



BAB 1

PENGANTAR DASAR ALGORITMA DAN PEMROGRAMAN

BAB 1

PENGANTAR ALGORITMA DAN

PEMROGRAMAN

Capaian Pembelajaran

Setelah mempelajari materi dalam bab ini, mahasiswa diharapkan mampu menjelaskan dan menyampaikan pendapat tentang pengantar dasar algoritma pemrograman

Pokok Bahasan

1. Konsep algoritma dan pemrograman
2. Penggunaan Algoritma Pemrograman di kehidupan sehari hari
3. Mengetahui Bahasa Java dalam Pemrograman
4. Instalasi Aplikasi Raptor dan Visual Studio Code

Evaluasi Pembelajaran

Soal Latihan Pengantar Algoritma dan Pemrograman

Referensi

1. Abdul Kadir, 2012, Algoritma & Pemrograman Menggunakan Java, Penerbit Andi Publisher, Jakarta
2. Munir Renaldi, 2011, Algoritma dan Pemrograman dalam Bahasa Pascal dan C (Edisi Revisi), Bandung: Informatika Bandung.
3. Ir. Sjukani, Moch. dkk, 2019, Modul Matakuliah Algoritma dan Struktur Data I, Jakarta: Universitas Budi Luhur

Pre Test

Prosedur Pengantar Algoritma dan Pemrograman

1. Apa yang dimaksud dengan Algoritma?
2. Berikan contoh penggunaan algoritma dalam kehidupan sehari-hari?
3. Sebutkan dan jelaskan tiga buah struktur dasar yang menyusun suatu algoritma?
4. Berikan penjelasan terkait Bahasa java dalam pemrograman?
5. Apa yang anda ketahui terkait aplikasi visual studio code?

Algoritma merupakan penyusunan aspek proses logika dari suatu pemecahan masalah dalam pemrograman komputer. Program dan algoritma ini saling berkaitan karena dalam pemrograman yang, algoritma adalah langkah pertama yang harus ditulis sebelum menuliskan program. Bab ini berisi penjelasan mengenai pengantar algoritma dan pemrograman, yang memiliki beberapa sub bab, diantaranya: Konsep algoritma dan pemrograman, Penggunaan Algoritma Pemrograman di kehidupan sehari hari, Mengenal Bahasa Java dalam Pemrograman, dan Instalasi Aplikasi Raptor dan Visual Studio Code.

1.1 Konsep Dasar Algoritma

Konsep dasar algoritma secara sederhana dapat diartikan sebagai urutan atau langkah-langkah yang sistematis untuk menyelesaikan sebuah masalah dimana algoritma sendiri berarti solusi. Algoritma dan pemrograman adalah dua hal yang berhubungan erat, program komputer pada dasarnya mengimplementasikan sebuah algoritma.

Dalam kehidupan sehari-hari sebenarnya banyak proses yang bisa dikatakan suatu algoritma. Contoh sederhananya adalah mengambil uang tunai di ATM, membuat masakan, memesan makanan di ojek online, cara merakit mainan, install aplikasi. Algoritma berisi urutan langkah untuk menyelesaikan persoalan yang telah diberikan, maka dari terdapat beberapa ciri atau karakteristik penting yang harus dimiliki oleh algoritma, di antara lain yaitu :

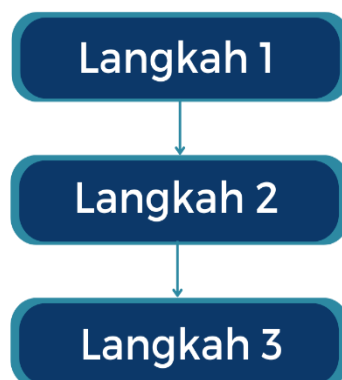
1. Finiteness: Semua proses dalam algoritma harus berakhir setelah sejumlah langkah dilakukan
2. Definiteness: Algoritma ditulis dengan bahasa yang standar secara sistematis dan rapi sehingga tidak menimbulkan arti ganda
3. Input: Algoritma bisa jadi tidak memiliki input atau memiliki beberapa input, dimana input ini merupakan sebuah parameter nilai yang diberikan di awal sebelum algoritma akan diproses.

