation between IndoBlockly with the learning outcomes of students in structured programming courses. The analysis that used are descriptive analysis, calculation of gain, inference analysis and correlation analysis. The data is the pretest and posttest learning outcomes of students. The results of this study is whether IndoBlockly have any impact on student learning outcomes at the structured programming courses.

Results from test compare mean analysis for pretest data produce p-value $0.749 > 0.05$ so H_0 is accepted that there is no difference in the mean pretest experimental class and control class. The result is evidence that the experimental class and the control class are homogeneous. Instead the results of the analysis of compare mean posttest data experimental class and control class produce p-value $0.000 < 0.05$ so H_0 is rejected and it was concluded that the mean of experimental class and control class is different. The results of the calculation of the gain is obtained 0.63 for experimental class is the gain medium and gain control class 0.16 is low, it's indicates that the value of the gain experimental class is greater than the gain control classes. Correlation analysis produces a correlation coefficient of 0.43 is a positive correlation with the moderate criteria. Then the significance test of the correlation coefficient obtained p-value $0.022 < 0.05$ which means that H_0 is rejected and it can thus be concluded that the contribution of the independent variable (X) is the use of IndoBlockly on the dependent variable (Y) for student learning outcomes is the significant. The results of the calculation of determination (r^2) is 0.1849 or 18.49% indicated that 18.49% IndoBlockly give effect to the improvement of learning outcomes of structured programming course. So using IndoBlockly learning model is better than conventional learning model with the influence of 18.49% of the increase in learning outcomes.

Keywords: IndoBlockly, learn programming, algorithm comprehension.

BAB I

PENDAHULUAN

1.1 Latar belakang

Pembelajaran Praktikum Pemrograman Terstruktur di Jurusan Teknik Informatika UIN Sunan Kalijaga Yogyakarta saat ini masih menggunakan model konvensional.

Model pembelajaran saat ini adalah sebagai berikut:

1. Asisten menerangkan di depan kemudian mahasiswa memperhatikan.
2. Editor yang digunakan adalah *Turbo C* atau *C Free*.
3. Mahasiswa dituntut untuk menyalin *source code* C yang ada di modul.
4. Mahasiswa dituntut untuk meng-*compile source code* yang sudah disalin tadi dan jika ditemukan *error* maka mahasiswa akan bertanya kepada asisten.

Kenyataan di lapangan model seperti ini tidak berjalan dengan baik, kebanyakan mahasiswa hanya datang, duduk, mendengarkan penjelasan dari asisten kemudian menjadi *tukang* ketik yaitu menyalin *source code* yang ada di modul ke dalam editor C. Mahasiswa sering tidak paham apa yang mereka tulis, bagi mereka setelah menulis *code* kemudian di-*compile* dan berjalan dengan *mulus* maka permasalahan selesai. Peneliti menilai bahwa model pembelajaran praktikum yang ada saat ini kurang efektif. Hal itu tidak hanya dirasakan oleh penulis tetapi juga para asisten Pemrograman Terstruktur dan juga dosen Pemrograman Terstruktur Teknik Informatika UIN Sunan Kalijaga.

Berbeda dengan model pembelajaran pemrograman di Negara maju. Penelitian yang dilakukan oleh Wislow menyebutkan bahwa butuh waktu 10 tahun bagi programmer pemula untuk menjadi programmer expert (Wislow, 1996). Sistem pendidikan di Negara maju sudah mengantisipasi hal tersebut, bahasa pemrograman sudah mulai dikenalkan kepada anak-anak mulai dari TK(Taman Kanak-Kanak) dan SD(Sekolah Dasar) dalam bentuk permainan puzzle, permainan logika dan sebagainya, contohnya adalah *Scratch* , *Greenfoot* , *App Inventor* , dan *Google Blockly*. Software tersebut di-design dengan tujuan menjadikan belajar memprogram itu menyenangkan, disamping itu software tersebut juga dikemas dengan begitu menarik dan pengguna hanya perlu menyusun *puzzle* untuk membuat sebuah aplikasi atau program. Software-software tersebut tidak hanya di peruntukkan untuk anak-anak TK dan SD, tapi diperuntukkan bagi siapa saja yang masih pemula dalam pemrograman. (Mafrur, 2012).

Pada studi ini peneliti akan mencoba menerapkan model pembelajaran baru pada pembelajaran praktikum mata kuliah Pemrograman Terstruktur. Model pembelajaran yang baru ini kemudian akan dibandingkan dengan model konvensional yang saat ini masih berjalan dan nantinya akan ditarik kesimpulan apakah model pembelajaran yang baru ini dapat meningkatkan pemahaman mahasiswa terhadap mata kuliah Pemrograman Terstruktur atau tidak. Model pembelajaran yang peneliti usulkan hampir mirip dengan contoh model pembelajaran yang sudah peneliti sebutkan yaitu pembelajaran menggunakan *IndoBlockly*. *IndoBlockly* adalah sebuah aplikasi *open source* berbasis web yang

dikembangkan oleh tim *IndoBlockly*. *IndoBlockly* sendiri merupakan *Google Blockly* yang oleh tim *IndoBlockly* diterjemahkan menjadi berbahasa Indonesia dan ditambahkan berbagai fitur yang mendukung dengan pendidikan di Indonesia. Rincinan model pembelajaran menggunakan *IndoBlockly* adalah sebagai berikut:

1. Editor yang digunakan adalah browser (Firefox, Chrome, Opera, Safari dll) editor *IndoBlockly* : <http://apps.developers.or.id/>
2. Mahasiswa mencoba menyelesaikan *maze (logic game)* yang ada di *IndoBlockly*
3. Mahasiswa langsung memulai membuat program dengan *IndoBlockly*
4. Mahasiswa tidak merasa seperti *coding* tetapi seperti bermain puzzle menggunakan *IndoBlockly*.
5. Program langsung bisa dijalankan dengan output dalam bentuk *dialog box javascript*
6. Blok-blok puzzle langsung bisa di-*generate* menjadi *source code C* yang langsung bisa di *compile* menggunakan *C compiler*.
7. Mahasiswa tidak disibukkan dengan *script* yang rumit (Mafrur, 2012).

1.2 Rumusan Masalah

Apakah ada perbedaan terkait dengan pemahaman mahasiswa antara menggunakan pembelajaran model konvensional dan menggunakan *IndoBlockly* pada mata kuliah Perprograman Terstruktur di Teknik Informatika UIN Sunan Kalijaga Yogyakarta.

1.3 Batasan Masalah

Batasan penelitian ini sebagai berikut:

1. Sampel yang diambil adalah mahasiswa kelas reguler semester I Teknik Informatika UIN Sunan Kalijaga tahun 2012/2013 dengan teknik *purposive sampling*.
2. Peneliti tidak melakukan proses karantina terhadap variabel kontrol maupun variabel percobaan sehingga bisa jadi ada berbagai faktor eksternal yang dapat mempengaruhi kedua variabel tersebut.
3. Peneliti tidak menghitung besaran faktor eksternal yang peneliti sebutkan pada point ke 2.
4. Proses pembelajaran dilakukan lima kali pertemuan dengan penekanan pada pemahaman input output, variabel, array, kondisi, dan perulangan.

1.4 Tujuan Penelitian

Mengetahui apakah pembelajaran menggunakan *IndoBlockly* dapat meningkatkan pemahaman (hasil belajar) mahasiswa semester I tahun 2012/2013 terhadap mata kuliah Pemrograman Terstruktur dibandingkan menggunakan model pembelajaran konvensional.

1.5 Manfaat Penelitian

Adapun manfaat yang dapat diambil dari penelitian ini adalah :

- A. Bagi Pengembang IndoBlockly

Hasil dari penelitian ini akan dijadikan landasan dasar apakah IndoBlockly ini layak untuk dikembangkan atau tidak. Jika hasil dari penelitian ini menunjukkan bahwa IndoBlockly mempunyai pengaruh yang baik terhadap pemahaman mahasiswa untuk belajar pemrograman tentu ini akan dijadikan dasar bahwa IndoBlockly memang layak untuk dikembangkan lebih lanjut.

B. Bagi Mahasiswa/Umum

Hasil Penelitian ini akan membuktikan apakah memang IndoBlockly layak menjadi tool untuk belajar pemrograman baik bagi mahasiswa atau orang umum yang ingin belajar pemrograman.

C. Bagi Peneliti

Bagi peneliti, untuk menambah pengetahuan dan wawasan agar peneliti lebih terampil dalam penelitian khususnya yang melibatkan objek manusia secara langsung dan implementasi dari sebuah software. Hal itu disebabkan karena pada umumnya jurusan teknik informatika hanya berfokus pada penelitian pengembangan sistem tidak sepenuhnya implementasi ke *end user*.

D. Bagi Peneliti Selanjutnya

Karena disini peneliti juga termasuk pengembang dari IndoBlockly tentu dengan hasil penelitian ini jika memang hasilnya menunjukkan positif bahwa IndoBlockly dapat membantu siapapun yang ingin belajar pemrograman tentu peneliti akan lebih bersemangat untuk mengembangkan IndoBlockly. Adapun jika hasilnya berkebalikan ataupun

sama saja antara menggunakan IndoBlockly dan menggunakan model konvensional maka peneliti akan berusaha untuk mencari apa penyebabnya apakah karena IndoBlockly memang belum memenuhi kriteria sebagai software yang baik atau dari kesalahan implementasi.

1.6 Keaslian Penelitian

Jenis penelitian seperti ini sudah banyak dilakukan khususnya oleh mahasiswa atau dosen dari jurusan pendidikan. Begitu juga untuk penelitian implementasi dari sebuah tool kemudian menganalisis apakah tool tersebut dapat membantu pemahaman dalam mempelajari suatu hal, penelitian semacam ini juga sudah banyak dilakukan oleh berbagai peneliti. Akan tetapi tool yang kami teliti adalah IndoBlockly, sebuah tool yang peneliti juga ikut menjadi pengembang didalamnya jadi sudah pasti penelitian ini sama sekali belum pernah dilakukan sebelumnya.

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Penelitian-penelitian sebelumnya yang relevan dengan model penelitian ini cukup banyak. Model penelitian seperti ini yaitu penelitian pengujian suatu metode baru untuk membuktikan apakah metode tersebut lebih baik dari metode yang lama sudah sering dilakukan terutama oleh kalangan pendidik. Yang berbeda dari penelitian ini adalah penelitian ini menguji software yang kami kembangkan oleh tim kami sendiri. Dari berbagai literatur yang penulis baca ada beberapa literatur yang relevan dengan penelitian kami.

Sebuah studi literatur yang berkaitan dengan psikologi dan pendidikan pemrograman yang dilihat dari persepektif seorang pengajar telah dilakukan oleh Robin dan kawan-kawannya. Tujuan penelitiannya adalah untuk mengidentifikasi permasalahan-permasalahan dalam proses belajar programer pemula. Ada beberapa hal yang membuat kami tertarik seperti yang tercantum dalam studi literturnya yaitu penelitian sebelumnya dari Guindon dalam (Robins dkk, 2003) yang menyebutkan karakter dari seorang ahli adalah memiliki skema pengetahuan efisien terorganisir dan khusus, paham dalam algoritma yang mendasar (bukan rincian dangkal seperti sintaks bahasa). Kesimpulan dari penelitian Robins dan kawan-kawannya adalah sifat yang mendasari apa yang membuat seorang pemula yang efektif? Bagaimana kita dapat mengubah efektif pemula menjadi lebih efektif?. Berbagai faktor yang berpotensi relevan yaitu motivasi, kepercayaan diri

atau tanggapan emosional aspek umum atau pengetahuan khusus, dan strategi. Jadi disamping faktor dari dalam diri programmer itu sendiri juga ada faktor dari luar yaitu strategi (strategi pengajaran) yang dapat mempercepat proses menjadikan programmer pemula menjadi programmer ahli (Robins dkk, 2003).

Selanjutnya adalah Penelitian dari Lahtinen dan kawan-kawannya. Penelitian ini merupakan survei internasional dengan responden 500 siswa dan guru. Survei ini memberikan informasi tentang kesulitan yang dirasakan ketika belajar dan mengajar pemrograman. Hasil survei juga dapat dijadikan sebagai dasar untuk rekomendasi dan *developing* bahan pembelajaran. Dalam penelitian tersebut disimpulkan bahwa kesulitan yang biasa di alami baik siswa maupun guru dalam belajar dan mengajar pemrograman tidak hanya konsep pemrograman yang abstrak yang cukup susah untuk diinterpretasikan dalam dunia nyata akan tetapi konstruksi program juga termasuk didalamnya. Penelitian tersebut dilakukan menggunakan dua bahasa pemrograman sample yaitu C++ dan Java. Jadi selain konsep pemrograman yang abstrak, susunan bahasa / konstruksi dari bahasa yang di pelajari itu juga mempengaruhi apakah bahasa itu sulit atau mudah untuk dipelajari (Lahtinen dkk, 2005).

Ketiga yaitu penelitian yang dilakukan oleh Kelleher dan Pausch. Dalam penelitiannya Kelleher dan Pausch menyajikan taksonomi bahasa dan lingkungan yang dirancang untuk membuat pemrograman lebih mudah diakses untuk programmer pemula dari segala usia. Penelitian ini menjelaskan semua kategori dalam taksonomi tersebut, memberikan gambaran singkat tentang sistem dalam setiap kategori, dan menyarankan beberapa jalan keluar untuk masa depan dalam

lingkungan pemrograman pemula dan bahasa pemrograman .Penelitian tersebut membahas satu pertanyaan pokok yaitu dalam menciptakan lingkungan pemrograman untuk pemula, salah satu pertanyaan pertama yang harus dijawab adalah mengapa siswa/mahasiswa perlu memprogram?. Ada berbagai motivasi yang mendasari mengapa siswa/mahasiswa belajar memprogram yaitu : pemrograman sebagai karir, memprogram untuk belajar bagaimana memecahkan masalah dengan cara yang terstruktur dan logis, untuk membangun perangkat lunak untuk keperluan pribadi, untuk mengeksplorasi ide-ide dalam bidang studi lainnya, dan lain sebagainya.

Kelleher dan Pausch menyebutkan bahwa taksonomi atau urutan tertinggi pada lingkungan pemrograman/*programming environment* adalah *teaching system* (sistem pengajaran) dan *Empowering Systems* (pemberdayaan sistem) kemudian disusul dengan berbagai hirarki taksonomi dibawahnya. *Teaching system* dan *Empowering Systems* sebagai hirarki yang paling tinggi, jelas bagaimana seorang dapat memahami pemrograman jika dia tidak belajar, dan bagaimana dia belajar jika tidak diajar. Kemudian disusul hirarki seperti *Mechanics of programming* (sisi teknis dari pemrograman) dan seterusnya. Dalam tulisannya juga disebutkan berbagai bahasa pemrograman yang dapat mempermudah siapapun yang ingin belajar pemrograman dan dapat digunakan dari segala usia. Disamping itu dia juga menyebutkan berbagai kendala dan hambatan bagi siapa saja yang belajar bahasa pemrograman. Kendala tersebut ada dua yaitu kendala teknis dan kendala sosial.

Berbagai kendala teknis seperti konsep yang abstrak, sintaks yang susah untuk dihafal dan sebagainya, untuk kendala teknis ini sudah banyak dibahas oleh banyak peneliti. Beberapa peneliti mengajukan konsep dan design seperti pemrograman berbasis visual, game, dan sebagainya. Begitu juga dengan penelitian Kelleher dan Pausch ini mereka memberikan banyak contoh-contoh bahasa pemrograman visual yang dapat mempermudah pemula dalam memahami bahasa pemrograman. Kendala yang kedua yaitu kendala sosial. Analisis Kelleher dan Pausch menyebutkan bahwa ternyata tidak hanya kendala teknis yang dihadapi oleh programmer pemula yang ingin belajar pemrograman tapi juga ada kendala sosial. Contoh kendala sosial adalah siswa/mahasiswa memutuskan dan memilih untuk tidak belajar pemrograman. Kelleher dan Pausch mengatakan bahwa kendala sosial ini lebih sulit untuk diselesaikan daripada kendala teknis. Solusi dari kendala sosial ini adalah sosialisai dan dukungan sosial. Kelleher dan Pausch menyimpulkan beberapa solusi yang bisa digunakan untuk mengatasi kendala-kendala baik itu teknis maupun sosial dalam belajar pemrograman yaitu: menyederhanakan mekanisme pemrograman, memberikan dukungan bagi pelajar, dan memotivasi siswa untuk belajar memprogram. Dalam hal mengatasi hambatan sosial bisa dengan mendukung peserta didik atau memberikan alasan yang menarik untuk belajar pemrograman (Kelleher dan Pausch, 2003).

Sesuai dengan penelitian Kelleher dan Pausch ini respon dari penulis adalah bahwa kendala-kendala itu memang benar-benar nyata. Hal tersebut bisa penulis utarakan berdasarkan pengalaman penulis sendiri baik ketika penulis belajar pemrograman maupun pengalaman penulis ketika mengajar pemrograman.

Kendala teknis adalah kendala yang akan pertama dihadapi oleh siapapun yang ingin belajar pemrograman, oleh karena itu penulis mengusulkan sebuah konsep dan design yang sebenarnya sudah banyak diusulkan dan diimplementasikan sejak dahulu akan tetapi belum di implementasikan dalam lingkungan penulis yaitu VPLs(*Visual Programming Languages*).

Penelitian yang relevan ke empat yaitu paper dari scratch yang dilakukan oleh Maloney dan kawan-kawannya. Maloney dan kawan-kawannya ini menjelaskan scratch yaitu bahasa pemrograman visual blok yang dirancang dan difasilitasi berbagai media untuk programmer pemula. Pada penelitian ini Maloney dan kawan-kawannya membentuk sebuah Clubhouse yang di peruntukkan bagi siapa saja untuk datang ke Clubhouse. Penelitian tersebut dilakukan selama periode 18 bulan, setelah periode 18 bulan kemudian Maloney menganalisis 34 % atau sebanyak 536 proyek-proyek scratch yang ada di Clubhouse tadi. Dalam paper tersebut juga dibahas mengenai motivasi pemuda perkotaan yang memilih untuk memprogram menggunakan scratch daripada menggunakan salah satu dari banyak paket perangkat lunak lain yang tersedia bagi mereka. Ada yang menurut penulis cukup menarik disini. Pada tulisan Maloney dkk di jelaskan sebagai berikut: Pertanyaan yang lebih mendesak adalah mengapa pemuda memilih untuk terlibat dalam memprogram menggunakan scratch di Scratch Clubhouse mengingat bahwa mereka memiliki banyak pilihan perangkat lunak lain? Jawaban terbaik mungkin sudah dijawab oleh Kelleher dan Pausch yang mencatat bahwa sistem dapat membuat pemrograman menjadi lebih mudah diakses oleh pemula dan dikarenakan scratch dapat menyederhanakan mekanisme pemrograman,

dengan memberikan dukungan bagi pelajar, dan dengan memberikan siswa dengan motivasi untuk belajar memprogram.

Maloney menyatakan kami berpikir bahwa scratch mempunyai tiga hal yang menarik. Yang pertama, desain blok scratch menyederhanakan mekanisme pemrograman dengan meminimalisir kesalahan sintaks, scratch langsung dapat memberikan umpan balik mengenai penempatan blok , dan scratch langsung dapat memberikan umpan balik untuk percobaan/ketika program dijalankan. Selain itu, kita berpikir bahwa infrastruktur sosial Komputer Clubhouse penting dalam memberikan dukungan bagi programer pemula. Jawaban tersebut jelas masuk akal karena di scratch Clubhouse kami juga menyediakan beberapa mentor yang bisa mengajari para programer pemula, dan juga mungkin dengan seringnya mereka berada di scratch Clubhouse mereka bisa bertukar program dan ide dengan teman-teman mereka (Maloney dkk, 2008).

Dari beberapa penelitian yang relevan dengan penelitian ini yang sudah penulis sampaikan diatas yang paling mendekati dengan penelitian yang penulis lakukan ini adalah penelitian ke empat yang dilakukan oleh Maloney dan kawan-kawannya. Tentu banyak yang mengetahui tentang scratch dan begitu mudahnya memprogram dengan menggunakan scratch. Penelitian yang penulis lakukan jelas masih jauh dari penelitian ini, kalau penelitian yang dilakukan Maloney dan kawan-kawannya memang akan memberikan hasil yang general karena sampel yang digunakan berupa proyek-proyek general dari berbagai anak/siswa yang sering *nongkrong*/bermain di scratch ClubHouse. Proyek scatch memang tidak main-main scratch berani membuat sebuah ClubHouse demi untuk kelangsungan

dan analisis penelitian proyek scratch itu sendiri dan disini inilah yang belum bisa penulis lakukan. Semoga pada penelitian IndoBlockly berikutnya akan lebih baik.