4. Output: Algoritma bisa jadi tidak memiliki output atau memiliki beberapa output, dimana output merupakan pesan atau suatu besaran yang yang berhubungan dengan masukan
5. Effectiveness: Algoritma ditulis dengan format yang mudah dipahami, sederhana, dan mudah diimplementasikan ke dalam bahasa pemrograman dan dibutuhkan terdefinisi dengan jelas

Pada dasarnya terdapat tiga buah struktur dasar yang menyusun suatu algoritma. Ketiganya adalah sebagai berikut:

1. Struktur sekuensial (Sequence)

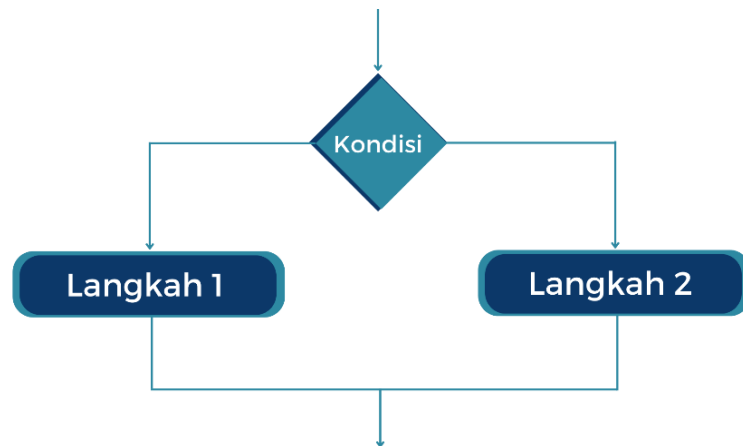
Struktur algoritma yang instruksinya dikerjakan setelah instruksi sebelumnya selesai dikerjakan. Struktur algoritma ini terdiri dari satu atau lebih intruksi dimana setiap intruksi yang diberikan akan dikerjakan secara sistematis atau berurutan sesuai dengan urutan penulisan algoritma tersebutada playstore di android, dan lain sebagainya.



2. Struktur seleksi (Selection)

Struktur seleksi merupakan instruksi atau pemilihan langkah yang dikerjakan berdasarkan kondisi tertentu (pengambilan keputusan). Kondisi adalah persyaratan yang dapat bernilai benar atau salah. Satu atau beberapa instruksi

hanya dilaksanakan apabila kondisi bernilai benar, sebaliknya apabila salah maka instruksi tidak akan dilaksanakan.



Terdapat beberapa struktur seleksi, yaitu:

A. #If - then

Instruksi hanya akan dikerjakan apabila kondisi bernilai benar. Dan sebaliknya, apabila kondisi bernilai salah, maka instruksi ini tidak akan dikerjakan.

Struktur Umum pada if-then yaitu :

```
If kondisi then instruksi
```

Struktur pemilihan if-then hanya melakukan satu instruksi ketika syarat dipenuhi atau bernilai benar, dan Ketika bernilai salah maka intruksi tidak dilakukan.

B. #If-then-else

Struktur pemilihan ini memberikan dua buah instruksi yang akan dikerjakan.

Struktur umum pada if-then-else yaitu :

```
If kondisi then  
instruksi 1  
Else  
instruksi 2
```

Else artinya “Jika tidak”. Bila kondisi benar, instruksi 1 akan dikerjakan. Bila kondisi salah (jika tidak) maka instruksi 2 yang akan dikerjakan.

Contoh:

```
If x < y then
Tulis x sebagai bilangan terkecil
Else
Tulis y sebagai bilangan terkecil
```

Contoh diatas untuk menentukan nilai yang lebih kecil dari dua bilangan bulat, x dan y (andaikan x tidak sama dengan y).

C. #If-then-else if

Kondisi ini terjadi ketika instruksi lebih dari satu. Terdapat tambahan pada struktur pilihannya, biasanya untuk pemilihan seperti ini disebut pemilihan bersarang.