2.2 Landasan Teori

Landasan teori menjelaskan berbagai teori yang berhubungan dengan penelitian yang dilakukan oleh penulis baik yang berhubungan dengan pendidikan dan pembelajaran, kemudian pemrograman dan juga software yang digunakan sebagai objek penelitian.

2.2.1 Pengertian Belajar

Belajar adalah suatu aktivitas mental/psikis yang berlangsung dalam interaksi aktif dengan lingkungan yang menghasilkan perubahan dalam pengetahuan, pemahaman, keterampilan dan nilai sikap (Winkel, 1983). Sedangkan menurut (Gagne dalam Dahar, 1989) belajar adalah suatu proses di mana suatu organisme berubah perilakunya sebagai akibat dari pengalaman. Dengan belajar tindakan perilaku siswa akan berubah ke arah yang lebih baik. Berhasil baik atau tidaknya belajar tergantung dari faktor-faktor yang mempengaruhinya. Faktor-faktor tersebut terdiri dari faktor internal, eksternal dan pendekatan belajar.

1. Faktor internal adalah faktor dari dalam diri siswa, yaitu keadaan/kondisi jasmani dan rohani siswa meliputi aspek fisiologis (kondisi tubuh dan panca indera), dan aspek psikologis antara lain: intelegensi dalam, sikap misalnya dalam beradaptasi dengan teman, bakat dalam mengerjakan soal, minat dalam mengikuti pelajaran serta punya kemauan besar untuk belajar

dan mempunyai motivasi untuk belajar baik individu maupun dalam kelompok.

2. Faktor eksternal adalah faktor dari luar diri siswa, yaitu kondisi lingkungan di sekitar siswa meliputi faktor lingkungan sosial (guru, teman, masyarakat, dan keluarga) dan faktor lingkungan non-sosial (gedung, sekolah, tempat tinggal, alat belajar, cuaca dan waktu belajar) .

2.2.2 Media Pembelajaran

Kata media berasal dari bahasa latin *Medius* yang secara harfiah berarti '*perantara*' atau '*pengantar*'. Menurut Boove (dalam Ena, 2007), media adalah sebuah alat yang berfungsi untuk menyampaikan pesan pembelajaran. Bentuk-bentuk stimulus yang bisa dipergunakan sebagai media diantaranya adalah hubungan atau interaksi manusia, realita, gambar bergerak atau tidak, tulisan dan suara yang direkam (Ena, 2007).

Menurut Gerlach dan Ely (dalam Arsyad, 2002), ciri-ciri media ada tiga, yaitu.

1. Ciri Fiksatif (Fixative Property)

Ciri ini menggambarkan kemampuan media merekam, menyimpan, melestarikan, dan merekonstruksi suatu peristiwa atau objek.

2. Ciri Manipulatif (Manipulatif Property)

Transformasi suatu kejadian atau objek dimungkinkan karena media memiliki ciri manipulatif. Kejadian yang memakan waktu berhari-hari dapat disajikan kepada siswa dalam waktu dua atau tiga menit dengan teknik pengambilan gambar time-lapse recording.

3. Ciri Distributif (Distributive Property)

Ciri distributif dari media memungkinkan suatu objek atau kejadian ditransportasikan melalui ruang, dan secara bersamaan kejadian tersebut disajikan kepada sejumlah besar siswa dengan stimulus pengalaman yang relatif sama mengenai kejadian itu.

Fungsi utama media pembelajaran adalah sebagai alat bantu mengajar yang ikut mempengaruhi iklim, kondisi, dan lingkungan belajar yang ditata dan diciptakan oleh guru. Sedangkan manfaat penggunaan media (Arsyad, 2002), antara lain:

1. Media pembelajaran dapat memperjelas penyajian pesan dan informasi sehingga dapat memperlancar dan meningkatkan proses dan hasil belajar.
2. Media pembelajaran dapat meningkatkan dan mengarahkan perhatian anak sehingga dapat menimbulkan motivasi belajar, interaksi yang lebih langsung antara siswa dan lingkungannya, dan kemungkinan siswa untuk belajar sendiri-sendiri sesuai dengan kemampuan dan minatnya.
3. Media pembelajaran dapat mengatasi keterbatasan indera, ruang, dan waktu.
4. Media pembelajaran dapat memberikan kesamaan pengalaman kepada siswa tentang peristiwa-peristiwa di lingkungan mereka, serta memungkinkan terjadinya interaksi langsung.

2.2.3 Proses Pembelajaran

Penilaian adalah upaya atau tindakan untuk mengetahui sejauh mana tujuan yang telah ditetapkan itu tercapai atau tidak. Penilaian berfungsi sebagai

alat untuk mengetahui keberhasilan proses dan hasil belajar siswa/mahasiswa. Dalam sistem pendidikan nasional rumusan tujuan pendidikan, baik tujuan kurikuler maupun tujuan instruksional, menggunakan klasifikasi hasil belajar dari Benyamin Bloom yang secara garis besar membaginya menjadi tiga ranah, yakni ranah kognitif, ranah afektif, dan ranah psikomotorik (Bloom, 1956).

Salah satu prinsip dasar yang harus senantiasa diperhatikan dalam pendidikan adalah melaksanakan evaluasi hasil belajar secara menyeluruh terhadap peserta didik, baik dari segi pemahamannya terhadap materi atau bahan ajar yang telah diberikan (aspek kognitif), maupun dari segi penghayatan (aspek afektif), dan pengamalannya (aspek psikomotor). Benjamin S. Bloom dan kawan-kawannya itu berpendapat bahwa pengelompokkan tujuan pendidikan itu harus senantiasa mengacu kepada tiga jenis domain (ranah) yang melekat pada diri peserta didik, yaitu:

- a) Ranah proses berfikir (*cognitive domain*)
- b) Ranah nilai atau sikap (*affective domain*)
- c) Ranah keterampilan (*psychomotor domain*)

Dalam konteks evaluasi hasil belajar, maka ketiga domain atau ranah itulah yang harus dijadikan sasaran dalam setiap kegiatan evaluasi hasil belajar. Sasaran kegiatan evaluasi hasil belajar adalah:

1. Apakah peserta didik sudah dapat memahami semua bahan atau materi pelajaran yang telah diberikan pada mereka?

2. Apakah peserta didik sudah dapat menghayatinya?
3. Apakah materi pelajaran yang telah diberikan itu sudah dapat diamalkan secara kongkret dalam praktek atau dalam kehidupannya sehari-hari?

Diantara ketiga ranah tersebut, ranah kognitiflah yang paling banyak dinilai oleh para dosen /guru karena berkaitan dengan kemampuan para siswa dalam menguasai isi bahan pengajaran.

2.2.3.1 Pengertian Ranah Penilaian Kognitif

Ranah kognitif adalah ranah yang mencakup kegiatan mental (otak). Menurut Bloom, segala upaya yang menyangkut aktivitas otak adalah termasuk dalam ranah kognitif. Ranah kognitif berhubungan dengan kemampuan berfikir, termasuk didalamnya kemampuan menghafal, memahami, mengaplikasi, menganalisis, mensintesis, dan kemampuan mengevaluasi. Dalam ranah kognitif itu terdapat enam aspek atau jenjang proses berfikir, mulai dari jenjang terendah sampai dengan jenjang yang paling tinggi. Keenam jenjang atau aspek yang dimaksud adalah:

a. Pengetahuan/hafalan/ingatan (*knowledge*)

Adalah kemampuan seseorang untuk mengingat-ingat kembali (*recall*) atau mengenali kembali tentang nama, istilah, ide, rumus-rumus, dan sebagainya, tanpa mengharapkan kemampuan untuk menggungkannya. Pengetahuan atau ingatan adalah merupakan proses berfikir yang paling rendah. Salah satu contoh hasil belajar kognitif pada jenjang pengetahuan adalah dapat menghafal jenis-jenis variabel yang biasa digunakan dalam memprogram dsb.

b. Pemahaman (*comprehension*)

Adalah kemampuan seseorang untuk mengerti atau memahami sesuatu setelah sesuatu itu diketahui dan diingat. Dengan kata lain, memahami adalah mengetahui tentang sesuatu dan dapat melihatnya dari berbagai segi. Seseorang peserta didik dikatakan memahami sesuatu apabila ia dapat memberikan penjelasan atau memberi uraian yang lebih rinci tentang hal itu dengan menggunakan kata-katanya sendiri. Pemahaman merupakan jenjang kemampuan berfikir yang setingkat lebih tinggi dari ingatan atau hafalan. Salah satu contoh hasil belajar ranah kognitif pada jenjang pemahaman ini adalah mahasiswa bisa paham terhadap algoritma searching kemudian dapat menguraikannya kembali dengan bahasa mereka sendiri.

c. Penerapan (*application*)

Adalah kesanggupan seseorang untuk menerapkan atau menggunakan ide-ide umum, tata cara ataupun metode-metode, prinsip-prinsip, rumus-rumus, teori-teori dan sebagainya, dalam situasi yang baru dan kongkret. Penerapan ini adalah merupakan proses berfikir setingkat lebih tinggi ketimbang pemahaman. Salah satu contoh hasil belajar kognitif jenjang penerapan adalah mahasiswa mampu menerapkan algoritma yang sudah ia pahami ke dalam bentuk code dengan bahasa pemrograman yang ia kuasai.

d. Analisis (*analysis*)

Adalah kemampuan seseorang untuk merinci atau menguraikan suatu bahan atau keadaan menurut bagian-bagian yang lebih kecil dan mampu

memahami hubungan di antara bagian-bagian atau faktor-faktor yang satu dengan faktor-faktor lainnya. Jenjang analisis adalah setingkat lebih tinggi ketimbang jenjang aplikasi. Contoh dari hasil belajar kognitif jenjang analisis adalah mahasiswa mampu menganalisis algoritma dan *source code*, misalnya terjadi error ketika di jalankan maka mahasiswa dalam tahap analisis ini harus mampu menyelesaikan permasalahan tersebut (*debugging*) sampai program bisa benar-benar berjalan dengan baik.

e. Sintesis (*syntesis*)

Adalah kemampuan berfikir yang merupakan kebalikan dari proses berfikir analisis. Sintesis merupakan suatu proses yang memadukan bagian-bagian atau unsur-unsur secara logis, sehingga menjelma menjadi suatu pola yang berstruktur atau bebrbentuk pola baru. Jenjang sintesis kedudukannya setingkat lebih tinggi daripada jenjang analisis. Contoh: mahasiswa tidak hanya mampu memahami algoritma kemudian menuliskan dalam *source code* dan juga melakukan *debugging* tetapi mahasiswa juga mampu untuk memodifikasi atau membuat algoritma baru.

f. Penilaian (*evaluation*)

Penilaian merupakan jenjang berpikir paling tinggi dalam ranah kognitif dalam taksonomi Bloom. Penilaian/evaluasi disini merupakan kemampuan seseorang untuk membuat pertimbangan terhadap suatu kondisi, nilai atau ide, misalkan jika seseorang dihadapkan pada beberapa pilihan maka ia akan mampu memilih satu pilihan yang terbaik sesuai dengan patokan-patokan atau kriteria

yang ada. Salah satu contoh hasil belajar kognitif jenjang evaluasi adalah: mahasiswa mampu untuk menilai dan memilih algoritma yang paling efektif yang akan dia gunakan. Keenam jenjang berpikir yang terdapat pada ranah kognitif menurut Taksonomi Bloom itu, jika diurutkan secara hirarki piramidal adalah sebagai tertulis pada Gambar 2.1.



Gambar 2.1 Hirarki Piramida Taksonomi Bloom

2.2.3.2 Ranah Afektif dan Psikomotorik

Pemrograman komputer memang lebih banyak berkaitan dengan ranah kognitif akan tetapi bukan berarti kedua ranah atau domain yang lain itu dihilangkan. Ranah afektif adalah ranah penghayatan kemudian disusul dengan ranah psikomotorik atau ranah skill. Proses belajar pemrograman itu semua berada pada ranah kognitif dimulai dari menghafal sintaks, memahami,

menganalisis kesalahan, mensintesis algoritma baru, menilai dan memilih algoritma yang paling efektif dan sebagainya.

Ranah afektif akan ada ketika mahasiswa yang tadinya dia tidak mengetahui apa-apa mengenai pemrograman, kemudian dia sudah melalui ranah kognitif dan akhirnya jiwa programmer secara tidak sadar akan terbentuk di dalam diri mahasiswa tersebut. Ketika penghayatan sebuah algoritma dan juga penghayatan dalam menulis kode dalam proses selanjutnya ini akan menuji ranah psikomotorik yaitu ranah skill. Ranah skill adalah ranah dimana yang tadinya hanya seorang mahasiswa atau programmer pemula akan mempunyai skill yang cukup hebat yang menjadikan dirinya menjadi programmer ahli.

2.2.4 Belajar Pemrograman (*Learning Programming*)

Pemrograman adalah keterampilan yang sangat berguna dan dapat menjadi karir. Baru-baru ini kebutuhan akan permintaan terhadap programmer dan minat siswa dalam pemrograman berkembang pesat, dan kursus pemrograman telah menjadi semakin populer. Belajar pemrograman membutuhkan usaha yang keras bagi seorang programmer pemula biasanya mengalami berbagai kesulitan. Kursus/Sekolah Pemrograman umumnya dianggap sulit, dan sering memiliki risiko tingkat putus sekolah paling tinggi (Winslow, 1996). Dalam penelitian disebutkan bahwa dibutuhkan sekitar 10 tahun untuk mengubah programmer pemula menjadi programmer ahli (Robins dkk, 2003).

Pemahaman dalam memprogram merupakan bagian penting dari keterampilan pemrograman komputer, baik dari yang praktis, perspektif maupun

teoritis. Ini adalah sebuah keterampilan kognitif yang kompleks, yang melibatkan akuisisi representasi mental dari struktur program dan fungsi. Dari sudut pandang Theoretical , pemahaman melibatkan penugasan makna tertentu yaitu sesuatu yang membutuhkan pengetahuan khusus. Dari sudut pandang praktis, kemampuan untuk memahami program yang ditulis oleh orang lain atau ketika membuat program sendiri adalah komponen penting keahlian seorang programmer. Skill/keahlian pemahaman ini akan digunakan programmer untuk melakukan tugas-tugas pemrograman seperti *debugging*, *editing* dan, *code reuse*. (Navarro-Prieto, 2000).

Pennington Memperkirakan bahwa lebih dari 50% dari semua waktu programmer profesional dihabiskan untuk tugas-tugas pemeliharaan program yang melibatkan modifikasi/editing dan update dari program yang sebelumnya ditulis. Dari berbagai hal diatas dapat kita simpulkan bahwa pemahaman memainkan peran sentral dalam pemrograman. Oleh karena itu, penelitian tentang strategi pemahaman akan sangat berguna karena hasilnya dapat memberikan informasi yang berguna untuk meningkatkan kinerja programmer, kemajuan pendidikan, kemajuan teknologi desain dan lingkungan pemrograman (*programming environments*) (Pennington, 1987a).

Dari berbagai referensi diatas kita dapat menyimpulkan bahwa belajar pemrograman itu memang susah terutama bagi programmer pemula/ mahasiswa yang sedang belajar pemrograman. Seperti penelitian yang dilakukan oleh Wislow bahwa bagi programmer pemula butuh 10 tahun untuk menjadi programmer yang ahli. Untuk menjadi programmer yang baik, mahasiswa membutuhkan skill dan

usaha yang keras untuk menghafalkan sintak dan memahami aturan belajar pemrograman itu sendiri (Esteves dan Mendes, 2004). Mahasiswa harus tahu dan paham bagaimana menyelesaikan suatu masalah dan belajar membuat dan merinci algoritma yang efisien itulah yang menyebabkan belajar pemrograman terkesan sulit. Pada Kenyataanya banyak mahasiswa yang gagal dan menemui masalah ini pada awal pelajaran/mata kuliah pemrograman untuk pertama kalinya. Sudah banyak peneliti yang meneliti masalah ini (Lahtinen dkk, 2005). Salah satu penyebab yang disebutkan adalah mahasiswa tidak siap untuk berfikir dan menyelesaikan masalah. Pada proses belajar mengajar sebelumnya mahasiswa tidak terbiasa untuk menyelesaikan suatu masalah/kasus, sehingga mahasiswa tidak mempunyai kemampuan problem solving yang baik (Gomes, 2006). Beberapa mahasiswa yang cukup mampu menyelesaikan masalah dan membuat algoritma biasanya juga terkendala dalam masalah waktu penyelesaian.

Penyebab lain dari kegagalan belajar pemrograman adalah mahasiswa sudah mengatakan bahwa dia tidak menyukai pemrograman dikarenakan konsepnya susah untuk dipahami, seperti variabel, tipe data, alamat memori dikarenakan itu adalah konsep yang abstrak yang tidak bisa direpresentasikan di dunia nyata (Miliszewska, 2007). Penyebab lain adalah proses pembelajaran yang cenderung masih tradisional yaitu dosen hanya mengajarkan sintak bahasa pemrograman tertentu dan tidak matang dalam hal pemahaman konsep (Lethbridge, 2007). Dengan kemajuannya teknologi dan perkembangan software yang begitu pesat beberapa peneliti mengusulkan design dan software pemrograman berbasis visual dan memang kenyataanya sekarang banyak

software-software yang diciptakan dengan tujuan untuk mengatasi masalah-masalah diatas. Memang untuk saat ini masih dominan software pemrograman yang berupa text based, tetapi ada juga yang visual based bahkan ada yang berbentuk game (Hundhausen, 2007).

2.2.5 Bahasa Pemrograman Visual (*Visual Programming Language /VPLs*)

Dari mulai lukisan di gua hieroglif sejak dulu manusia sudah lama berkomunikasi satu dengan lainnya menggunakan gambar. Banyak para peneliti di bidang VPL bertanya: Mengapa kemudian kita saat ini berkomunikasi dengan komputer menggunakan pemrograman tekstual? Bukankah akan lebih produktif jika dapat menginstruksikan komputer hanya menggunakan gambar.? Jelas para pendukung VPLs menjawab ya. Pertanyaan-pertanyaan diatas menjadi motivasi utama para peneliti di bidang VPLs. Pertama, banyak orang berpikir dan mengingat hal hal dari segi gambar. Selain itu pemrograman tekstual telah terbukti menjadi masalah bagi orang-orang kreatif dan cerdas , mereka harus banyak belajar terlebih dahulu mengenai sintak-sintak tekstualnya dari bahasa pemrograman itu sendiri. Permasalahan yang lain adalah ide lebih mudah diterjemahkan dalam bentuk gambar visual atau grafik akan tetapi kita harus menerjemahkan ide tersebut kedalam bentuk tekstual.Selain itu banyak bidang-bidang ilmu yang memerlukan simulasi dalam bentuk visual dan tidak dapat di interpretasikan dalam bentuk tekstual. Dari berbagai alasan diatas itu membuktikan bahwa penelitian mengenai VPLs itu menarik dan penting (Boshernitsan dan Downes, 2004).

2.2.5.1 Kelebihan dan Kekurangan Bahasa Pemrograman Visual

Pemrograman grafis/visual memiliki kelebihan dan kekurangan dibandingkan dengan pemrograman tekstual. Landasan teroi mengenai bahasa pemrograman visual ini kami mengambil dari tulisan.

2.2.5.1.1 Kelebihan VPLs

Andrew Begel mengatakan bahwa dengan menggunakan representasi grafis dari benda-benda anda dapat lebih konkret merepresentasikan suatu objek (klik dua kali pada objek koper dan melihat apa yang ada di dalam), menghilangkan sintak-sintak yang susah untuk dipahami (seperti { } dan () dalam C, BEGIN dan END dan ()dalam Pascal, dan () dalam Lisp) (Begel, 1996).