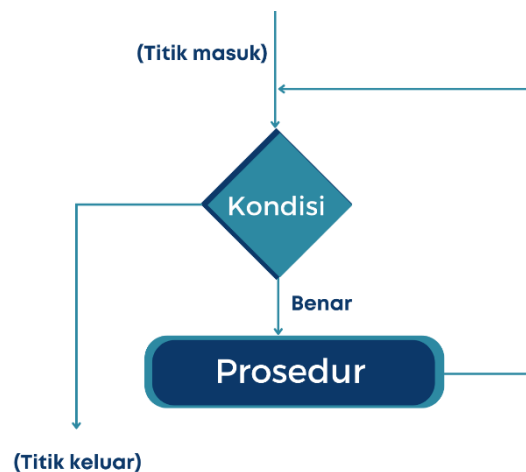
Contoh:

```
Menentukan bilangan terkecil dari tiga buah bilangan: x, y, z:
If x < y then
If x < z then
Tulis x sebagai bilangan terkecil
Else
Tulis z sebagai bilangan terkecil
Else If y < z then
Tulis y sebagai bilangan terkecil
Else
Tulis z sebagai bilangan terkecil
```

Struktur pemilihan memberikan pilihan untuk memiliki beberapa instruksi yang berbeda. Yang sesuai dengan kondisi atau syarat yang terpenuhi.

3. Struktur pengulangan (Repetition)

Struktur ini memberikan suatu perintah atau tindakan yang dilakukan beberapa kali. Dengan menggunakan struktur ini, pemrogram cukup menulis instruksi yang dapat dikerjakan beberapa kali. Pengulangan merupakan kegiatan mengerjakan sebuah atau sejumlah aksi yang sama sebanyak jumlah yang ditentukan atau sesuai dengan kondisi yang diinginkan.



A. #Pernyataan for

Pengulangan for digunakan apabila jumlah perulangan sudah diketahui terlebih dahulu. Bentuk umum pernyataan for adalah sebagai berikut:

```
For i = batas_awal to batas_akhir do
  Aksi
```

Aksi akan dilakukan sebanyak jumlah i, yaitu dari batas_awal sampai batas_akhir.

B. #Pernyataan repeat-until

Pengulangan dengan pernyataan repeat-until akan mengerjakan aksi sampai kondisinya terpenuhi. Bentuk umum pernyataan repeat until adalah sebagai berikut:

```
Repeat
  Aksi
```

```
Until (kondisi)
```

Untuk struktur repeat-until bisa digunakan apabila kondisi awal belum diketahui.

C. #Pernyataan while-do

Pernyataan while-do juga digunakan untuk pengulangan yang belum diketahui batasan jumlah pengulangannya. Pengulangan aksi akan terus dikerjakan apabila kondisi masih terpenuhi. Namun, apabila kondisi sudah tidak terpenuhi, maka pengulangan akan berakhir. Dengan kata lain, selama kondisi masih terpenuhi maka aksi akan terus dikerjakan. Bentuk umum pernyataan while-do adalah sebagai berikut:

```
While (kondisi) do  
Aksi
```

1.2 Penggunaan Algoritma di Kehidupan Sehari-hari

Pada dasarnya algoritma ini digunakan dalam menjalani kehidupan sehari-hari. Jika ketika ada orang memasak maupun merakit makanan pasti terdapat prosedur bagaimana langkah-langkah dalam melakukannya, hal ini lah secara tidak langsung juga menyatakan semacam algoritma. Prosedur ini merupakan urutan yang memandu orang untuk melakukan suatu proses.

Contoh 1. Algoritma Memasak nasi di magic com

```
Mulai  
Siapkan beras dan panci penanak nasi.  
Tuangkan beras ke dalam panci penanak nasi tersebut.  
Cuci beras dengan air hingga bersih.  
Setelah bersih, masukan air yang siap untuk memasak beras  
Masukan panci penanak nasi ke dalam magic com  
Tutup magic com  
Colokan kabel magic com ke kontak listrik  
Tekan tombol On agar magic com memasak beras  
Menunggu nasi hingga matang  
Nasi siap disajikan  
Selesai
```

Contoh 2. Algoritma Menghitung Luas Segitiga

Mulai
Masukan nilai alas (a)
Masukan nilai tinggi (t)
Hitung luas dengan rumus $Luas = \frac{1}{2} \times Alas \times Tinggi$
Nilai luas di tampilan
Selesai