Pemrograman grafis/visual juga dapat dikatakan sebagai metafora dari kehidupan nyata untuk membuat pemrograman lebih mudah. Sebagai contoh, pemrograman tombol lampu untuk menghidupkan dan mematikan lampu. Pemrograman grafis juga memungkinkan lebih mudah untuk berbagi program. Anda dapat mendefinisikan program Anda menjadi blok tertentu dan hanya "memberikan" blok ke teman untuk mencoba sendiri. Hal ini mirip dengan manfaat yang diusulkan desain OOP dalam bahasa tekstual. Keuntungan lain adalah mudah dipahami. Melihat gambar dari sebuah program, pengguna dapat lebih mudah membedakan maknanya, daripada melihat program tekstual yang

terdiri dari file dan kode yang sangat banyak. Mungkin salah satu keuntungan terbaik adalah penggunaan isyarat visual dalam bahasa grafis/visual. Sambungan/koneksi antar objek dapat dibuat lebih eksplisit melalui desain dan grafis daripada menggunakan tekstual.

2.2.5.2 Kekurangan VPLs

Disamping ada kelebihan di sisi lain juga ada kerugian dari VPLs. Beberapa bahasa grafis/visual yang dapat menyebabkan frustrasi bagi programmer expert yang ingin mengungkapkan pernyataan yang mungkin lebih baik dan efektif diwakili menggunakan teks. Selain itu Masalah pemrograman visual adalah bahwa Anda tidak dapat memiliki lebih dari 50 primitif visual pada layar pada saat yang sama. "Pertanyaan selanjutnya adalah Bagaimana Anda akan menulis sebuah sistem operasi dengan pemrograman visual? "Masalah lain adalah extendibility bahasa. C dan Lisp dibangun untuk dikembangkan oleh para programmer. Pada saat ini, bahasa grafis cenderung terbatas pada pembuat desain tanpa berpikir untuk menambahkan fitur tambahan.

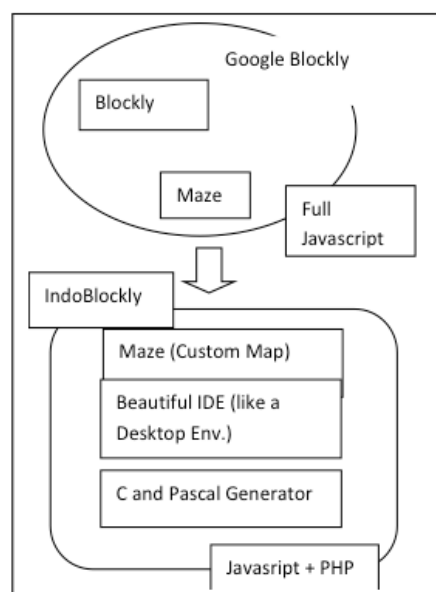
Ada daya tarik yang sangat tinggi dari Pemrograman Visual yaitu untuk anak-anak. Anak-anak suka memanipulasi blok dan mengumpulkan koleksi benda. seringkali masalah utama untuk mengajarkan pemrograman kepada anak-anak adalah bahwa sintak-sintak pemrograman yang sulit dipahami. VPLs menghilangkan masalah ini, anak-anak akan lebih mudah untuk membuat program. Akan tetapi Sebuah bahasa grafis/visual yang terlalu sederhana dapat mengurangi kreatifitas seorang anak untuk membuat program/sesuatu yang

menyenangkan. Ketika anak bertambah usia dan lebih cerdas, ia dapat bermigrasi ke bahasa tekstual dan menggunakan kompleksitas yang lebih besar (Begel, 1996).

2.2.6 IndoBlockly

IndoBlockly mirip dengan bahasa visual block sebelumnya seperti *scratch*, *Greenfoot*, *AppInventor*, dan *Google Blockly*. Dengan menggunakan IndoBlockly pengguna tidak akan merasa coding tetapi yang dirasakan adalah menyusun puzzle. “*Semua bisa jadi programmer dengan IndoBlockly*”, itu adalah slogan IndoBlockly, IndoBlockly dilengkapi dengan generator bahasa C dan Pascal, blok-blok yang sudah disusun oleh pengguna langsung dapat diubah menjadi source code C atau Pascal (Mafrur, 2012).

2.2.6.1 Konsep IndoBlockly



Gambar 2.2 . Konsep IndoBlockly

Konsep IndoBlockly adalah membuat sebuah software dengan kriteria sebagai berikut :

1. Berbahasa Indonesia.
2. Menarik, tidak membosankan, user tidak terasa seperti coding tetapi seperti bermain.
3. Meminimalisir penggunaan sintak-sintak yang susah untuk dimengerti oleh pengguna baru.
4. IDE yang portabel (cloud/web based).
5. IDE yang mempunyai tampilan seperti IDE Desktop Environment.
6. Hasilnya dapat dikonvert ke dalam source code C atau Pascal dan langsung dapat di eksekusi dengan C atau Pascal compailer
7. Format file penyimpanan yang portabel.
8. Ada beberapa game logika untuk mengasah kemampuan otak anak.
9. Dapat meningkatkan pemahaman user terhadap konsep pemrograman dan algoritma.

Pada awal tahun 2012 Google meluncurkan *Google Blockly* dengan lisensi opensource. Kemudian kami bersepakat untuk mengambil source dari *Google Blockly* kemudian kami modifikasi sesuai dengan konsep awal tadi. Nama IndoBlockly juga diambil dari *Google Blockly*, Indo adalah Indonesia dan Blokly adalah blok-blok puzzle dari *Google Blockly*. Konsep IndoBlockly dalam bentuk gambar dapat dilihat pada Gambar 2.2.

2.2.6.2 Design Interface IndoBlockly

Saat ini IndoBlockly belum sepenuhnya mendukung semua fitur-fitur yang ada dalam tool bahasa pemrograman. Oleh sebab itu pada batasan masalah peneliti juga membatasi bahwa yang penulis teliti adalah hasil belajar pemrograman terstruktur meliputi pemahaman terhadap input output, variabel, array, kondisi dan perulangan. Pada pembahasan ini kami menyertakan screenshot dari aplikasi indoBlockly.

1. Halaman Indeks IndoBlockly

Gambar 2.2 adalah gambar tampilan awal tool IndoBlockly yang dapat di akses di <http://apps.developers.or.id> .

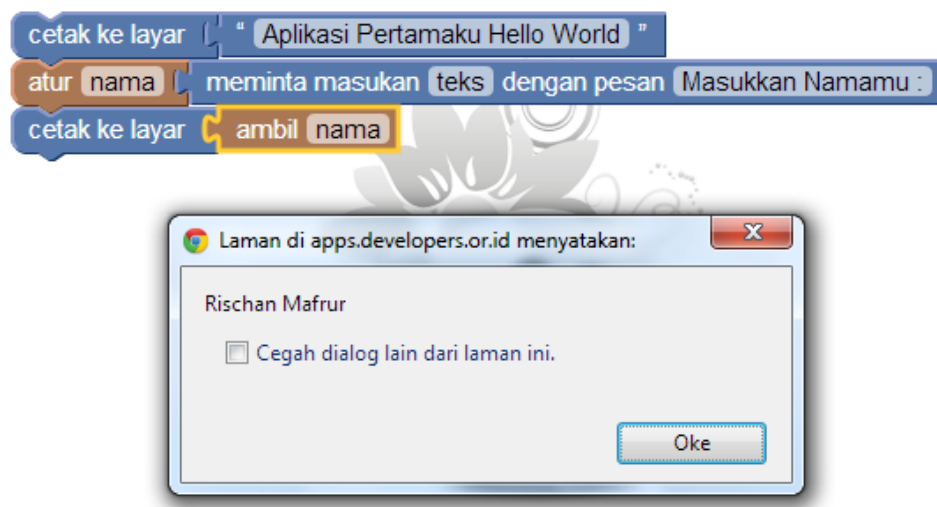


Gambar 2.2 Halaman Indeks aplikasi IndoBlockly

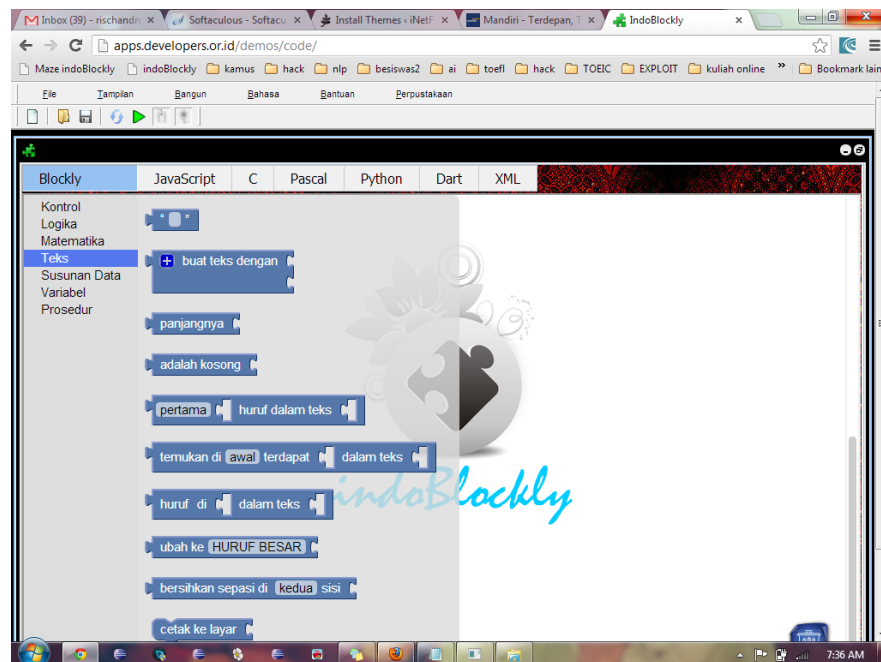
2. Input output

Pemrograman C menggunakan *scanf* untuk proses input dan *printf* untuk output sedangkan C++ menggunakan *cin* dan *cout*. Pada

IndoBlockly user tinggal menekan menu text untuk mengambil puzzle cetak ke layar untuk output dan atur variabel kemudian meminta masukkan teks dengan pesan . Penggunaan input output pada IndoBlockly dapat dilihat pada Gambar 2.3 dan 2.4.



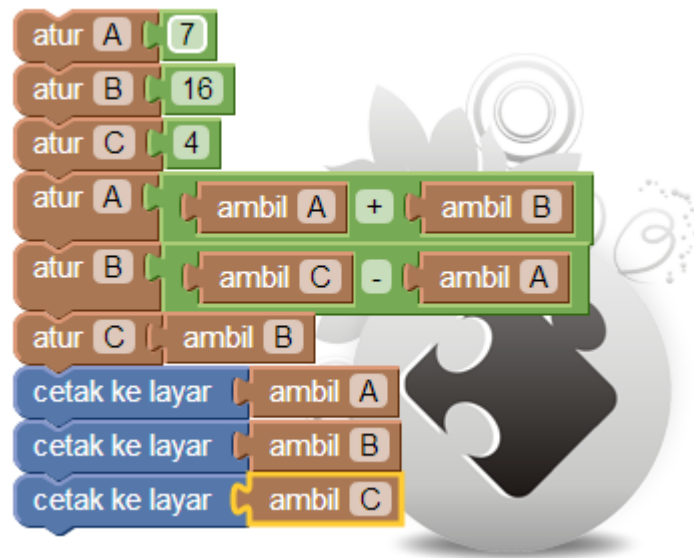
Gambar 2.3 Input output sederhana menggunakan IndoBlockly



Gambar 2.4 Menu Teks pada IndoBlockly

3. Variabel

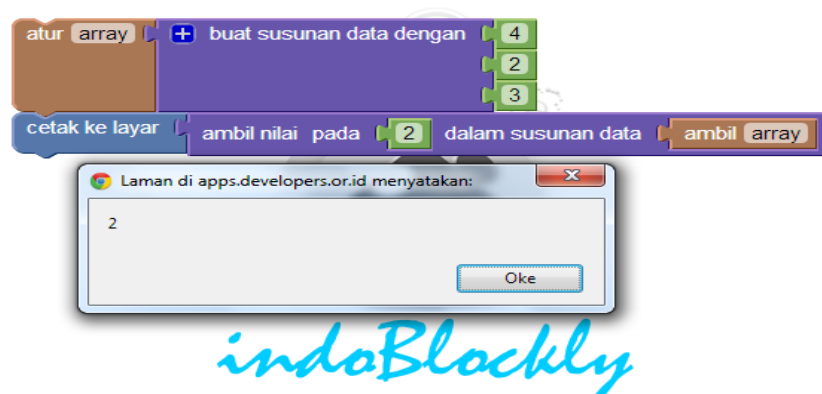
Pembuatan variabel pada IndoBlockly seperti pada pemrograman orientasi objek yaitu dengan menggunakan *set* dan *get* atau “atur” dan “ambil”. Tidak ada deklarasi seperti *int*, *string* dsb. Contoh penggunaan variable dapat dilihat pada Gambar 2.5.



Gambar 2.5 Contoh penggunaan variabel pada IndoBlockly

4. Array

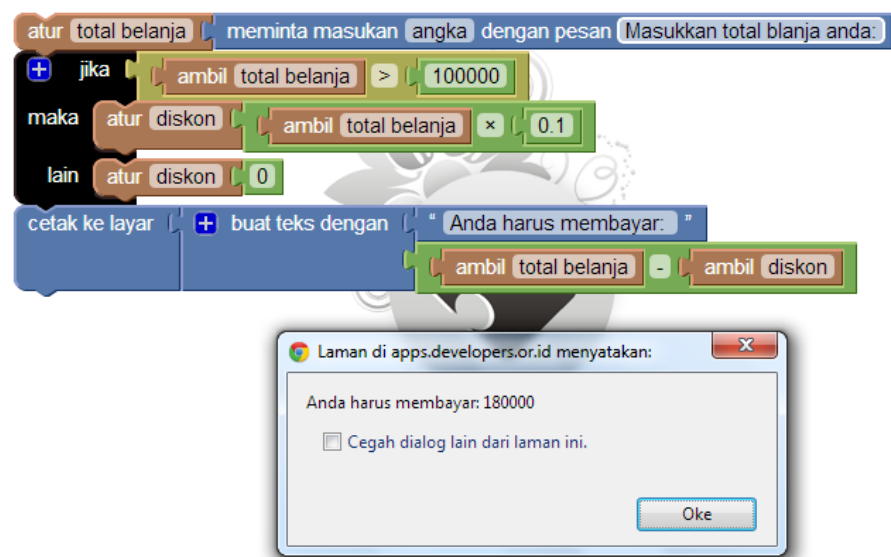
Gambar 2.6 adalah contoh penggunaan array menggunakan IndoBlockly.



Gambar 2.6 Contoh array di IndoBlockly

5. Kondisi

Biasanya pada beberapa bahasa pemrograman untuk sebuah kondisi itu menggunakan *if* atau *case*. IndoBlockly belum mendukung *case* jadi pada IndoBlockly hanya dapat menggunakan *if*. Contoh penggunaan *if* pada IndoBlockly dapat dilihat pada Gambar 2.7.



Gambar 2.7 Penggunaan if pada IndoBlockly

6. Perulangan

Perulangan pada IndoBlockly dapat menggunakan *for*, *while* dan *do while*. Gambar 2.8 adalah contoh penggunaan perulangan *for* menggunakan IndoBlockly.



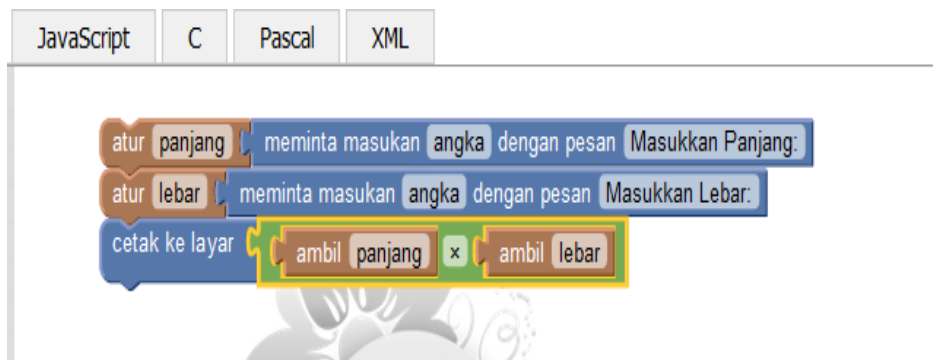
Gambar 2.8 Penggunaan for pada IndoBlockly

2.2.6.3 Contoh Pembuatan Program Sederhana menggunakan IndoBlockly

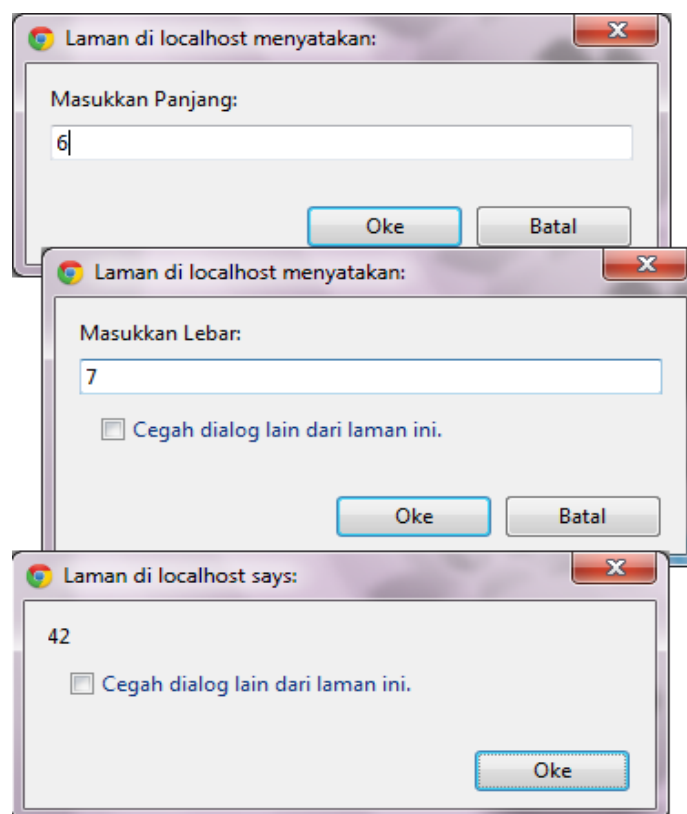
Pada pembahasan ini ada beberapa contoh program sederhana menggunakan IndoBlockly. Gambar 2.9 adalah contoh program untuk mencari nilai maksimum menggunakan IndoBlockly. Gambar 2.10 adalah program sederhana perhitungan luas persegi panjang. Fitur yang lain dari IndoBlockly adalah user dapat mengkonvert code puzzle IndoBlockly langsung menjadi source code C. Pada Gambar 2.11 adalah gambar hasil output aplikasi sederhana perhitungan persegi panjang menggunakan IndoBlockly dan Gambar 2.12 adalah gambar hasil konvert *source code* aplikasi persegi panjang menjadi *source code* C. Dan Gambar 2.13 adalah gambar hasil running program menggunakan C Free dari aplikasi sederhana perhitungan luas persegi panjang.



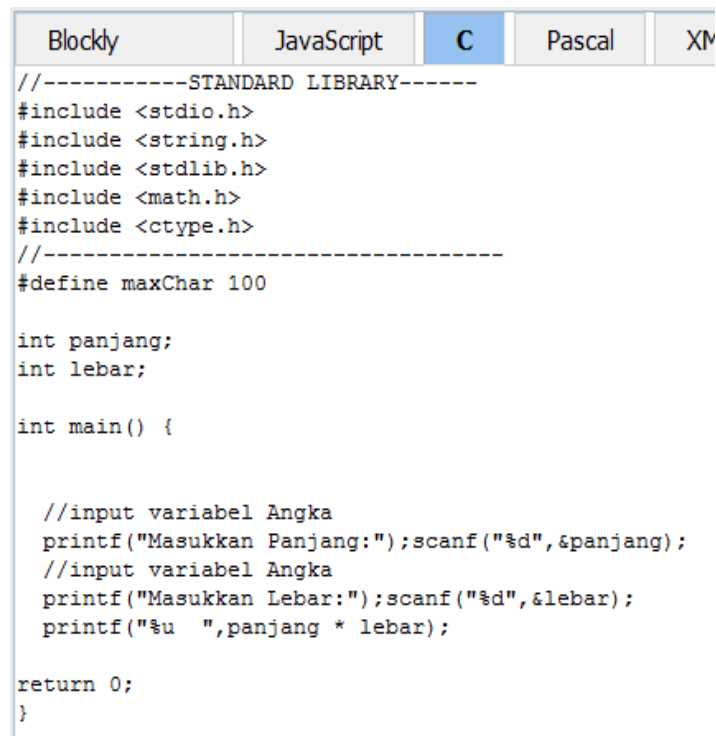
Gambar 2.9 Program sederhana untuk mencari nilai maksimum dari inputan user menggunakan IndoBlockly



Gambar 2.10 Aplikasi sederhana perhitungan luas persegi panjang menggunakan Indoblockly



Gambar 2.11 Hasil output running aplikasi sederhana perhitungan luas persegi panjang menggunakan Indoblockly



```

Blockly  JavaScript  C  Pascal  XML
//-----STANDARD LIBRARY-----
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
#include <ctype.h>
//-----
#define maxChar 100

int panjang;
int lebar;

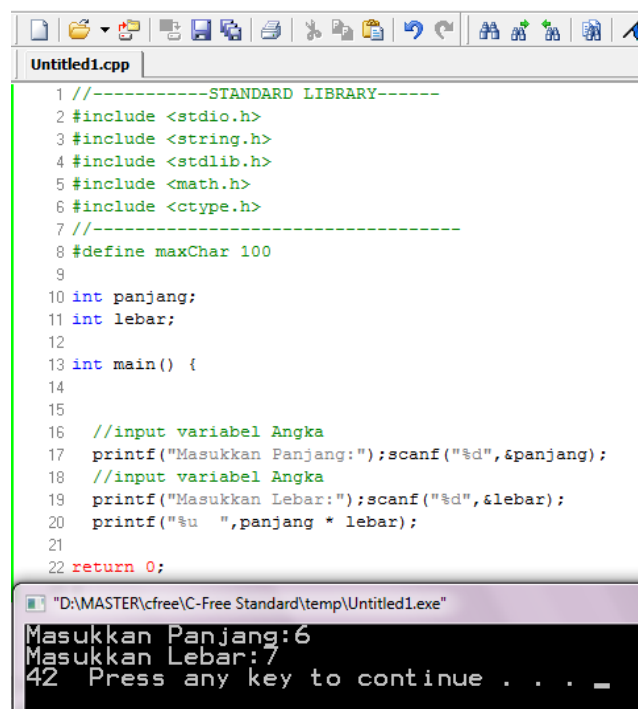
int main() {

    //input variabel Angka
    printf("Masukkan Panjang:");scanf("%d",&panjang);
    //input variabel Angka
    printf("Masukkan Lebar:");scanf("%d",&lebar);
    printf("%u ",panjang * lebar);

return 0;
}

```

Gambar 2.12. Hasil konvert code puzzle IndoBlockly ke *source code C*



```

1 //-----STANDARD LIBRARY-----
2 #include <stdio.h>
3 #include <string.h>
4 #include <stdlib.h>
5 #include <math.h>
6 #include <ctype.h>
7 //-----
8 #define maxChar 100
9
10 int panjang;
11 int lebar;
12
13 int main() {
14
15
16     //input variabel Angka
17     printf("Masukkan Panjang:");scanf("%d",&panjang);
18     //input variabel Angka
19     printf("Masukkan Lebar:");scanf("%d",&lebar);
20     printf("%u ",panjang * lebar);
21
22     return 0;

```

"D:\MASTER\cfree\C-Free Standard\temp\Untitled1.exe"

```

Masukkan Panjang:6
Masukkan Lebar:7
42 Press any key to continue . . . _

```

Gambar 2.13 Hasil output running program dengan C Free.

BAB III

METODOLOGI PENELITIAN

3.1 Metode Penelitian

Metode penelitian adalah cara yang digunakan oleh peneliti dalam mengumpulkan data penelitian (Arikunto, 2007). Metode penelitian yang digunakan dalam penelitian ini adalah metode eksperimen. Metode eksperimen yaitu metode penelitian yang dipergunakan untuk meneliti suatu objek dengan melakukan suatu percobaan secara nyata di lapangan (Arikunto, 2006). Penelitian ini bersifat eksperimen, yaitu sengaja mengusahakan tumbuhnya variabel-variabel dan selanjutnya dikontrol untuk dilihat pengaruhnya terhadap kemampuan pemahaman konsep mahasiswa melalui hasil belajar mahasiswa. karena disini penulis membandingkan antara dua kelas yang menggunakan pendekatan konstruktivisme dan yang menggunakan pendekatan konvensional serta apakah pengaruhnya terhadap kemampuan pemahaman konsep mahasiswa melalui hasil belajar.

3.2 Populasi dan Sampel

Penentuan populasi dan pengambilan sampel termasuk hal yang paling penting dalam penelitian kuantitatif seperti ini. Kesalahan dalam pengambilan sampel dapat menyebabkan sebuah penelitian menjadi penelitian yang tidak valid.

3.2.1 Populasi

Populasi adalah himpunan yang lengkap dari satuan-satuan atau individu yang karakteristiknya ingin diketahui (Anggoro, 2007). Populasi dalam penelitian ini adalah mahasiswa Teknik Informatika UIN Sunan Kalijaga Yogyakarta tahun 2012/2013 yang terdiri atas dua kelas yaitu kelas regular dan kelas mandiri.

3.2.2 Sampel

Sampel adalah sebagian anggota populasi yang memberikan keterangan atau data yang diperlukan dalam suatu penelitian (Anggoro, 2007). Gay dan Diehl menuliskan, untuk penelitian deskriptif, sampelnya 10% dari populasi, penelitian korelasional, paling sedikit 30 elemen populasi, penelitian perbandingan kausal, 30 elemen per kelompok, dan untuk penelitian eksperimen 15 elemen per kelompok. (Gay dan Diehl, 1992).

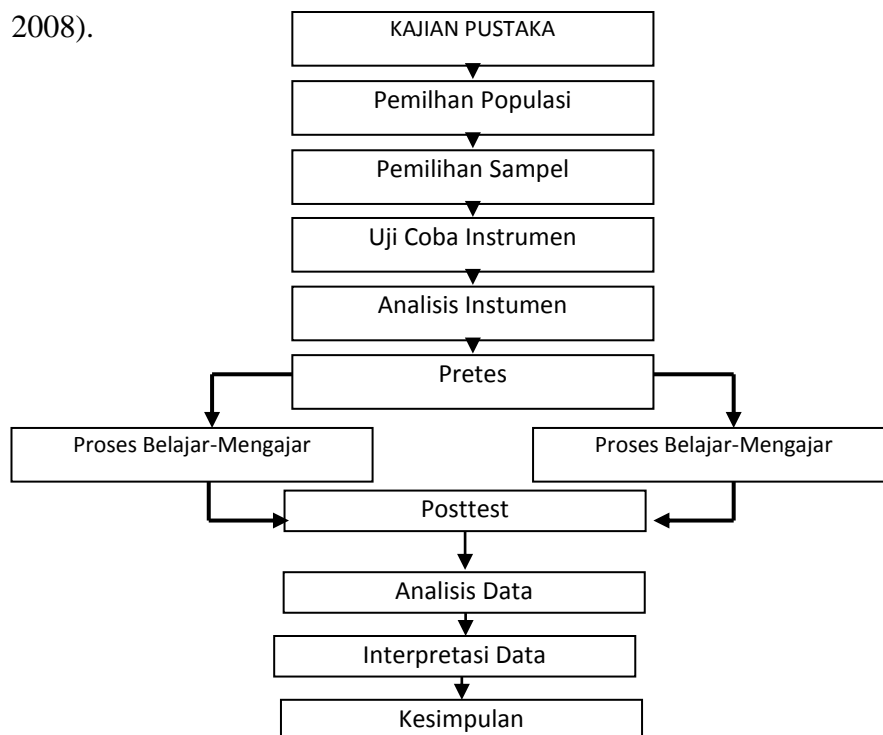
(Roscoe, 1975) dalam Uma Sekaran (1992: 252) memberikan pedoman penentuan jumlah sampel sebagai berikut:

- Sebaiknya ukuran sampel di antara 30 s/d 500 elemen.
- Jika sampel dipecah lagi ke dalam subsampel (laki/perempuan, SD/SLTP/SMU, dsb), jumlah minimum subsampel harus 30.
- Pada penelitian multivariate (termasuk analisis regresi multivariate) ukuran sampel harus beberapa kali lebih besar (10 kali) dari jumlah variable yang akan dianalisis.
- Untuk penelitian eksperimen yang sederhana, dengan pengendalian yang ketat, ukuran sampel bisa antara 10 s/d 20 elemen.

Sampel yang digunakan pada penelitian ini menggunakan teknik *sampling* yaitu *purposive sampling*, sampelnya diambil berdasarkan tujuan tertentu. Sampel diambil dari populasi terjangkau yang dibagi menjadi dua kelompok. Kelompok pertama sebagai kelas eksperimen dan kelompok kedua sebagai kelas kontrol. Kedua kelompok tersebut diambil sebagai sampel karena memiliki karakteristik yang hampir sama.

3.3 Desain Penelitian

Penelitian ini menggunakan desain *pretes postes control group design*, dalam desain ini digunakan dua kelompok subjek, satu diantaranya yang diberikan perlakuan. Desain penelitian ini dapat digambarkan sebagai berikut: (Emzir , 2008).



Gambar 3.1

Alur Penelitian

Tabel 3.1 Desain Kelompok Eksperimen dan Kontrol Pretes-Posttes

<i>Kelompok</i>	<i>Pretes</i>	<i>Treatment</i>	<i>Postes</i>
A	T ₁	X ₁	T ₂
B	T ₁	X ₂	T ₂

Keterangan:

A : Kelompok eksperimen (kelompok yang menggunakan *indoBlockly*)

B : Kelompok kontrol (kelompok yang menggunakan model konvensional)

T₁ : Tes awal yang sama pada kedua kelompok (*pretes*)

T₂ : Tes akhir yang sama pada kedua kelompok (*postes*)

X₁ : Perlakuan dengan menerapkan pembelajaran menggunakan *indoBlockly*

X₂ : Perlakuan dengan menerapkan model konvensional

3.4 Alur Penelitian

Alur penelitian dapat dilihat pada diagram alur Gambar 3.1.

3.5 Prosedur Penelitian

Dalam penelitian ini prosedur penelitian dibagi dalam tiga tahap, yaitu tahap persiapan, pelaksanaan, dan penyelesaian penelitian.

3.5.1 Persiapan Penelitian

Dalam melaksanakan penelitian ini ditempuh melalui langkah-langkah sebagai berikut:

- a. Mengajukan judul penelitian.

- b. Merancang instrumen penelitian.
- c. Bersamaan dengan penyusunan instrumen, penulis memohon izin untuk melakukan penelitian pada mata kuliah pemrograman terstruktur teknik informatika UIN Sunan Kalijaga.
- d. Menyusun jadwal kegiatan penelitian.
- e. Menguji reliabilitas, validitas, daya pembeda serta indeks kesukaran instrumen uji coba, kemudian melakukan revisi.
- f. Pelaksanaan penelitian.

3.5.2 Pelaksanaan Penelitian

Langkah-langkah yang dilakukan pada pelaksanaan penelitian adalah:

- a. Memilih sampel *purposive sampling* sebanyak dua kelas.
- b. Memberikan lembar soal pretes yang sama kepada kedua kelompok sampel kelas tersebut.
- c. Proses pembelajaran pada kelas eksperimen dengan menggunakan IndoBlockly dan kelas kontrol dengan menggunakan model praktikum konvensional.
- d. Setelah proses pembelajaran berakhir, dilakukan postes untuk kedua kelompok yang diteliti.
- e. Diadakan penilaian hasil pretes dan postes pada kedua kelompok kelas tersebut.

Langkah-langkah pelaksanaan penelitian dapat dilihat pada Lampiran A Pelaksanaan Penelitian halaman 83.

3.5.3 Penyelesaian Penelitian

Langkah-langkah penyelesaian penelitian adalah sebagai berikut:

- a. Menganalisis data hasil pretes dan postes dari kelompok eksperimen dan kelompok kontrol.
- b. Penyusunan skripsi.

3.6 Instrumen Penelitian

Instrumen penelitian merupakan alat bantu untuk pengumpulan dan pengolahan data tentang variabel-variabel yang diteliti. Oleh karena itu untuk memperoleh data dalam menjawab permasalahan untuk menguji hipotesis yang telah dirumuskan, maka diperlukan alat atau instrumen. Adapun instrumen dalam penelitian ini adalah:

3.6.1 Seperangkat Soal

Seperangkat soal dalam penelitian ini berupa tes, dimana untuk mengukur ada atau tidaknya serta besarnya pengaruh model pembelajaran menggunakan IndoBlockly terhadap pemahaman konsep algoritma pada mata kuliah pemrograman terstruktur.

Tes ini dikenakan kepada kedua kelompok subyek penelitian dengan kriteria tes yang sama, yaitu tes awal (pretes) dan tes akhir (postes). Tes awal dan tes akhir ini diadakan untuk mengetahui tingkat pemahaman algoritma sebelum dan sesudah pemberian perlakuan. Bentuk tes yang digunakan adalah tes uraian sebanyak sepuluh soal. Soal tes sebelumnya diujicobakan terlebih dahulu untuk

mengetahui kualitas instrumen atau alat pengumpul data yang digunakan. Instrumen disebut berkualitas dan dapat dipertanggungjawabkan pemakaiannya apabila sudah terbukti validitas, reliabilitas, indeks kesukaran, dan daya pembedanya.

3.6.1.1 Menentukan Validitas Soal

Suatu alat evaluasi disebut valid (absah atau sah) apabila alat tersebut mampu mengevaluasi apa yang seharusnya di evaluasi. Validitas butir soal essay (uraian) dihitung dengan menggunakan rumus koefisien korelasi *Product Moment* memakai angka kasar, yaitu (Suherman dan Sukjaya, 1990):

$$r_{xy} = \frac{N \cdot \sum XY - \sum X \sum Y}{\sqrt{(N \cdot \sum X^2 - (\sum X)^2)(N \cdot \sum Y^2 - (\sum Y)^2)}}$$

Keterangan:

- r_{xy} = Koefisien korelasi antara variebel X dan Y
- N = banyaknya mahasiswa yang mengikuti tes
- X = nilai hasil uji coba
- Y = skor total

Tabel 3.2 Klasifikasi Koefisien Validitas

No	Nilai r_{xy}	Interpretasi
1	$0,80 < r_{xy} \leq 1,00$	Validitas Sangat Tinggi
2	$0,60 < r_{xy} \leq 0,80$	Validitas Tinggi
3	$0,40 < r_{xy} \leq 0,60$	Validitas Sedang
4	$0,20 < r_{xy} \leq 0,40$	Validitas Rendah
5	$0,00 < r_{xy} \leq 0,20$	Validitas Sangat Rendah
6	$r_{xy} \leq 0,00$	Tidak Valid

Sumber: (Suherman dan Sukjaya, 1990).

Kemudian untuk menguji keberartian validitas (koefisien korelasi) soal essay digunakan statistik uji t yang dikemukakan oleh (Sugiyono, 2007) yaitu:

$$t = r_{xy} \sqrt{\frac{n-2}{1-r_{xy}}}$$

Keterangan: t = daya beda

Bila $t_{hitung} > t_{tabel}$ maka soal sah tetapi jika $t_{hitung} \leq t_{tabel}$, maka soal tersebut tidak sah dan tidak digunakan untuk instrumen penelitian.

3.6.1.2 Menentukan Reliabilitas Soal

Reliabilitas suatu alat ukur sebagai suatu alat yang memberikan hasil yang tetap sama (konsisten, ajeg). Hasil pengukuran itu harus tetap sama (relative sama) jika pengukurannya diberikan kepada subjek yang sama meskipun dilakukan oleh orang yang berbeda, waktu berbeda, dan tempat yang berbeda. Menurut Suherman dan Sukjaya untuk menentukan reliabilitas soal berbentuk essay (uraian) digunakan rumus *Alpha*, yaitu:

$$r_{11} = \left(\frac{n}{n-1} \right) \left(1 - \frac{\sum S_i^2}{S_i^2} \right)$$

Keterangan:

r_{11} = koefisien reliabilitas instrumen

n = banyaknya butir soal

$\sum S_i^2$ = jumlah skor tiap butir soal

S_i^2 = varians skor total

Sedangkan untuk menghitung varians skor digunakan rumus:

$$S_i^2 = \frac{\sum x_i^2 - \frac{(\sum x_i)^2}{N}}{N}$$

Keterangan:

N = banyaknya sampel / peserta tes

x_i = skor butir soal ke- i

i = nomor soal

Tabel 3.3 Klasifikasi Koefisien Reliabilitas

No	Nilai r_{11}	Interpretasi
1	$0,90 < r_{11} \leq 1,00$	Sangat tinggi
2	$0,70 < r_{11} \leq 0,90$	Tinggi
3	$0,40 < r_{11} \leq 0,70$	Sedang
4	$0,20 < r_{11} \leq 0,40$	Rendah
5	$r_{11} \leq 0,20$	Sangat rendah

Sumber: (Suherman dan Sukjaya, 1990)

3.6.1.3 Menentukan Daya Pembeda Soal

Seberapa jauh kemampuan butir soal tersebut mampu membedakan antara testi yang mengetahui jawabannya dengan benar dengan testi yang tidak dapat menjawab soal tersebut (Suherman dan Sukjaya, 1990). Daya pembeda dihitung dengan menggunakan rumus DP untuk tes sebagai berikut:

$$DP = \frac{SA - SB}{\frac{1}{2} \times N \times Maks}$$

Keterangan:

DP = daya pembeda

SA = jumlah skor yang dicapai mahasiswa kelompok atas

SB = jumlah skor yang dicapai mahasiswa kelompok bawah

N = jumlah mahasiswa dari kelompok atas dan kelompok bawah

$Maks$ = Skor maksimal

Tabel 3.4 Klasifikasi Koefisien Daya Pembeda

No	Nilai Daya Pembeda (<i>DP</i>)	Interpretasi
1	$DP \leq 0,00$	Sangat jelek
2	$0,00 < DP \leq 0,20$	Jelek
3	$0,20 < DP \leq 0,40$	Cukup
4	$0,40 < DP \leq 0,70$	Baik
5	$0,70 < DP \leq 1,00$	Sangat Baik

Sumber: Suherman dan Sukjaya (1990)

3.6.1.4 Menentukan Tingkat Kesukaran Soal

Untuk menghitung tingkat kesukaran soal yang berbentuk uraian digunakan rumus:

$$IK = \frac{SA + SB}{N \times Maks}$$

Keterangan:

IK = Indeks kesukaran tiap butir soal

SA = Jumlah skor yang dicapai mahasiswa kelompok atas

SB = jumlah skor yang dicapai mahasiswa kelompok bawah

N = Jumlah mahasiswa dari kelompok atas dan kelompok bawah

Maks = skor maksimal

Tabel 3.5 Klasifikasi Koefisien Indeks Kesukaran

No	Nilai Daya Pembeda (<i>DP</i>)	Interpretasi
1	$IK \leq 0,00$	Terlalu Sukar
2	$0,00 < IK \leq 0,30$	Sukar
3	$0,30 < IK \leq 0,70$	Sedang
4	$0,70 < IK \leq 1,00$	Mudah
5	$IK = 1,00$	Sangat Mudah

Sumber: (Suherman dan Sukjaya, 1990)

3.6.2 Hasil Uji Instrumen

Hasil uji instrumen dibahas pada bab III ini dikarenakan uji instrumen tidak termasuk dalam pembahasan. Instrumen penelitian merupakan alat bantu untuk pengumpulan dan pengolahan data tentang variabel-variabel yang diteliti. Dalam kasus ini instrumen penelitian adalah soal. Uji instrumen perlu dilakukan karena untuk menguji apakah instrumen itu layak atau tidak digunakan dalam penelitian. Jika hasil uji instrumen mengatakan sebuah instrumen itu tidak layak digunakan maka jelas instrumen tersebut tidak boleh digunakan dalam penelitian. Ada dua cara uji instrumen dan dalam penelitian ini kedua-duanya sudah dilakukan yaitu uji instrumen expert yaitu dengan bertanya dengan ahlinya dalam kasus ini yaitu dosen pemrograman terstruktur. Kedua yaitu uji instrumen uji coba dengan cara mengujikan soal tersebut kepada sampel kemudian hasil skornya dianalisis menggunakan analisis uji statistik uji instrumen.