Contoh 3. Algoritma mencetak dokumen dengan printer

Mulai
Buka dokumen yang akan dicetak
Nyalakan alat pencetak
Pilih menu Cetak atau Print (Ctrl+P)
Pilih alat pencetak yang akan digunakan
Tentukan halaman yang akan dicetak
Tentukan jumlah salinan yang akan dibuat
Pilih Cetak
Dokumen telah berhasil tercetak
Selesai

Contoh 4. Algoritma menentukan kelulusan mahasiswa

Mulai
Nama mahasiswa dan nilai (terbaca)
Kalau mahasiswa mendapat nilai >70 , ket "lulus"
Kalau mahasiswa mendapat nilai <70 , ket "tidak lulus"
Data nama, nilai, dan keterangan akan ditampilkan
Selesai

Contoh 5. Algoritma mencetak dokumen dengan printer

Mulai
Inputan besaran suhu dalam celcius (satuan celcius)
Konversi suhu dari celcius ke reamur dengan rumus $R = \frac{4}{5} \times C$
Melakukan konversi suhu dari celcius ke Fahrenheit dengan rumus $F = \frac{9}{5} \times C + 3$
Hasil dalam besaran Reamur dan Fahrenheit akan ditampilkan
Selesai

1.3 Bahasa Java dalam Pemrograman

1.3.1 Bahasa Pemrograman

Program merupakan implementasi algoritma yang ditulis dalam bahasa pemrograman tertentu sehingga dapat dilaksanakan oleh komputer. Program ditulis dalam salah satu bahasa pemrograman. Hingga saat ini terdapat cukup

banyak bahasa pemrograman antara lain, yaitu Bahasa C, C++, C#, Java, Python, R, Arduino, PHP, Algol, Pascal, Basic, Cobol, dan masih banyak lainnya. Berdasarkan tujuan aplikasinya, bahasa pemrograman dikelompokkan menjadi dua kategori, yaitu:

1. *General purpose programming language*

Pada kategori dapat digunakan untuk berbagai aplikasi dan bahasa yang biasa digunakan adalah bahasa pascal, basic, java, C, C++, dan sebagainya,

2. *Specific purpose programming language*

Pada kategori dapat digunakan untuk beberapa jenis aplikasi misalnya Cobol digunakan untuk terapan administrasi dan bisnis, assembly digunakan untuk aplikasi pemrograman mesin, prolog digunakan untuk aplikasi kecerdasan buatan.

1.3.2 Mengenal Bahasa Java Dalam Pemrograman

Bahasa java merupakan salah satu Bahasa pemrograman yang cukup dikenal, bahasa ini berorientasi pada objek. Pemrograman berorientasi objek ini merupakan bahasa pemrograman tingkat tinggi yang berorientasi objek dan program java tersusun dari bagian yang disebut *Class*, dimana *Class* terdiri atas metode-metode yang melakukan pekerjaan dan mengembalikan informasi setelah melakukan tugasnya. Beberapa keunggulan java yaitu java merupakan bahasa yang sederhana. Bahasa java ini dirancang agar mudah dipelajari dan digunakan secara efektif. Program java dapat ditulis dengan menggunakan beberapa teks editor seperti notepad++ dan jEdit.

Adapun fitur-fitur terpenting dari java yaitu, sebagai berikut:

1. *General-purpose computer-programming language*

Bahasa java dirancang sebagai bahasa pemrograman umum yang bisa dipakai untuk membuat berbagai aplikasi, mulai untuk aplikasi website, desktop, aplikasi mobile, maupun *embedded device* seperti microprocessor.

2. *Concurrent*

Concurrent programming merupakan fitur di sebuah bahasa pemrograman yang bisa menjalankan banyak hal dalam satu waktu. Lawan dari kata *concurrent* adalah *sequential programming*, yaitu dimana alur pemrograman di proses secara berurutan dari awal hingga akhir.