3.6.2.1 Hasil Uji Instrumen Expert

Sesuai dengan batasan masalah bahwa penelitian ini hanya menguji pada pemahaman mengenai input output, variabel, array, kondisi, dan perulangan. Penulis sudah membuat soal yang terdiri dari 10 soal uraian yang masing-masing soal mempunyai skor 10 sehingga total skor adalah 100. Dari 10 soal yang penulis ajukan pada dosen pemrograman terstruktur semuanya diterima dan dinyatakan telah sesuai dengan tiga soal yang perlu di revisi. Revisi tiga soal tersebut sebagai berikut :

1. Soal nomor satu mengenai flowchart ditambah soal untuk menyebutkan jenis-jenis flowchart yang mahasiswa kenal.
2. Soal nomor tiga mengenai array ditambah array dua dimensi jadi tidak hanya array satu dimensi.
3. Soal nomor delapan mengenai mencari bilangan terbesar direvisi menjadi soal dengan variabel angka statis tidak menggunakan input angka dinamis.

3.6.2.2 Hasil Uji Instrumen Uji Coba

Teknik pengolahan data hasil dari uji instrumen menggunakan *MS Excel* dengan langsung mengimplementasikan rumus uji validitas, reliabilitas, daya beda dan tingkat kesukaran. Hasil perhitungan uji coba instrumen dapat dilihat pada Tabel 3.6. Dari hasil analisis uji coba instrumen seluruh soal memenuhi kriteria untuk dipakai dalam penelitian. Akan tetapi ada tiga soal yaitu soal nomor 8,9, dan 10 yang mempunyai daya beda jelek

dan tergolong sulit, itu terbukti dari hasil test uji coba instrumen hanya ada satu mahasiswa yang mendapatkan skor cukup baik untuk soal nomor 8,9, dan 10. Sehingga berdasarkan hasil penelitian uji instrumen penulis memutuskan untuk menggunakan tujuh soal yaitu soal nomor 1,2,3,4,5,6, dan 7 dalam pretes dan postes. Perhitungan selengkapnya mengenai uji coba instrumen ini dapat dilihat pada Lampiran B Analisis Uji Instrumen halaman 84. Soal pretes dan postes yang telah direvisi berdasarkan hasil uji instrumen baik uji instrumen expert maupun uji instrumen uji coba soal dapat dilihat pada Lampiran C Soal Pretes dan Postes halaman 87.

Adapun hasil perhitungan instrumen disajikan pada Tabel 3.6 sebagai berikut:

Tabel 3.6 Hasil Analisis Instrumen Tes Uji Coba

No Soal	Validitas		Reliabilitas		Daya Pembeda		Indeks Kesukaran	
	Indeks	Interpretasi	Indeks	Interpretasi	Indeks	Interpretasi	Indeks	Interpretasi
1	0,7126652	Tinggi	0,597349	Sedang	0,43	Baik	0,44	Sedang
2	0,6850126	Tinggi			0,24	Sedang	0,27	Sedang
3	0,9131264	Sangat Tinggi			0,33	Sedang	0,46	Sedang
4	0,6340893	Tinggi			0,35	Sedang	0,49	Sedang
5	0,9099195	Sangat Tinggi			0,47	Baik	0,49	Sedang
6	0,9387633	Sangat Tinggi			0,39	Sedang	0,51	Sedang
7	0,6942779	Tinggi			0,28	Sedang	0,35	Sedang
8	0,5049758	Sedang			0,09	Jelek	0,15	Sulit
9	0,5476324	Sedang			0,16	Jelek	0,23	Sulit
10	0,4252906	Sedang			0,19	Jelek	0,19	Sulit

3.7 Teknik Pengumpulan Data

Pengumpulan data dilakukan pada awal penelitian yaitu menggunakan pretes dan akhir penelitian yaitu menggunakan postes. Secara garis besar teknik pengumpulan data dalam penelitian ini dapat disajikan dalam Tabel 3.7 sebagai berikut:

Tabel 3.7 Teknik Pengumpulan Data

No	Jenis data	Teknik Pengumpulan
1	Data kuantitatif dapat dilihat dari hasil belajar mahasiswa.	Tes (pretes dan postes)

3.8 Teknik Pengolahan Data

Data yang diperoleh dalam penelitian ini berasal dari tes (pretes dan postes) yang berupa soal uraian. Data-data yang diperoleh adalah sebagai berikut:

3.8.1 Pengolahan Data Kuantitatif

Ada dua macam data yang biasanya digunakan dalam penelitian yaitu data kuantitatif dan kualitatif. Penelitian ini hanya menggunakan dan menganalisis hasil perolehan skor pretes dan postes yang berupa data kuantitatif. Hasil dari analisis data penelitian ini adalah sebagai berikut:

3.8.1.1 Analisis Data Tes

Analisis data tes dilakukan untuk menjawab rumusan masalah apakah ada pengaruh pembelajaran menggunakan IndoBlockly dibandingkan dengan

pembelajaran konvensional pada praktikum pemrograman terstruktur. Selain dari hasil pretes dan postes, data kuantitatif juga diperoleh dari gain kedua kelas. Gain yang dimaksud dalam penelitian ini adalah gain yang ternormalisasi atau *Normalized Gain (NG)*. *Normalized Gain* adalah proporsi gain actual (postes-pretes) dengan gain maksimal yang dicapai. Rumus yang digunakan untuk menjelaskan gain dibuat oleh Hake (Mustika, 2009), yaitu:

$$NG = \frac{POS - PRE}{IDEAL - PRE}$$

Keterangan:

GN = Gain yang ternormalisasi

POS = Skor postes mahasiswa

PRE = Skor pretes mahasiswa

IDEAL = Skor ideal / Nilai maksimum

Tabel 3.8 Kategori Gain yang dinormalisasi

No	Normalized Gain	Kriteria
1	$NG \geq 0,70$	Tinggi
2	$0,30 < NG \leq 0,70$	Sedang
3	$NG < 0,30$	Rendah

Langkah-langkah yang dilakukan dalam teknik analisis data tes, baik pretes, postes, maupun indeks gain adalah sebagai berikut:

3.8.1.1.1 Analisis Deskriptif

Analisis deskriptif dilakukan untuk memperoleh gambaran umum mengenai data pretes dan data postes dari kelas eksperimen dan kelas kontrol yang diperoleh berupa skor rata-rata (mean) dan standar deviasi.

3.8.1.1.2 Analisis Inferensi

Analisis inferensi dilakukan untuk memperoleh kesimpulan ada atau tidaknya pengaruh pembelajaran menggunakan IndoBlockly pada kelas eksperimen dan dibandingkan dengan kelas kontrol yang menggunakan model pembelajaran konvensional. Analisis ini pada intinya merupakan uji perbedaan dua rata-rata, baik uji dua pihak maupun satu pihak. Sebelum melakukan uji perbedaan dua rata-rata, terlebih dahulu dilakukan uji normalitas dan homogenitas. Uji normalitas dan uji homogenitas dipandang perlu dilakukan karena dengan dilakukannya uji normalitas dan homogenitas, langkah-langkah penelitian dapat dipertanggungjawabkan dan kesimpulan yang dibuat berdasarkan teori dapat berlaku (Yusniati, 2009). Pengolahan data penganalisan data hasil penelitian dilakukan dengan bantuan *software R tool for OSX* dan *Microsoft Excel 2011*. Adapun langkah-langkah analisis inferensi adalah sebagai berikut:

3.8.1.1.2.1 Uji Normalitas

Uji ini dilakukan untuk mengetahui data dari masing-masing kelompok sampel berdistribusi normal atau tidak. Data-data yang diuji adalah data pretes kelas kontrol, pretes kelas eksperimen, postes kelas

kontrol, postes kelas eksperimen, gain kelas kontrol dan gain kelas eksperimen. Dalam uji normalitas ini digunakan uji *Shapiro –Wilk*. Jika data berasal dari distribusi yang normal, maka analisa data dilanjutkan dengan uji homogenitas varians untuk menentukan uji parametrik yang sesuai. Namun, jika data berasal dari populasi yang tidak berdistribusi normal, maka tidak dilakukan uji homogenitas varians tetapi langsung dilakukan uji kesamaan dua rata-rata (uji non-parametrik) yaitu dengan menggunakan *Mann Whitney U*.

3.8.1.1.2.2 Uji Homogenitas Varians

Uji homogenitas varians dilakukan jika data yang diolah berdistribusi normal. Uji homogenitas ini dilakukan untuk mengetahui apakah variansi populasi data yang diuji memiliki variansi yang homogen atau tidak. Untuk menguji homogenitas varians digunakan uji *Lavene's Test* dengan mengambil taraf kepercayaan 95% (taraf signifikansi 5%). Jika data yang telah dianalisis bersifat normal dan homogen, maka data tersebut dilakukan uji perbedaan dua rata-rata.

3.8.1.1.2.3 Uji Perbedaan Dua Rata-rata

Uji perbedaan dua rata-rata yang dilakukan yaitu untuk menguji apakah terdapat perbedaan rata-rata (*mean*) pretes dan postes antara kelas eksperimen dan kelas kontrol. Uji perbedaan dua rata-rata terhadap skor pretes dilakukan dengan menggunakan uji dua pihak dan uji perbedaan dua rata-rata terhadap skor postes dilakukan dengan menggunakan uji satu pihak. Jika data telah berdistribusi normal dan homogen, maka dilakukan pengujian

perbedaan dua rata-rata dengan menggunakan uji-t. Sedangkan untuk data yang berdistribusi normal tetapi tidak homogen, maka dilakukan pengujian perbedaan dua rata-rata dengan menggunakan uji-t'. Adapun data yang tidak berdistribusi normal maka pengujiannya menggunakan uji non-parametrik yaitu uji *Mann Whitney U*. Jika rata-rata pretes kelas eksperimen dan kelas kontrol secara statistik tidak sama, maka untuk menguji peningkatan kemampuan mahasiswa diambil dari *indeks gain*.

3.8.1.1.2.4 Analisis Korelasi Pretes Postes Kelas Ekperimen

Analisis korelasi bertujuan untuk mengetahui korelasi antara nilai pretes dan postes pada kelas eksperimen. Besarnya korelasi antara nilai pretes dan postes menunjukkan besarnya pengaruh penerapan model pembelajaran IndoBlockly. Hasil dari analisis korelasi adalah apakah penerapan model pembelajaran IndoBlockly itu mempunyai korelasi yang positif, negatif, atau tidak berpengaruh. Dalam penelitian ini analisis korelasi digunakan untuk menjelaskan derajat hubungan antara variabel bebas (independent) dengan variabel terikat (dependent) dengan nilai : $-1 \leq rs \leq 1$, dimana :

- a. Bilai nilai $rs = -1$ atau mendekati -1 , maka korelasi kedua variabel dikatakan sangat kuat dan negatif artinya sifat hubungan dari kedua variabel berlawanan arah, maksudnya jika nilai **X** naik maka nilai **Y** akan turun atau sebaliknya.
- b. Bila nilai $rs = 0$ atau mendekati 0 , maka korelasi dari kedua variabel sangat lemah atau tidak terdapat korelasi sama sekali.

- c. Bila nilai $r_s = 1$ atau mendekati 1 , maka korelasi dari kedua variabel sangat kuat dan positif, artinya hubungan dari kedua variabel yang diteliti bersifat searah, maksudnya jika nilai X naik maka nilai Y juga naik atau sebaliknya.

Adapun kriteria penilaian korelasi adalah sebagai berikut (Sugiyono, 2003) :

Tabel 3.9 Kriteria Penilaian Korelasi

Interval Koefisien	Tingkat Hubungan
0.00 – 0.199	Sangat Rendah
0.20 – 0.399	Rendah
0.40 – 0.599	Sedang
0.60 – 0.799	Kuat
0.80 – 1.000	Sangat Kuat

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

Bab ini akan membahas mengenai analisis data dari hasil pengolahan data yang diperoleh dari hasil penelitian. Hasil analisis data yang diperoleh merupakan gambaran keseluruhan hasil dari penelitian yang telah dilaksanakan. Data yang diperoleh dari penelitian ini yaitu data kuantitatif. Data kuantitatif merupakan data yang diperoleh dari hasil tes yang berupa pretes dan postes. Dari hasil pretes dan postes diperoleh data kuantitatif lainnya yaitu data gain. Pengolahan data dilakukan menggunakan software *Statistic R Tool For OSX* dan *Microsoft Office Excel 2011*.

4.1 Hasil Penelitian dan Pembahasan

Soal pretes dan postes yang tadinya berjumlah sepuluh butir soal kemudian dilakukan uji instrumen expert dan analisis uji coba soal menghasilkan tujuh soal yang valid sehingga hanya tujuh soal yang digunakan untuk pretes dan postes. Populasi dari penelitian ini adalah mahasiswa baru Teknik Informatika UIN Sunan Kalijaga 2012/2013 yang berjumlah kurang lebih seratus mahasiswa yang terdiri dari dua kelas yaitu kelas regular dan kelas mandiri. Sampel pada penelitian ini adalah kelas regular yang berjumlah 56 mahasiswa kemudian kami bagi menjadi dua kelas yaitu kelas kontrol dan kelas eksperimen. Pengambilan sampel berdasarkan teori *purposive sampling* dengan jumlah sampel 28 kelas kontrol dan 28 kelas eksperimen. Jumlah sampel tersebut sudah memenuhi kaidah

pengambilan sampel sesuai dengan teori Gay dan Diehl yang sudah kami bahas pada BAB III Metodologi Penelitian.

4.1.1 Analisis Deskriptif Data Hasil Pretes dan Postes

Pada bab sebelumnya telah dijelaskan bahwa analisis deskriptif dilakukan untuk memperoleh gambaran umum mengenai data pretes dan data postes dari kelas eksperimen dan kelas kontrol yang diperoleh berupa skor rata-rata (mean) dan standar deviasi. Untuk mengetahui kemampuan awal mahasiswa sebelum diberi perlakuan pembelajaran dengan menggunakan IndoBlockly, maka pada kelas eksperimen dan kelas kontrol diberikan soal pretes untuk mengetahui kemampuan mahasiswa baik dari kelas eksperimen maupun kelas kontrol. Sedangkan untuk mengetahui kemampuan akhir mahasiswa serta untuk melihat peningkatan kemampuan mahasiswa terhadap pemahaman algoritma, maka pada kelas eksperimen dan kelas kontrol diberikan soal postes. Selanjutnya, untuk mengetahui kualitas peningkatan kemampuan mahasiswa dilakukan analisis data gain kelas eksperimen dan juga kelas kontrol. Berdasarkan data yang diperoleh, gambaran umum tersebut dapat dilihat pada Tabel 4.1 di bawah ini.

Tabel 4.1 Statistik Deskriptif Data Hasil Pretes dan Postes

Kelas	Banyak Mhs	Pretes				Postes			
		Min	Max	Mean	Sd	Min	Max	Mean	Sd
Eksperimen	28	5,00	65,00	19,32	12,46	37,00	70,00	50,61	10,54
Kontrol	28	7,00	50,00	19,25	9,78	5,00	67,00	26,39	15,09

Berdasarkan Tabel 4.1 dapat dilihat bahwa mahasiswa pada kelas eksperimen dan kelas kontrol berjumlah sama, yaitu masing-masing kelas terdiri dari 28 mahasiswa. Rata-rata skor pretes kelas eksperimen dan kelas kontrol masing-masing adalah 19,32 dan 19,25 dimana rata-rata skor pretes kelas eksperimen lebih tinggi sedikit (hanya terpaut koma) daripada rata-rata skor pretes kelas kontrol. Sedangkan rata-rata skor postes kelas eksperimen dan kelas kontrol masing-masing adalah 50,61 dan 26,39 dari rata-rata ini juga sudah terlihat bahwa rata-rata skor postes kelas eksperimen yang menggunakan IndoBlockly lebih tinggi daripada skor kelas kontrol yang menggunakan pembelajaran model konvensional. Sementara itu, standar deviasi pretes kelas eksperimen dan kelas kontrol masing-masing adalah 12,46 dan 9,78 yaitu kelas eksperimen lebih besar standar deviasinya dibandingkan kelas kontrol. Akan tetapi, untuk standar deviasi postes dimana kelas kontrol lebih besar daripada kelas eksperimen masing-masing adalah 15,09 dan 10,54. Perhitungan lengkap analisis deskriptif dapat dilihat pada Lampiran D Perolehan Data dan Analisis Deskriptif halaman 90.

Hasil analisis deskriptif memberikan gambaran bahwa terdapat selisih yang cukup besar antara rata-rata kemampuan akhir mahasiswa antara kelas eksperimen dan kelas kontrol. Selain perbedaan rata-rata kemampuan awal dan perbedaan rata-rata kemampuan akhir pada kedua kelas, hasil tersebut juga memberikan gambaran bahwa penyebaran data pada kedua kelas berbeda, di mana

skor pretes kelas eksperimen lebih menyebar (bervariasi) dari pada skor pretes kelas kontrol. Akan tetapi, skor postes kelas kontrol lebih menyebar dari pada skor postes kelas eksperimen. Perbedaan penyebaran data tersebut dapat dilihat dengan adanya perbedaan standar deviasi pada kedua kelas dan batas nilai maksimum dan minimum kedua kelas, baik pada skor pretes maupun skor postes. Namun demikian, untuk mengetahui apakah rata-rata kemampuan awal dan akhir mahasiswa kedua kelas berbeda secara signifikan atau tidak, akan dilakukan uji statistik. Untuk menguji hal tersebut, berikut ini dilakukan analisis inferensi terhadap data hasil dari pretes dan postes mahasiswa.

4.1.2 Analisis Inferensi Data Hasil Pretes dan Postes

Seperti yang telah diuraikan pada Bab III, analisis inferensi dilakukan untuk memperoleh kesimpulan apakah model pembelajaran menggunakan IndoBlockly yang diterapkan pada mahasiswa kelas eksperimen lebih baik daripada kelas kontrol yang menggunakan model pembelajaran biasa/konvensional. Analisis ini pada intinya merupakan uji perbedaan dua rata-rata, baik uji satu pihak maupun dua pihak.

4.1.2.1 Analisis Data Kemampuan Awal Mahasiswa

Analisis data kemampuan awal mahasiswa ini diperoleh dari data pretes. Sesuai dengan tahap-tahap pelaksanaan penelitian, sebelum pembelajaran pada kelas eksperimen dan kelas kontrol berlangsung dilaksanakan tes awal (pretes). Tujuan diberikannya pretes pada masing-masing kelas

adalah untuk mengetahui kemampuan awal mahasiswa pada kelas eksperimen dan kelas kontrol sama atau berbeda. Berikut ini disajikan analisis deskriptif data pretes kelas eksperimen dan kelas kontrol.

Tabel 4.2 Data Statistik Skor Pretes Kelas Eksperimen dan Kelas Kontrol

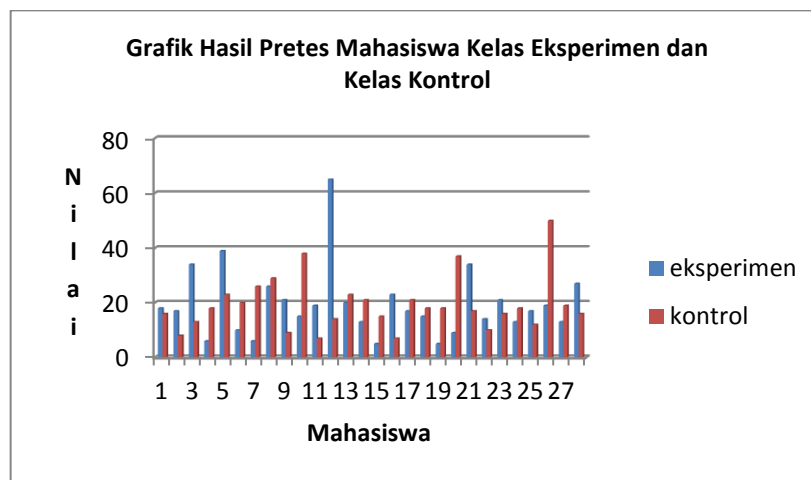
Kelas	Banyak Mhs	Pretes				
		Min	Max	Mean	Sd	variance
Eksperimen	28	5,00	65,00	19,32	12,46	155,25
Kontrol	28	7,00	50,00	19,25	9,78	95,65

Berdasarkan Tabel 4.2 terlihat bahwa rata-rata skor pretes yang diperoleh mahasiswa kelas eksperimen dan mahasiswa kelas kontrol hampir sama. Hal ini menunjukkan bahwa kemampuan awal mahasiswa baik kelas eksperimen maupun kelas kontrol hampir sama secara eksak. Secara grafik hasil pretes kemampuan mahasiswa antara kelas eksperimen dan kelas kontrol dapat dilihat pada Gambar 4.1.

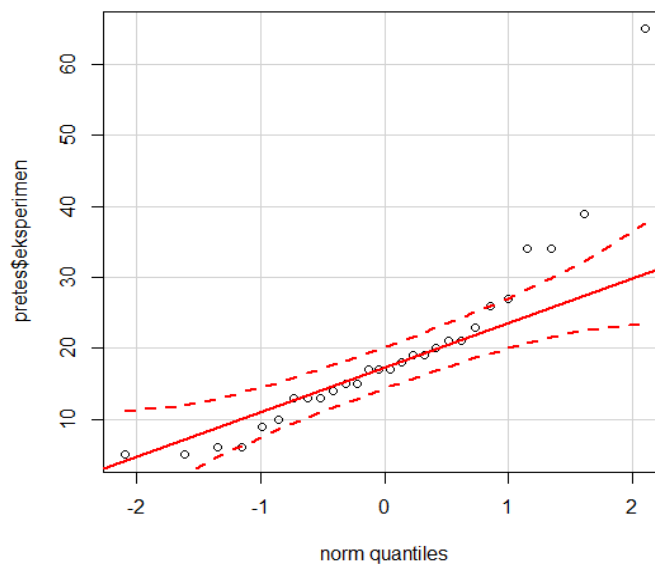
Sekilas melihat dari hasil rata-rata tersebut dapat dikatakan bahwa kelas eksperimen dan kelas kontrol homogen (berasal dari populasi yang sama), tapi jelas ini harus dibuktikan dengan uji statistik terlebih dahulu. Uji statistik yang pertama yaitu uji normalitas, dari hasil dari uji normalitas kita dapat menentukan langkah-langkah analisis data berikutnya. Jika data terbukti normal jelas kita akan menggunakan analisis data parametrik, tapi jika data ternyata tidak normal maka akan digunakan metode analisis data non parametrik. Pengujian konormalan dapat disajikan menggunakan Q-Q Plot seperti pada Gambar 4.2 dan Gambar 4.3.

Gambar 4.1 dan Gambar 4.2 terlihat bahwa sebaran data pretes tidak tersebar pada garis lurus, sehingga dapat diduga bahwa data sampel kelas

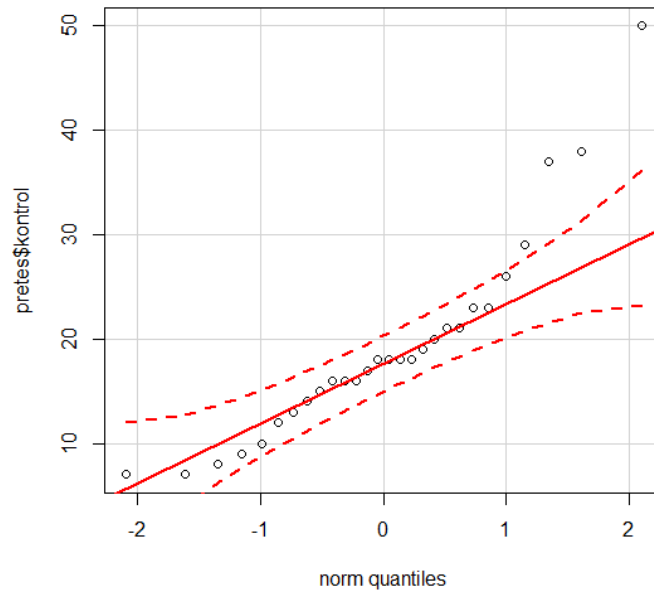
eksperimen dan kelas kontrol berasal dari populasi yang berdistribusi tidak normal. Namun untuk mengetahui apakah prediksi tersebut itu benar, maka selanjutnya dilakukan uji normalitas.



Gambar 4.1 Hasil Pretes Kelas Eksperimen dan Kelas Kontrol



Gambar 4.2 .Q-Q Plot pretes kelas eksperimen



Gambar 4.3. Q-Q Plot pretes kelas kontrol

4.1.2.1.1 Uji Normalitas Data Pretes

Langkah pertama yang dilakukan adalah dengan menguji normalitas data pretes kedua kelas, untuk mengetahui apakah kedua kelas tersebut berdistribusi normal atau tidak. Untuk menguji normalitas data pretes, digunakan uji statistik *Shapiro-Wilk*. Perumusan hipotesis pengujian normalitas data pretes sebagai berikut:

H_0 : Skor pretes sampel berasal dari populasi yang berdistribusi normal.

H_1 : Skor pretes sampel berasal dari populasi yang tidak berdistribusi normal.

Dengan menggunakan taraf signifikansi (α) sebesar 0,05 kriteria pengujiannya adalah sebagai berikut:

1. Jika nilai p -value lebih dari atau sama dengan 0,05 maka H_0 diterima.
2. Jika nilai p -value kurang dari 0,05 maka H_0 ditolak.

Adapun hasil dari analisis uji normalitas skor pretes kelas eksperimen dan kelas kontrol dengan uji *Shapiro-Wilk* menggunakan *R statistic tool* disajikan dalam Tabel 4.3. Analisis secara lengkap dapat dilihat pada Lampiran E Uji Normalitas Data Pretes halaman 93.

Tabel 4.3 Hasil Uji Normalitas Data Pretes

Kelas	Shapiro-Wilk	
	W	p -value
Eksperimen	0,83	0,0003
Kontrol	0,87	0,003

Dari Tabel 4.3 terlihat bahwa (p -value) uji *Shapiro-Wilk* untuk kelas eksperimen adalah 0,0003 dan kelas kontrol adalah 0,003. Nilai p -value baik kelas eksperimen maupun kelas kontrol kurang dari 0,05 maka H_0 ditolak, artinya data pretes kelas eksperimen dan kelas kontrol tidak berdistribusi normal. Hasil pengujian menggunakan uji *Shapiro-Wilk* menunjukkan hasil bahwa data hasil dari pretes kedua kelas terbukti tidak normal sehingga pengujian yang dilakukan selanjutnya adalah uji homogenitas. Dikarenakan hasil dari uji normalitas menunjukkan bahwa data berdistribusi tidak normal maka digunakan uji kesamaan dua rata-rata non-parametrik yaitu uji *Mann Whitney*.

4.1.2.1.2 Uji Perbedaan Dua Rata-rata Data Pretes

Uji kesamaan dua rata-rata dalam penelitian ini menggunakan uji *Mann-Whitney*. Hipotesis dalam pengujian kesamaan dua rata-rata dirumuskan sebagai berikut:

H0 : Tidak terdapat perbedaan kemampuan awal mahasiswa antara kelas eksperimen dengan kelas kontrol.

H1 : Terdapat perbedaan kemampuan awal mahasiswa antara kelas eksperimen dengan kelas kontrol.

Pasangan hipotesis tersebut jika dirumuskan ke dalam bentuk hipotesis statistik adalah sebagai berikut:

$$H_0: \mu_e = \mu_k$$

$$H_1: \mu_e \neq \mu_k$$

Keterangan:

μ_e = rata-rata nilai pretes kelas eksperimen.

μ_k = rata-rata nilai pretes kelas kontrol.

Dengan menggunakan taraf signifikansi (α) adalah 0,05 maka kriteria pengujiannya sebagai berikut:

1. Jika nilai *p-value* lebih dari atau sama dengan 0,05 maka H0 diterima.
2. Jika nilai *p-value* kurang dari 0,05 maka H0 ditolak.

Hasil analisis uji *Mann-Whitney* skor pretes disajikan dalam Tabel 4.4. Analisis secara lengkap dapat dilihat pada Lampiran F Uji Beda Rata-rata Pretes halaman 94.

Tabel 4.4 Hasil Uji Mann-Whitney Data Pretes

Mann-Whitney	Nilai
W	411,5
<i>p-value</i>	0,749

Pada Tabel 4.4 diperoleh bahwa nilai *p-value Mann-Whitney* adalah 0,749. Karena 0,749 lebih dari 0,05 maka berdasarkan kriteria pengujian di

atas H_0 diterima. Hal ini menunjukkan bahwa tidak terdapat perbedaan rata-rata skor pretes antara kelas eksperimen dan kelas kontrol. Dengan kata lain, kemampuan awal mahasiswa kelas eksperimen dengan kelas kontrol adalah sama.

4.1.2.2 Analisis Data Kemampuan Akhir Mahasiswa

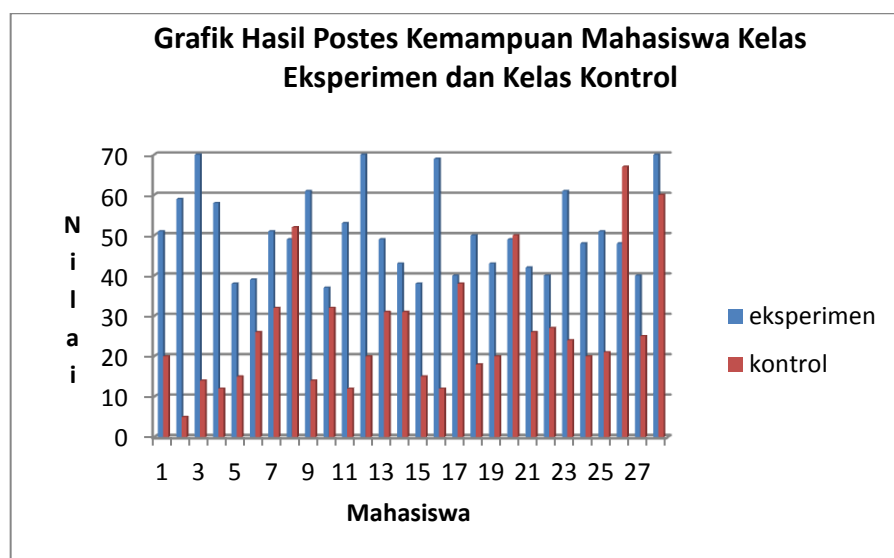
Berdasarkan analisis data pretes diperoleh kesimpulan tidak terdapat perbedaan rata-rata yang signifikan kemampuan awal mahasiswa antara kelas eksperimen dan kelas kontrol, atau dengan kata lain kemampuan awal mahasiswa pada kedua kelas tersebut adalah sama. Maka penelitian ini layak untuk dilanjutkan dikarenakan memang terbutki bahwa kedua kelas yaitu kelas eksperimen dan kelas kontrol tidak terdapat perbedaan rata-rata yang signifikan. Selanjutnya adalah analisis data kemampuan akhir mahasiswa atau analisis data postes. Hasil uji kemampuan mahasiswa dari kelas eksperimen dan kelas kontrol yaitu analisis data hasil postes dapat dilihat pada Lampiran D Perolehan Data dan Analisis Deskriptif halaman 90, berikut ini disajikan analisis deskriptif data postes kelas eksperimen dan kelas kontrol.

Tabel 4.5 Statistika Deskriptif Skor Postes Kelas Eksperimen dan Kelas Kontrol

Kelas	Banyak Mhs	Postes				
		Min	Max	Mean	Sd	variance
Eksperimen	28	37,00	70,00	50,61	10,54	111,09
Kontrol	28	5,00	67,00	26,39	15,09	227,71

Berdasarkan Tabel 4.5 memberikan gambaran bahwa rata-rata kemampuan mahasiswa kelas eksperimen berbeda dengan kelas kontrol. Secara grafik

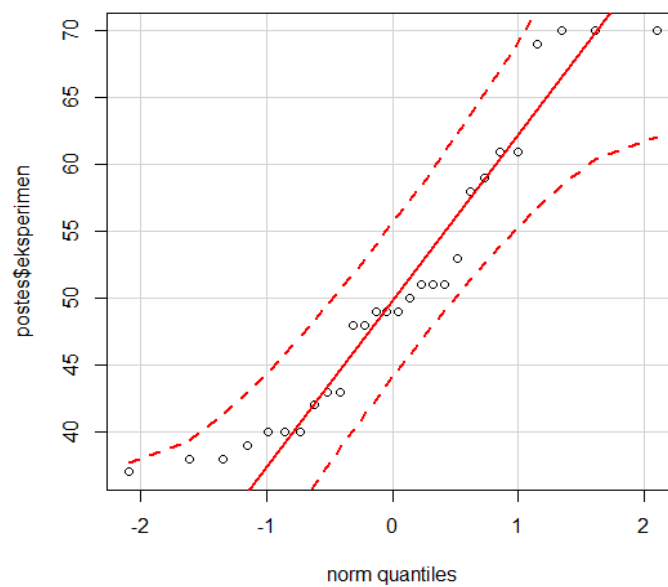
hasil postes kemampuan mahasiswa dapat dilihat bahwa terdapat perbedaan rata-rata kemampuan akhir mahasiswa antara kelas eksperimen dan kelas kontrol pada Gambar 4.4 . Berdasarkan Gambar 4.4 terlihat bahwa diagram batang yang berwarna biru yaitu diagram nilai postes kelas eksperimen rata-rata memiliki nilai yang lebih besar dibandingkan dengan diagram batang yang berwarna merah tua yaitu diagram batang nilai postes kelas kontrol.



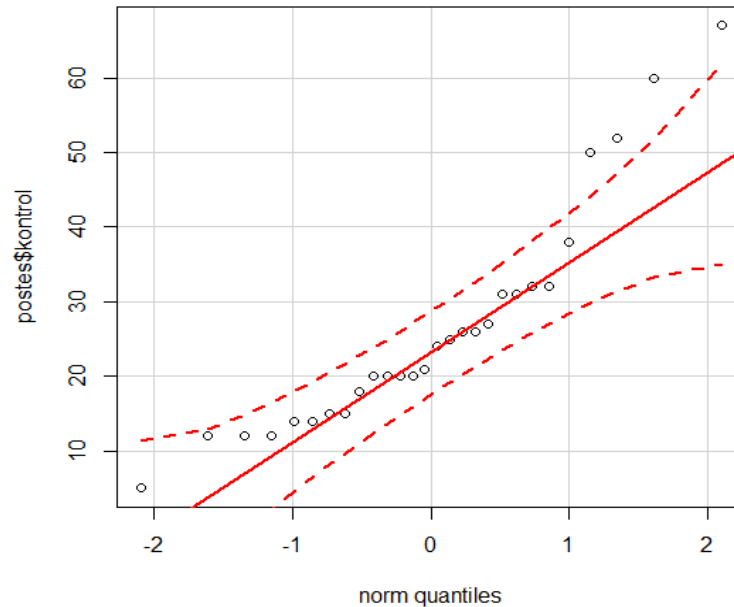
Gambar 4.4 Hasil Postes Kelas Eksperimen dan Kelas Kontrol

Langkah berikutnya yaitu uji normalitas data postes, berikut ini kami sajikan Q-Q plot uji kenormalan data postes kelas eksperimen dan kelas kontrol. Gambar 4.5 yaitu Q-Q plot dari postes kelas eksperimen sebaran data postes tersebar mendekati garis lurus, sehingga dapat diduga data berdistribusi normal. Sedangkan sebaran data postes kelas kontrol yang disajikan dalam bentuk Q-Q plot pada Gambar 4.6 terlihat bahwa sebaran data postes tersebar menjauh dari garis lurus, sehingga dapat diduga bahwa data sampel kelas kontrol

berasal dari populasi yang tidak berdistribusi normal. Karena ada salah satu kelas yang tidak berdistribusi normal, maka data postes kedua kelas tersebut berasal dari populasi yang tidak berdistribusi normal. Namun untuk mengetahui apakah prediksi tersebut itu benar, maka selanjutnya dilakukan uji statistik dengan langkah-langkah sebagai berikut.



Gambar 4.5 Uji Normalitas dengan Q-Q Plot data postes kelas eksperimen



Gambar 4.6 Uji Normalitas dengan Q-Q plot data postes kelas kontrol

4.1.2.2.1 Uji Normalitas Data Postes

Sama halnya dengan uji normalitas data pretes, langkah pertama yang dilakukan adalah menguji normalitas data postes kedua kelas untuk mengetahui apakah kedua kelas tersebut berdistribusi normal atau tidak. Untuk menguji normalitas data postes, digunakan uji statistik *Shapiro-Wilk*. Perumusan hipotesis pengujian normalitas data postes sebagai berikut:

H_0 : Data sampel berasal dari populasi yang berdistribusi normal.

H_1 : Data sampel berasal dari populasi yang tidak berdistribusi normal.

Dengan menggunakan taraf signifikansi (α) sebesar 0,05 kriteria .

pengujiannya adalah sebagai berikut:

1. Jika nilai *p-value* lebih dari atau sama dengan 0,05 maka H_0 diterima.

2. Jika nilai p -value kurang dari 0,05 maka H_0 ditolak.

Adapun hasil dari analisis uji normalitas skor postes kelas eksperimen dan kelas kontrol dengan uji *Shapiro-Wilk* disajikan dalam Tabel 4.6. Analisis secara lengkap dapat dilihat pada Lampiran G Uji Normalitas Data Postes halaman 95.

Tabel 4.6 Hasil Uji Normalitas Data Postes

Kelas	Shapiro-Wilk	
	W	p -value
Eksperimen	0,90	0,0146
Kontrol	0,88	0,0038

Dari Tabel 4.6 terlihat bahwa (p -value) uji *Shapiro-Wilk* untuk kelas eksperimen adalah 0,0146 dan kelas kontrol adalah 0,0038. Nilai p -value untuk kelas eksperimen yaitu 0,0146 maupun kelas kontrol yaitu 0,0038 keduanya kurang dari 0,05 maka H_0 ditolak, artinya data postes kelas kontrol berasal dari populasi yang tidak berdistribusi normal. Karena data tidak berdistribusi normal, maka tidak dilakukan uji homogenitas varians. Sehingga pengujian yang dilakukan selanjutnya adalah uji kesamaan dua rata-rata dengan menggunakan uji non-parametrik yaitu uji *Mann-Whitney*.

4.1.2.2.2 Uji Perbedaan Dua Rata-rata Data postes

Berdasarkan hasil dari uji normalitas yang telah dilakukan, diketahui bahwa data postes berasal dari populasi yang tidak berdistribusi normal, maka selanjutnya dilakukan analisis dengan menggunakan statistika nonparametrik, yaitu uji *Mann-Whitney*. Perumusan hipotesis untuk uji perbedaan dua rata-rata skor postes ini adalah sebagai berikut:

H0 : Kemampuan mahasiswa kelas eksperimen tidak berbeda daripada kemampuan mahasiswa kelas kontrol.

H1 : Kemampuan mahasiswa kelas eksperimen berbeda daripada kemampuan mahasiswa kelas kontrol.

Kriteria pengambilan keputusan untuk pengujiannya adalah sebagai berikut:

- a) Jika nilai *p-value* lebih besar atau sama dengan 0,05 maka H0 diterima.
- b) Jika nilai *p-value* lebih kecil dari 0,05 maka H0 ditolak.

Hasil analisis uji *Mann-Whitney* skor postes disajikan dalam Tabel 4.7. Analisis secara lengkap dapat dilihat pada Lampiran H Uji Beda Rata-rata Postes halaman 96.

Tabel 4.7 Hasil Uji Mann Whitney Data Postes

Mann-Whitney	Nilai
W	82,5
<i>p-value</i>	0,000

Dari Tabel 4.7 terlihat bahwa nilai *p-value* adalah 0,000 kurang dari 0,05 maka berdasarkan kriteria pengujian H0 ditolak. Dengan kata lain, kemampuan mahasiswa dalam pemahaman algoritma pada praktikum pemrograman terstruktur yang menggunakan IndoBlockly berbeda daripada mahasiswa yang menggunakan pembelajaran konvensional.