3. *Class-based, object-oriented*

Fitur paling terkenal di bahasa Java adalah pemrograman berbasis object. Berbeda dengan kebanyakan bahasa pemrograman yang mendukung pemrograman prosedural dan object sekaligus (seperti C++, PHP, atau Python), bahasa Java hanya menyediakan konsep object saja, memudahkan dalam pembuatan aplikasi yang kompleks

Didalam Bahasa java terdapat beberapa struktur dasar dalam program java yang secara umum dibagi menjadi 4 bagian, yaitu :

1. Deklarasi *Package*

Package merupakan sebuah folder yang berisi sekumpulan program Java. Package berguna dalam melakukan organisir sebuah file dalam suatu paket atau library. Adapun ketentuan dalam deklarasi *package* yaitu:

- Satu *package* merupakan satu subfolder di dalam file system
- Package sangat berguna untuk mengorganisir sebuah file dalam sebuah project atau library
- Nama package menggunakan lowercase
- Keyword package name

2. *Impor Library*

Import pada java merupakan suatu perintah untuk memasukan suatu method atau perintah dalam Bahasa Pemrograman Java sehingga perintah tersebut dapat Aktif dan digunakan atau berfungsi. Pada bagian ini, kita melakukan impor library yang dibutuhkan pada program. Library merupakan sekumpulan class dan fungsi yang bisa kita gunakan dalam membuat program.

3. Bagian *Class*

Setiap program harus dikemas di dalam *class* agar dapat dibuat menjadi sebuah objek. Dalam java terdapat dua bagian utama dari *class* yaitu deklarasi *class* dan *class body*. Deklarasi *class* berguna dalam mendeklarasikan *Class*, sedangkan *class body* berguna dalam mendeklarasikan variabel dan *method*.

4. Method *Main*

Method *main()* atau fungsi *main()* merupakan blok program yang akan dieksekusi pertama kali.

Sebuah contoh dari program java dapat ditunjukkan seperti contoh dibawah ini :

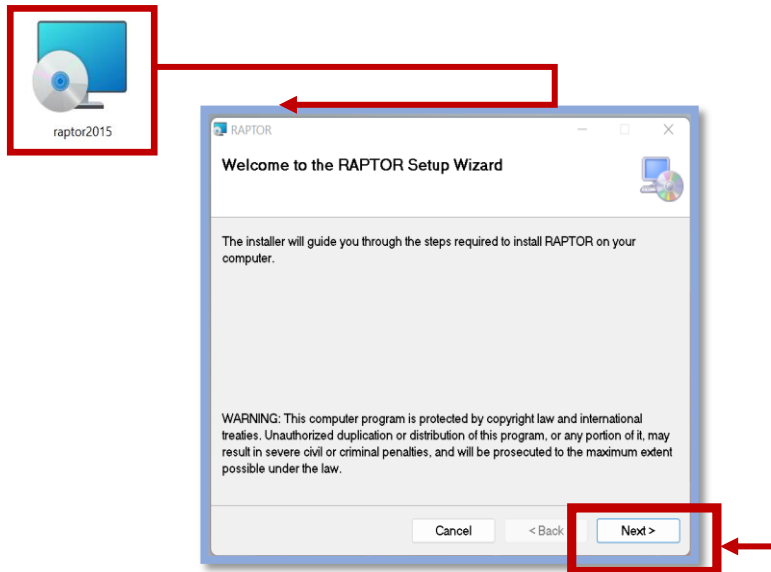
```
package hello.world.program; // → 1. deklarasi
package
import java.io.File; // → 2. Impor library
class Program { // → 3. Bagian class
    public static void main(String args[]){ // → 4.
Method main
        System.out.println("Hello World");
    }
}
```