4.1.2.3 Analisis Data Kualitas Peningkatan Kemampuan Mahasiswa

Variabel yang diukur dalam penelitian ini adalah peningkatan kemampuan mahasiswa yang mendapat model pembelajaran menggunakan IndoBlockly. Kemampuan mahasiswa antara kelas eksperimen dengan kelas kontrol setelah

pembelajaran sudah diketahui pada analisis postes dengan kesimpulan bahwa kemampuan mahasiswa pada kelas eksperimen lebih baik daripada mahasiswa kelas kontrol. Untuk mengetahui kualitas peningkatan kemampuan mahasiswa kelas eksperimen dan kelas kontrol, dilakukan uji statistik deskriptif terhadap skor indeks gain tes kemampuan mahasiswa kelas eksperimen dan kelas kontrol sebagai berikut:

Tabel 4.8 Statistik Deskriptif Skor Indeks Gain Tes Kemampuan Mahasiswa Kelas Eksperimen dan Kelas Kontrol

Kelas	N	Mean Gain	Kriteria
Eksperimen	28	0,625128218	Sedang
Kontrol	28	0,159511988	Rendah

Dari Tabel 4.8 di atas, terlihat bahwa rata-rata indeks gain tes kemampuan mahasiswa kelas eksperimen adalah 0,63 dan rata-rata indeks gain tes kemampuan mahasiswa untuk kelas kontrol adalah 0,16. Berdasarkan kriteria indeks gain menurut Hake, ini berarti bahwa kualitas peningkatan kemampuan mahasiswa kelas eksperimen tergolong sedang sedangkan kelas kontrol tergolong rendah. Berikut ini adalah daftar persentase kualitas peningkatan kemampuan mahasiswa kelas eksperimen dan kelas kontrol.

Tabel 4.9 Daftar Persentase Kualitas Peningkatan Kemampuan Mahasiswa Kelas Eksperimen dan Kelas Kontrol

Kualitas Peningkatan Kemampuan Mahasiswa	Kelas Eksperimen		Kelas Kontrol	
	Jumlah Mhs	Persentase	Jumlah Mhs	Persentase
Tinggi	9	32,14	2	7,14
Sedang	17	60,71	3	10,71
Rendah	2	7,14	23	82,14

Berdasarkan Tabel 4.9 terlihat bahwa untuk kelas eksperimen 60,71% dari jumlah (28) mahasiswa termasuk dalam kategori sedang dan 32,14% lainnya termasuk dalam kategori tinggi. Hal ini menunjukkan bahwa peningkatan kemampuan mahasiswa yang terjadi di kelas eksperimen termasuk dalam kategori sedang, dimana lebih dari 50% (60,71%) mahasiswa kelas eksperimen mengalami peningkatan kemampuan yang sedang dan (32,14%) mengalami peningkatan yang tinggi cukup banyak jika dibandingkan dengan kelas kontrol yang hanya (7,14%). Analisis dan perhitungan indeks gain secara lengkap dapat dilihat pada Lampiran I Perhitungan Indeks Gain halaman 97.

4.1.2.4 Analisis Data Korelasi Pretes Postes Kelas Eksperimen

Sebelum menghitung koefisien korelasi tentu harus dilakukan uji normalitas terhadap data pretes postes kelas eksperimen terlebih dahulu. Hasil dari uji normalitas akan menentukan cara kita untuk menghitung koefisien korelasi. Jika data pretes postes kelas eksperimen berdistribusi normal maka akan digunakan rumus *Pearson* dan jika data terbukti tidak normal maka akan digunakan rumus *Spearman*.

Untuk menguji normalitas data pretes postes kelas eksperimen, digunakan uji statistik *Shapiro-Wilk*. Perumusan hipotesis pengujian normalitas data pretes postes sebagai berikut:

H_0 : Data sampel berasal dari populasi yang berdistribusi normal.

H1 : Data sampel berasal dari populasi yang tidak berdistribusi normal.

Dengan menggunakan taraf signifikansi (α) sebesar 0,05 kriteria .

pengujiannya adalah sebagai berikut:

1. Jika nilai *p-value* lebih dari atau sama dengan 0,05 maka H0 diterima.
2. Jika nilai *p-value* kurang dari 0,05 maka H0 ditolak.

Adapun hasil dari analisis uji normalitas skor pretes postes kelas eksperimen dengan uji *Shapiro-Wilk* disajikan dalam Tabel 4.10. Analisis secara lengkap dapat dilihat pada Lampiran J Uji Normalitas Data Pretes Postes Kelas Eksperimen halaman 98.

Tabel 4.10 Hasil Uji Normalitas Data Pretes Postes kelas eksperimen

Uji	Shapiro-Wilk	
	W	<i>p-value</i>
Pretes	0,83	0,00038
Postes	0,90	0,0146

Berdasarkan Tabel 4.10 dapat dilihat bahwa *p-value* pretes dan postes keduanya $< 0,05$ sehingga disimpulkan bahwa data tidak berdistribusi normal. Dari hasil uji normalitas ternyata terbukti data berasal dari distribusi tidak normal sehingga digunakan rumus *Spearman* untuk menghitung koefisien korelasi.

Sebelum dilakukan analisis korelasi menggunakan *Spearman* dilakukan uji hipotesis korelasi terlebih dahulu. Dengan uji hipotesis korelasi kita dapat mengetahui apakah memang ada korelasi antara hasil belajar dengan penggunaan IndoBlockly pada kelas eksperimen. Dikarenakan hasil uji normalitas menunjukkan

bahwa data tidak normal maka untuk menguji hipotesis korelasi digunakan uji non parametrik dua sampel berhubungan yaitu menggunakan *Wilcoxon*.

Perumusan hipotesis pengujian korelasi data pretes sebagai berikut:

H0 : Data pretes dan postes tidak saling berhubungan (berkorelasi).

H1 : Data pretes dan postes saling berhubungan (berkorelasi).

pengujiannya adalah sebagai berikut:

1. Jika nilai *p-value* lebih dari atau sama dengan 0,05 maka H0 diterima.
2. Jika nilai *p-value* kurang dari 0,05 maka H0 ditolak.

Adapun hasil dari analisis uji hipotesis korelasi pretes dan postes kelas eksperimen dengan uji *Wilcoxon* disajikan dalam Tabel 4.10. Analisis secara lengkap dapat dilihat pada Lampiran K Analisis Uji Hipotesis Korelasi dengan *Wilcoxon* halaman 99.

Tabel 4.11 Hasil Uji Korelasi Wilcoxon Data Pretes Postes kelas eksperimen

Wilcoxon	
<i>p-value</i>	0,000

Berdasarkan Tabel 4.11 dapat dilihat bahwa $p\text{-value} < 0,05$ sehingga disimpulkan bahwa ada korelasi data pretes dan postes kelas eksperimen. Setelah dilakukan uji normalitas dan uji korelasi ternyata diketahui bahwa data tidak normal dan juga ada korelasi maka langkah selanjutnya yaitu menghitung besaran korelasi yaitu dengan *Sprearman*.

Adapun hasil dari perhitungan koefisien korelasi menggunakan *Spearman* dapat dilihat pada Tabel 4.12.

Tabel 4.12 Koefisien Korelasi antara Pretes dan Postes Kelas Eksperimen

	Spearman	
	Pretes	Postes
Pretes	1.0000000	0.4298874
Postes	0.4298874	1.0000000

Berdasarkan hasil output R yang dapat dilihat pada table di atas, diperoleh koefisien korelasi sebesar 0,43 yang berarti bahwa hubungan antara pretes dan postes yang menunjukkan pengalaman belajar menggunakan IndoBlockly mempunyai hubungan yang positif. Berdasarkan Tabel 3.6 yaitu tabel kriteria korelasi angka 0,43 menunjukkan bahwa ada korelasi yang positif dengan kriteria sedang.

Untuk melakukan uji signifikansi koefisien dapat dilihat dari nilai signifikasinya.

Hipotesis :

H0: Kontribusi variabel independen (X) dan variabel dependen (Y) tidak signifikan (X dan Y independen).

H1: Kontribusi variabel independen (X) dan variabel dependen (Y) signifikan (X dan Y dependen) .

Dengan menggunakan taraf signifikansi 5% maka kriteria pengujiannya adalah ” Jika Nilai $p\text{-value} < \alpha = 0,05$ maka H0 ditolak”. Adapun hasil dari perhitungan uji signifikansi dapat dilihat pada Tabel 4.13. Analisis secara lengkap

dapat dilihat pada Lampiran L Analisis Korelasi, Uji Signifikansi Korelasi, dan Determinasi halaman 100.

Tabel 4.13 Uji Signifikansi Koefisien Korelasi antara Pretes dan Postes Kelas Eksperimen

Pretes Postes data eksperimen	Spearman	
	<i>p-value</i>	r
	0.02242	0.4298874

Berdasarkan tabel 4.13 diperoleh nilai signifikansi sebesar $0,02242 < 0,05$, maka H_0 ditolak sehingga dapat disimpulkan bahwa Kontribusi variabel independen (X) terhadap variabel dependen (Y) signifikan. Jadi dapat kita simpulkan bahwa ada korelasi positif pembelajaran menggunakan IndoBlockly sebesar 0,43 adalah signifikan dengan kriteria korelasi sedang.

Langkah selanjutnya menghitung besaran hubungan antara pretes dan postes yang menunjukkan pengalaman belajar menggunakan IndoBlockly yaitu dapat dilihat dari koefisien determinasi atau r^2 . Berdasarkan Tabel 4.12 dan Tabel 4.13 diketahui koefisien korelasi sebesar 0,43, dengan demikian koefisien determinasi atau r^2 diperoleh sebesar 0,1849 atau 18,49 %. Hal ini berarti IndoBlockly memberikan pengaruh sebesar 18,49% terhadap peningkatan hasil belajar mahasiswa pada mata kuliah pemrograman terstruktur.

BAB V

PENUTUP

5.1 Kesimpulan

Uji data pretes dari kelas kontrol dan kelas eksperimen menyatakan bahwa kedua kelas berasal dari populasi yang sama, setelah dilakukan percobaan yaitu dengan memberikan perlakuan yang berbeda terhadap kelas eksperimen menghasilkan data postes yang setelah dianalisis ternyata menunjukkan bahwa data postes kelas kontrol dan kelas eksperimen berbeda. Kelas eksperimen mempunyai skor indeks gain berkriteria sedang yaitu bernilai 0,63 dan koefisien korelasi 0,43. Angka koefisien korelasi 0,43 menunjukkan adanya korelasi positif berkriteria sedang antara penggunaan IndoBlockly terhadap hasil skor postes kelas eksperimen. Hasil uji signifikansi koefisien korelasi menunjukkan bahwa kontribusi variabel independen (X) yaitu penggunaan IndoBlockly terhadap variabel dependen (Y) yaitu hasil belajar mahasiswa adalah signifikan. Besarnya pengaruh penggunaan IndoBlockly terhadap peningkatan hasil belajar mahasiswa pada mata kuliah pemrograman terstruktur sebesar 18,49 %.

5.2 Saran

1. Asisten praktikum pemrograman terstruktur dapat menggunakan IndoBlockly dalam proses kegiatan mengajar karena memang pembelajaran menggunakan IndoBlockly terbukti lebih baik dibandingkan dengan menggunakan model pembelajaran yang lama.

2. Bagi pengembang IndoBlockly hasil penelitian ini adalah sebuah tantangan untuk dapat lebih berinovasi supaya IndoBlockly dapat memberikan pengaruh positif yang lebih besar lagi.
3. Peneliti lain diharapkan dapat melakukan penelitian dengan lingkup yang lebih besar dan sampel yang random sehingga hasil penelitian akan berlaku umum/general.

DAFTAR PUSTAKA

- Anggoro, T .2007. *Metode Penelitian*. Jakarta: Universitas Terbuka.
- Arikunto, Suharsimi .2006. *Prosedur Penelitian Suatu Pendekatan Praktik*. Jakarta:Rineka Cipta.
- Arikunto, Suharsimi. 2007. *Manajemen Penelitian*. Jakarta: Rineka Cipta.
- Arsyad, Azhar. 2004. *Media Pembelajaran*. Jakarta : PT. Raja Grafindo Persada.
- Begel, A,. 1996. *LogoBlocks: A Graphical Programming Language for Interacting with the World*.MIT Media Laboratory
- Bloom B. S.1956. *Taxonomy of Educational Objectives, Handbook I: The Cognitive Domain*. New York: David McKay Co Inc.
- Boshernitsan, M., Downes, M.2004. *Visual Programming Languages: A Survey*.Computer Science Division (EECS) University of California Berkeley.
- Dahar, R.W. 1989. *Teori-teori Belajar*. Bandung: Erlangga.
- Emzir.2008.*Metodologi Penelitian Pendidikan*. Jakarta: Rajarafindo Persada.
- Ena,O.T.,2007.*Membuat Media Pembelajaran Interaktif dengan Piranti Lunak Presentasi*.Yogyakarta: ILCIC, Universitas Sanata Dharma.
- Erman Suherman dan Yaya Sukjaya K.1990.*Evaluasi Pendidikan Matematika*.Bandung: Wijayakusumah.
- Esteves, M. and Mendes, A., ” *A Simulation Tool to Help Learningof Object Oriented Programming Basics*”. In Proceedings of the 34th ASEE/IEEE Frontiers in Education Conference, Savannah,Georgia, USA, October 2004, 20-23.
- Gay, L.R. dan Diehl, P.L.1992. *Research Methods for Business and Management*. New York :MacMillan Publishing Company.
- Gomes, A. and Carmo, L. and Bigotte, E. and Mendes, A., "*Mathematics and programming problem solving*", 3rd E-Learning Conference – Computer Science Education, Coimbra, September 2006.
- Hundhausen, J. Brown, “*An experimental study of the impact of visual semantic feedback on novice programming*”, Journal of Visual Language and Computing, Vol. 18, 2007, 537-559.

- Jihad dan Haris. 2009. *Evaluasi Pembelajaran*. Yogyakarta: Multi Pressindo
- Kelleher, C. & Pausch, R.,” *Lowering the barriers to programming: a taxonomy of programming environments and languages for novice programmers*”. ACM Computing Surveys, 37(2), 88-137.
- Lahtinen, E., Mutka, K. A., and Jarvinen, H. M., “*A Study of the difficulties of novice programmers*”, In Proceedings of the 10th annual SIGSCE conference on Innovation and technology in computer science education (ITICSE 2005), Monte da Caparica, Portugal, June 27-29, 2005, ACM Press, New York, NY, pp. 14-18.
- Lethbridge, C.; Diaz-Herrera, J.; LeBlanc, Jr.; Thompson, B., “*Improving software practice through education: Challenges and future trends*”, Future of Software Engineering, (FOSE apos;07), May 2007 Page(s):12 – 28.
- Mafrur, R., “*(IndoBlockly) Visual Programming Editor for Indonesia*”, In Proceedings of Seminar Nasional Ilmu Komputer Universitas Diponegoro (SNIK UNDIP 2012), Semarang, September 15, 2012, Graha Ilmu, Yogyakarta, 155-160.
- Maloney, J., Peppler, K., Kafai, Y., Resnick, M., and Rusk, N., “*Programming by Choice: Urban Youth Learning Programming with Scratch*”. Proceedings of the 39th SIGCSE technical symposium on Computer science education, March 12-15, 2008, Portland, OR, USA .
- Miliszewska, I., Tan, G., “*Befriending Computer Programming: A Proposed Approach to Teaching Introductory Programming*”, Journal of Issues in Informing Science & Information Technology, Vol. 4, 2007, 277-289.
- Mustika Danang. 2009. *Matematika Dasar untuk Perguruan Tinggi*. Bandung : Rekayasa Sains.
- Navarro-Prieto, R., Jose J.Can., “*Are visual programming languages better? The role of imagery in program comprehension*”. Departamento de Psicologn H Experimental, Facultad de Psicologn H a, Universidad de Granada.
- Pennington, N. 1987. *Stimulus structures and mental representation in expert comprehension of computer programs*. Cognitive Psychology, 19, 295-341.
- Robins, A, J. Rountree, and N. Rountree. 2003. *Learning and teaching programming: A review and discussion*. Computer Science Education, 13(2):137–172.

- Roscoe, J T. 1992. *Fundamental Research Statistics for the Bahavior al Sciences*. Second Edition Holt. New York :Rinehart and Winston.
- Sugiyono. 2003. *Metode Penelitian Administrasi*. Bandung : CV Alfabeta.
- Sugiyono. 2007. *Statistik untuk Penelitian*. Bandung:Alfabeta.
- Winkel, W.S. 1983 .*Psikologi Pendidikan dan Evaluasi Belajar*. Jakarta:Gramedia
- Winslow, L.E. 1996. *Programming pedagogy – A psychological overview*. SIGCSE Bulletin, 28,17–22.
- Yusniati. 2009. *Pengaruh Model Penemuan Terbimbing Berbasis Konstektual untuk meningkatkan Kemampuan penalaran Matematis Siswa SMP*. Skripsi:Tidak diterbitkan.

Lampiran A Pelaksanaan Penelitian

No	Tanggal	Kegiatan	Keterangan
1.	13/09/2012	Uji Coba Soal (Test Uji Coba)	Hasil/data akan digunakan untuk uji coba instrumen penelitian
2.	17/09/2012	Pretes	Dilakukan pada kedua kelas (kelas kontrol dan kelas eksperimen)
3.	24/09/2012	Pertemuan I	Pengenalan IndoBlockly, Game maze, dan variabel/tipe data
4.	01/10/2012	Pertemuan II	Pendalaman variabel/tipe data, input output, array
5.	08/10/2012	Pertemuan III	Kondisi dan pembuatan aplikasi
6.	15/10/2012	Pertemuan IV	Kontrol, looping
7.	22/10/2012	Postes	Dilakukan pada kedua kelas (Kelas kontrol dan kelas eksperimen)

Lampiran B Analisis Uji Coba Instrumen

Jumlah mahasiswa peserta uji coba soal = 30 mahasiswa

Total soal = 10 soal

Skor tiap soal = 10

Total Skor = 100

No	Nomor Soal										Jumlah
	1	2	3	4	5	6	7	8	9	10	
	Skor Yang Dicapai mahasiswa										Skore
1	8	8	8	9	8	10	8	7	6	3	67
2	5	5	5	5	5	5	5	1	1	2	34
3	5	3	5	5	5	5	3	1	1	4	32
4	5	4	5	5	5	5	4	1	1	0	30
5	5	3	5	5	5	5	3	1	1	2	30
6	6	1	5	4	5	5	3	0	4	1	28
7	5	1	5	5	5	5	2	1	1	4	29
8	2	2	5	4	6	5	3	1	2	1	29
9	5	1	2	5	5	3	4	1	3	0	24
10	3	1	2	3	5	5	2	1	3	1	23
11	5	2	2	4	3	5	1	1	2	0	20
12	5	0	5	5	5	5	0	0	0	2	22
13	3	3	3	3	3	3	3	1	3	1	23
14	1	4	3	3	3	3	3	3	1	0	23
15	3	3	3	3	3	3	3	0	3	1	22
16	3	3	3	3	3	3	3	2	0	1	21
17	3	3	3	3	3	3	3	1	1	1	21
18	2	4	2	0	2	2	4	2	1	1	18
19	3	1	3	3	3	3	1	1	1	2	18
20	0	2	3	3	3	3	2	1	1	0	18
21	3	2	3	3	2	3	2	1	1	0	17
22	3	2	2	3	2	3	2	1	1	0	16
23	3	2	2	3	2	3	3	0	2	0	17
24	0	1	3	3	3	3	1	1	1	1	17
25	3	1	3	3	3	3	1	0	2	0	16
26	0	1	2	2	2	2	1	2	1	2	15
27	2	1	2	2	1	2	1	1	1	1	12
28	2	0	2	2	2	2	4	0	1	0	13
29	1	0	2	2	1	2	0	2	1	1	11
30	0	1	1	1	1	1	1	0	0	2	8

Hasil Analisis Tingkat Kesukaran dan Daya Beda

Nomor Soal	Tingkat Kesukaran		Daya Beda	
	Indeks	Tafsiran	Indeks	Tafsiran
1	0.44	Soal Sedang	0.43	Daya Beda Baik
2	0.27	Soal Sedang	0.24	Daya Beda Sedang
3	0.46	Soal Sedang	0.33	Daya Beda Sedang
4	0.49	Soal Sedang	0.35	Daya Beda Sedang
5	0.49	Soal Sedang	0.47	Daya Beda Baik
6	0.51	Soal Sedang	0.39	Daya Beda Sedang
7	0.35	Soal Sedang	0.28	Daya Beda Sedang
8	0.15	Soal Sulit	0.09	Daya Beda jelek
9	0.23	Soal Sulit	0.16	Daya Beda jelek
10	0.19	Soal Sulit	0.19	Daya Beda jelek

Analisis validitas instrumen soal

Soal 1	10946	5036	46844	Interpretasi validitas
		235906384	15359.24425	
		0.712665273		tinggi
Soal 2	7450	2525	46844	
		118281100	10875.71147	
		0.685012656		tinggi
Soal 3	8946	2049	46844	
		95983356	9797.109574	
		0.913126462		sangat tinggi
Soal 4	6616	2324	46844	
		108865456	10433.86103	
		0.634089335		tinggi
Soal 5	9736	2444	46844	
		114486736	10699.84748	
		0.909919513		sangat tinggi
Soal 6	10240	2540	46844	
		118983760	10907.96773	
		0.93876332		sangat tinggi
Soal 7	7244	2324	46844	
		108865456	10433.86103	
		0.694277984		tinggi
Soal 8	4240	1505	46844	
		70500220	8396.440913	
		0.504975863		sedang
Soal 9	4468	1421	46844	
		66565324	8158.757503	
		0.547632406		sedang
Soal 10	3086	1124	46844	
		52652656	7256.214991	
		0.425290596		sedang

Lampiran C Soal Pretes dan Postes

Petunjuk :

- a. Kerjakanlah Soal berikut di lembar balik dari soal.
- b. Kerjakanlah soal secara berurutan dengan memberikan nomor pada setiap jawaban, jika anda tidak mengetahui jawabanya maka kosongkan saja.