1.4 Instalasi Aplikasi Raptor dan Visual Studio Code

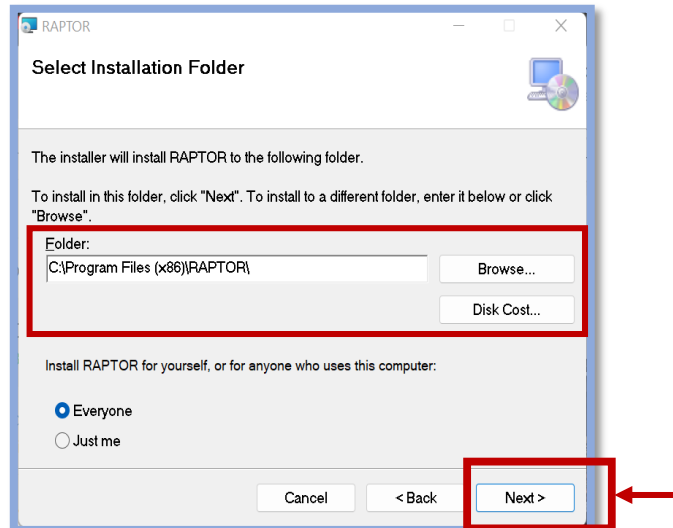
1.4.1 Instalasi Aplikasi Raptor

Raptor merupakan pemrograman yang berbasis flowchart yang dirancang khusus untuk membantu memvisualisasikan algoritma. Program Raptor dibuat secara visual dan dieksekusi secara visual dengan menelusuri eksekusi melalui flowchart. Adapun Langkah-langkah dalam instalasi aplikasi raptor ini yaitu sebagai berikut:

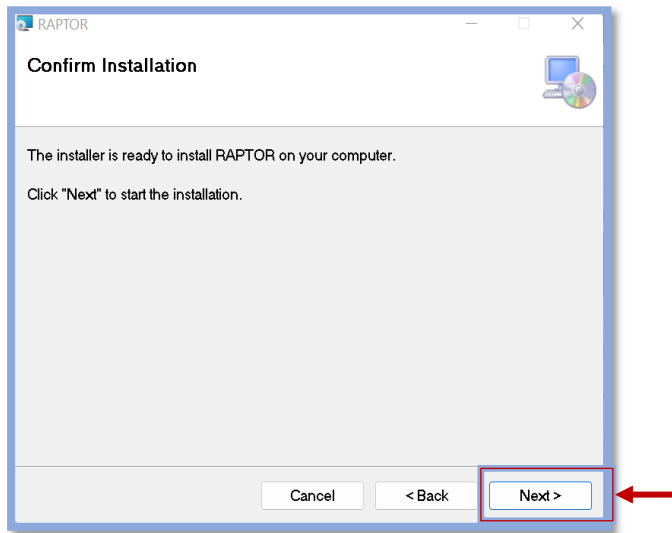
1. Unduh driver dari aplikasi raptor, buka hasil unduhan Raptor, kemudian pilih tombol "Next".



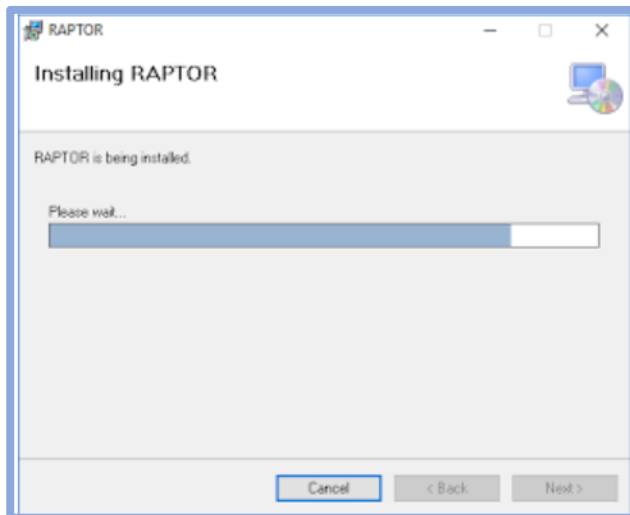
2. Tentukan tempat folder instalasi Raptor, kemudian pilih tombol "Next"