1. FlowChart

- a. apa yang anda ketahui tentang flowchart?
- b. Berilah contoh flowchart yang anda kenal?
- c. Buatlah Flowchart untuk kasus berikut ini. Jurusan teknik Informatika UIN Sunan Kalijaga Yogyakarta pada tahun 2012/2013 mengadakan ujian tulis masuk teknik informatika dengan syarat Untuk bisa mendaftar ikut ujian tulis siswa harus mempunyai nilai matematika dan bahasa inggris ≥ 80 di ijazah SMA/SMK/MA, jika tidak maka tidak bisa mendaftar. Dan UIN hanya menerima 60 mahasiswa baru di Teknik Informatika untuk tahun 2012/2013.

2. Tipe Data

- a. Apa yang anda ketahui tentang tipe data?
- b. Apa yang anda ketahui tentang beberapa hal di bawah ini:
karakter, bilangan asli, bilangan cacah, bilangan negatif, bilangan bulat, bilangan rasional, bilangan irasional, int, float, string, int[], char, Boolean.

3. Array

- a. Apa yang anda ketahui mengenai *array, array satu dimensi, array dua dimensi*?

- b.

Diketahui:

```
int A[4] = {1,4,5,3};
int B[2][2] = {{9,8},{11,23}};
```

Berapakah nilai

A[1] =?

A[2] =?

A[0] =?

B[0][0] =?

B[0][1] =?

4. Input Output

- Apa yang anda tahu tentang input, proses, output?
- Coba Jelaskan ketika kita ingin menghitung luas persegi panjang, apa saja yang harus kita inputkan, bagaimana proses perhitungannya dan apa outputnya?
- Apa yang anda ketahui tentang *printf*, *cout*, *cin*, *scanf*, *write*, *writeln*.

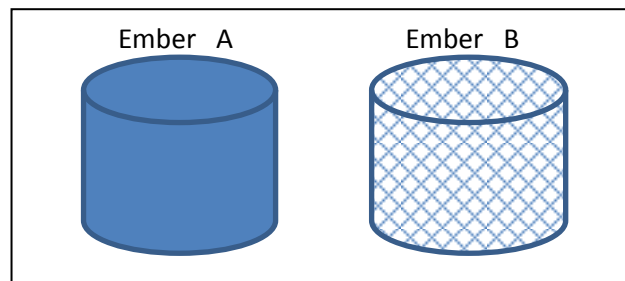
5. Kondisi, Perulangan

Apa yang anda ketahui tentang hal dibawah ini :

- if*, *then*, *else*
- for*, *while*, *do while*

6. Logika

- Diberikan 2 buah ember A dan B, ember A berisi larutan berwarna merah, ember B berisi larutan berwarna biru. Bagaimana cara menukar isi kedua ember itu sedemikian sehingga ember A berisi larutan warna biru dan ember B berisi larutan berwarna merah. (Buatlah urutan/prosedur langkah-langkahnya)



- Perhatikan Soal dibawah ini:

```
A=9;
B=6;
C=7;
A=B+C;
B=A+B;

C=A+B;
```

7. Ganjil Genap

- Apa yang anda ketahui tentang *div* dan *mod* ?

- b. Jika anda diminta untuk menampilkan bilangan ganjil kurang dari 100, sebagai orang teknik informatika apa yang akan anda lakukan, jelaskan?
- c. Jika anda diminta untuk menampilkan bilangan genap kurang dari 100, sebagai orang teknik informatika apa yang akan anda lakukan, jelaskan?

Soal 8,9,10 Tidak digunakan sesuai dengan analisis uji coba instrumen.

8. Bilangan Terbesar

Anda diberikan sederet bilangan {2,0,-6,4,9,8,-33,5} Sebagai orang TI coba buat alur/proses bagaimana cara untuk mencari bilangan yang paling besar?

9. Bilangan Prima

Sebagai orang TI apa yang akan anda lakukan jika diminta untuk membuat program yang bisa menampilkan deret bilangan prima < 100 ?

10. Faktorial

Sebagai orang TI apa yang akan anda lakukan jika diminta untuk membuat program factorial?, ketika user menginputkan bilangan berapapun, maka program akan menampilkan hasil dari factorial bilangan tersebut. Contoh user menginputkan angka 3, maka outpunya adalah $3! = 3 \times 2 \times 1 = 6$.

Lampiran D Perolehan Data dan Analisis Diskriptif

	Tahun Akademik :	2012/2013 - SEMESTER GANJIL		
	Prodi :	TEKNIK INFORMATIKA		
No	NIM	Nama Mahasiswa	Pretes	Postes
1	8650050	FAJRIA ANTONI	Tidak ikut pretes dan post	
2	8650069	SHANDY VEGA PRIMANDA	16	20
3	9650024	DISSA DAMALITA	8	5
4	9650042	ITA DEVIYANTI	13	14
5	9650047	IZZA ULINNUHA	18	12
6	9650058	OKKI PUTRAWAN	23	15
7	10650053	PANDU SETYOAJI N	20	26
8	12650001	Puguh Jayadi	26	32
9	12650002	Septri Kismarini	29	52
10	12650003	Siti Helmiyah	9	14
11	12650004	Agung Pambudi	38	32
12	12650005	Winda Rizky Astuti	7	12
13	12650006	M. Weddy Sumbogo	14	20
14	12650007	Wahyu Aprilynasari	23	31
15	12650008	Niki Min Hidayati Robbi	21	31
16	12650009	Irfan Afif Mustofa	15	15
17	12650010	Rian Wiguna	7	12
18	12650011	Nur Indah Fitriarningsih	21	38
19	12650012	JAKSANA ARIF FURKAN	18	18
20	12650013	AMI MEGANTARA P	18	20
21	12650014	YOGA PRATAMA	37	50
22	12650015	MUHAMMAD AFIF MUHTAR	17	26
23	12650016	KHOIRUL FUADI	10	27
24	12650017	LUQMAN HAKIM	Keluar *	
25	12650018	FAJAR NURROHMAT	16	24
26	12650019	ALFIAN NUR JAYANTO	18	20
27	12650020	AFHMULHASAN NOOR R	12	21
28	12650021	AHMAD MUSTAFID	50	67
29	12650022	ROYANUL FITRON	19	25
30	12650023	DANANG PURWOKO PUTRO	16	60
31	12650024	MOCHAMAD NOOR SYAMSU	Keluar *	
32	12650025	FERDIAN NOOR PAMBUDI	18	51
33	12650026	A.S WAHID FAIZIN	17	59
34	12650027	ELVANISA AYU MUHSINA	34	70
35	12650028	MUHAMMAD NUR ALFANI	6	58
36	12650029	DANANG SUDRAJAT	39	38
37	12650030	MUHAMMAD DZULFIKAR FAUZI	10	39

38	12650031	ANNISA D OKTAVIANITA	6	51
39	12650032	ALIF AZIZ	26	49
40	12650033	M MURAH PAMUJI	21	61
41	12650034	M. ZIDNAL FALAH	15	37
42	12650035	SULTON DAUD UL M	19	53
43	12650036	MUH ARFA AMRIZAL	65	70
44	12650037	LINA NUR LATIFAH	20	49
45	12650038	RIZKI RAMADHAN	13	43
46	12650039	LUTFIA LILIN KHARIROH	5	38
47	12650040	RIO KIRNANDA	23	69
48	12650043	YAUMI HASHIFUL INSI	17	40
49	12650045	MUHAMMAD SYAEFUL BAHRY	15	50
50	12650048	MR. SULKIFLI POHJI	5	43
51	12650051	MOH. MUBAROK DAWAM	9	49
52	12650064	IRWANTO	34	42
53	12650070	FARIS NUR ZAMAN	14	40
54	12650071	WAHIB RAMADHAN	21	61
55	12650072	ANWARUDDIN KAMAL IBRAHIM	13	48
56	12650084	AGUNG SETIYO PAMBUDI	17	51
57	12650091	AKH. BAINI TASLIHUDIN	19	48
58	12650096	MIRZA FIRDAUS AVECINNA	13	40
59	12650097	ALFIAN GAUTAMA HERMAN	27	70

Data = dataanalysis.RData

Statistics=>Summaries=>Active Data Set

```
> tapply(dataanalysis$pretes,
list(group=dataanalysis$group), mean, na.rm=TRUE)
group
kontrol eksperimen
19.25000 19.32143
> summary(dataanalysis)
      pretes      postes      group
Min.   : 5.00   Min.   : 5.00   kontrol   :28
1st Qu.:13.00   1st Qu.:23.25   eksperimen:28
Median :17.50   Median :39.50
Mean   :19.29   Mean   :38.50
3rd Qu.:21.50   3rd Qu.:51.00
Max.   :65.00   Max.   :70.00
```

Statistics=>Summaries=>Table of Statistics

```
> # Table for postes:
> tapply(dataanalysis$postes,
list(group=dataanalysis$group), sd, na.rm=TRUE)
group
```

```

    kontrol eksperimen
15.09069    10.54212
> # Table for pretes:
> tapply(dataanalisis$pretes,
list(group=dataanalisis$group), sd, na.rm=TRUE)
group
    kontrol eksperimen
9.785193  12.457494

```

Jumlah Mahasiswa yang Memperoleh skor X berdasarkan Soal Pretes dan Postes

Pretes

No	Soal	Skor										total mhs
		1	2	3	4	5	6	7	8	9	10	
1	Flowchart		8	13	4	9	9	3	6	2	2	56
2	Tipe Data			17	23	8	3	4			1	56
3	Array		1	16	18	5	9	6			1	56
4	Input Output					38		3	6	8	1	56
5	Kondisi, Perulangan		3	5	35	10		2			1	56
6	Logika			6		46	2				2	56
7	Ganjil Genap				44	3	3		5		1	56

Postes

No	Soal	Skor										total mhs
		1	2	3	4	5	6	7	8	9	10	
1	Flowchart		2	1	4	12	9	10	6	8	4	56
2	Tipe Data		1	6	6	15	2	12	6	5	3	56
3	Array		2	16	9	18		5	2	1	3	56
4	Input Output					2	4	18	14	15	3	56
5	Kondisi, Perulangan			5	13	24		9	1	1	3	56
6	Logika					3	7	4	12	3	27	56
7	Ganjil Genap				13	17		5		16	5	56

Lampiran E Uji Normalitas Data Pretes

```
Dataset = pretest.RData
```

```
Statistics=>Summaries=>Shapiro-Wilk Test of Normality
```

```
> load("/Users/macintosh/GIT/skripsi/R  
analisis/pretes.RData")
```

```
> shapiro.test(pretes$eksperimen)
```

```
Shapiro-Wilk normality test
```

```
data:  pretes$eksperimen  
W = 0.8301, p-value = 0.0003812
```

```
> shapiro.test(pretes$kontrol)
```

```
Shapiro-Wilk normality test
```

```
data:  pretes$kontrol  
W = 0.8746, p-value = 0.003057
```


Lampiran F Uji Beda Rata-rata Pretest

Data = dataanalysis.RData

Statistics=>Nonparametric tests=>Two-sample Wilcoxon test

```
Rcmdr> load("/Users/macintosh/GIT/skripsi/R/R
analysis/dataanalysis.RData")
RcmdrMsg: [7] NOTE: The dataset dataanalysis has 56
rows and 3 columns.
```

```
Rcmdr> tapply(dataanalysis$pretes, dataanalysis$group,
median, na.rm=TRUE)
      kontrol eksperimen
              18          17
```

```
Rcmdr> wilcox.test(pretes ~ group,
alternative='two.sided', exact=TRUE,
Rcmdr+   correct=FALSE, data=dataanalysis)
```

Wilcoxon rank sum test

```
data:   pretes by group
W = 411.5, p-value = 0.749
alternative hypothesis: true location shift is not
equal to 0
```

Lampiran G Uji Normalitas Data Postes

```
Data= postes.RData
```

```
Statistics=>Summaries=>Shapiro-Wilk test of Normality
```

```
> load("/Users/macintosh/GIT/skripsi/R  
analisis/postes.RData")
```

```
> shapiro.test(postes$eksperimen)
```

```
Shapiro-Wilk normality test
```

```
data: postes$eksperimen  
W = 0.9045, p-value = 0.01463
```

```
> shapiro.test(postes$kontrol)
```

```
Shapiro-Wilk normality test
```

```
data: postes$kontrol  
W = 0.8792, p-value = 0.003855
```

Lampiran H Uji Beda Rata-rata Postes

Data = dataanalysis.RData

Statistics=>Nonparametric tests=>Two-sample Wilcoxon test

```
> tapply(dataanalysis$postes, dataanalysis$group,
median, na.rm=TRUE)
```

```
    kontrol eksperimen
      22.5         49.0
```

```
> wilcox.test(postes ~ group, alternative="two.sided",
data=dataanalysis)
```

Wilcoxon rank sum test with continuity correction

data: postes by group

W = 82.5, p-value = 3.866e-07

alternative hypothesis: true location shift is not
equal to 0

Lampiran I Perhitungan Index Gain

NO	pretes	postes	group	A	B	G	
1	16	20	kontrol	4	54	0.074074074	Rendah
2	8	5	kontrol	-3	62	-0.048387097	Rendah
3	13	14	kontrol	1	57	0.01754386	Rendah
4	18	12	kontrol	-6	52	-0.115384615	Rendah
5	23	15	kontrol	-8	47	-0.170212766	Rendah
6	20	26	kontrol	6	50	0.12	Rendah
7	26	32	kontrol	6	44	0.136363636	Rendah
8	29	52	kontrol	23	41	0.56097561	Sedang
9	9	14	kontrol	5	61	0.081967213	Rendah
10	38	32	kontrol	-6	32	-0.1875	Rendah
11	7	12	kontrol	5	63	0.079365079	Rendah
12	14	20	kontrol	6	56	0.107142857	Rendah
13	23	31	kontrol	8	47	0.170212766	Rendah
14	21	31	kontrol	10	49	0.204081633	Rendah
15	15	15	kontrol	0	55	0	Rendah
16	7	12	kontrol	5	63	0.079365079	Rendah
17	21	38	kontrol	17	49	0.346938776	Sedang
18	18	18	kontrol	0	52	0	Rendah
19	18	20	kontrol	2	52	0.038461538	Rendah
20	37	50	kontrol	13	33	0.393939394	Sedang
21	17	26	kontrol	9	53	0.169811321	Rendah
22	10	27	kontrol	17	60	0.283333333	Rendah
23	16	24	kontrol	8	54	0.148148148	Rendah
24	18	20	kontrol	2	52	0.038461538	Rendah
25	12	21	kontrol	9	58	0.155172414	Rendah
26	50	67	kontrol	17	20	0.85	tinggi
27	19	25	kontrol	6	51	0.117647059	Rendah
28	16	60	kontrol	44	54	0.814814815	tinggi
rata2	19.25	26.39286				0.159511988	Rendah
1	18	51	eksperimen	33	52	0.634615385	Sedang
2	17	59	eksperimen	42	53	0.79245283	tinggi
3	34	70	eksperimen	36	36	1	tinggi
4	6	58	eksperimen	52	64	0.8125	tinggi
5	39	38	eksperimen	-1	31	-0.032258065	Rendah
6	10	39	eksperimen	29	60	0.483333333	Sedang
7	6	51	eksperimen	45	64	0.703125	tinggi
8	26	49	eksperimen	23	44	0.522727273	Sedang
9	21	61	eksperimen	40	49	0.816326531	tinggi
10	15	37	eksperimen	22	55	0.4	Sedang
11	19	53	eksperimen	34	51	0.666666667	Sedang
12	65	70	eksperimen	5	5	1	tinggi
13	20	49	eksperimen	29	50	0.58	Sedang
14	13	43	eksperimen	30	57	0.526315789	Sedang
15	5	38	eksperimen	33	65	0.507692308	Sedang
16	23	69	eksperimen	46	47	0.978723404	tinggi
17	17	40	eksperimen	23	53	0.433962264	Sedang
18	15	50	eksperimen	35	55	0.636363636	Sedang
19	5	43	eksperimen	38	65	0.584615385	Sedang
20	9	49	eksperimen	40	61	0.655737705	Sedang
21	34	42	eksperimen	8	36	0.222222222	Rendah
22	14	40	eksperimen	26	56	0.464285714	Sedang
23	21	61	eksperimen	40	49	0.816326531	tinggi
24	13	48	eksperimen	35	57	0.614035088	Sedang
25	17	51	eksperimen	34	53	0.641509434	Sedang
26	19	48	eksperimen	29	51	0.568627451	Sedang
27	13	40	eksperimen	27	57	0.473684211	Sedang
28	27	70	eksperimen	43	43	1	tinggi
	19.32142857	50.60714286				0.625128218	Sedang

Lampiran J Uji Normalitas Data Pretes Postes Kelas Eksperimen

```
Data= pre_pos_exp.RData
```

```
Statistics=>Summaries=>Correlation matrix=>Spearman  
rank-order
```

```
load("/Users/macintosh/GIT/skripsi/R  
analisis/pre_pos_exp.RData")
```

```
> showData(prepostext, placement='-20+200',  
font=getRcmdr('logFont'),  
+   maxwidth=80, maxheight=30)
```

```
> fix(prepostext)
```

```
> shapiro.test(prepostext$pos)
```

```
Shapiro-Wilk normality test
```

```
data:  prepostext$pos  
W = 0.9045, p-value = 0.01463
```

```
> shapiro.test(prepostext$pre)
```

```
Shapiro-Wilk normality test
```

```
data:  prepostext$pre  
W = 0.8301, p-value = 0.0003812
```

Lampiran K Analisis Uji Hipotesis Korelasi dengan Wilcoxon

Data= pre_pos_exp.RData

Statistics=>Summaries=>Non Parametrik Test=>Paired-Wilcoxon Test

```
Rcmdr> load("/Users/macintosh/GIT/skripsi/R/R
analisis/pre_pos_exp.RData")
RcmdrMsg: [2] NOTE: The dataset prepostext has 28 rows
and 2 columns.
```

```
Rcmdr> median(prepostext$pre - prepostext$pos,
na.rm=TRUE) # median difference
[1] -33.5
```

```
Rcmdr> wilcox.test(prepostext$pre, prepostext$pos,
alternative='two.sided',
Rcmdr+   paired=TRUE)
```

Wilcoxon signed rank test with continuity correction

```
data: prepostext$pre and prepostext$pos
V = 1, p-value = 4.428e-06
alternative hypothesis: true location shift is not
equal to 0
```

Lampiran L Analisis Korelasi, Uji Signifikansi Korelasi, dan Determinasi

Data= pre_pos_exp.RData

Statistics=>Summaries=>Correlation matrix=>Spearman rank-order

```
load("/Users/macintosh/GIT/skripsi/R
analisis/pre_pos_exp.RData")
```

```
> # Spearman rank-order correlations
```

```
> cor(prepostext[,c("pos","pre")], use="complete.obs",
method="spearman")
```

```
      pos      pre
pos 1.0000000 0.4298874
pre 0.4298874 1.0000000
```

```
> cor.test(prepostext[,c("pre")],
prepostext[,c("pos")], method="spearman")
```

Spearman's rank correlation rho

```
data: prepostext[, c("pre")] and prepostext[,
c("pos")]
```

S = 2083.191, p-value = 0.02242

alternative hypothesis: true rho is not equal to 0

sample estimates:

```
rho
0.4298874
```

Determinasi = r^2

Determinasi = $(0.43)^2 = 0.1849$ atau 18,49 %