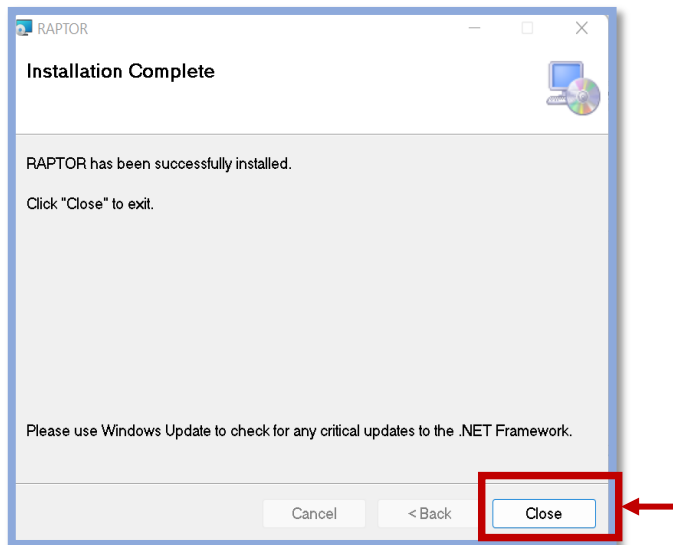
3. Kemudian pilih tombol "Next" lagi



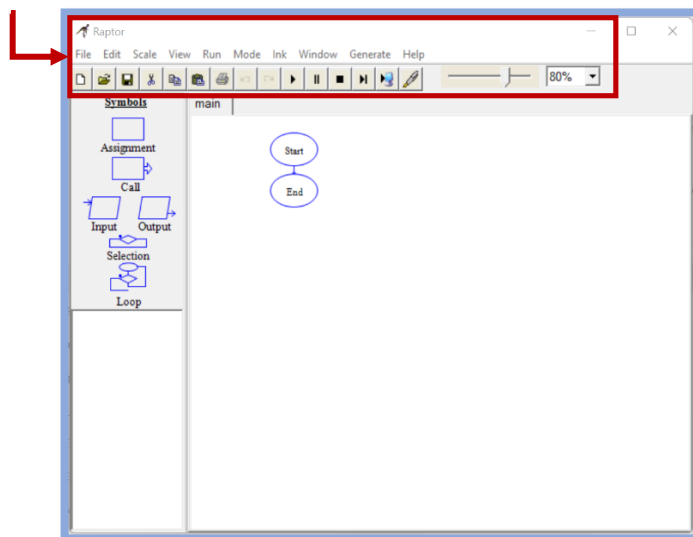
4. Tunggu proses instalasi selesai



5. Ketika instalasi berhasil maka akan ada tulisan "Installation Complete" selanjutnya aplikasi raptor bisa digunakan



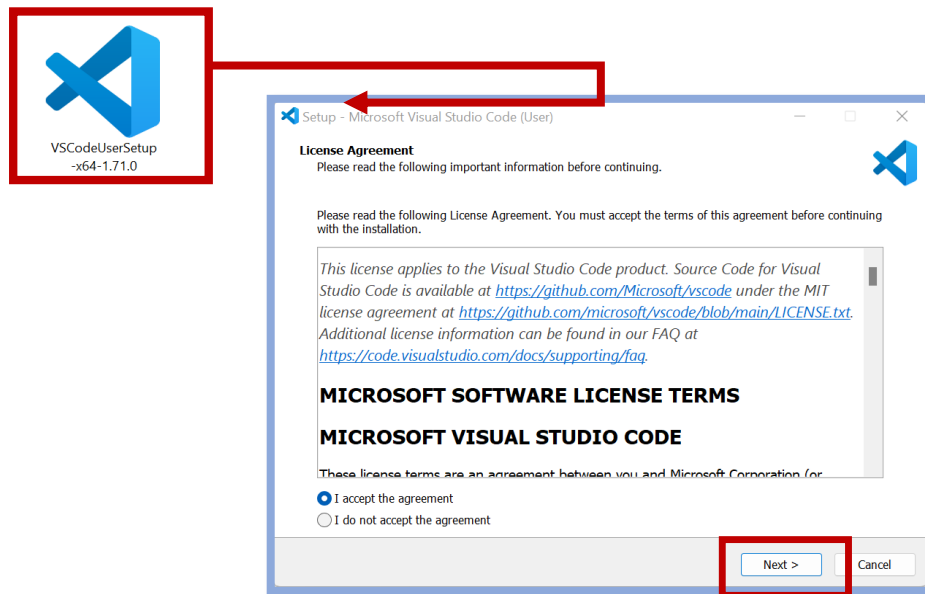
6. Ketika sudah terinstall maka aplikasi raptor bisa dibuka sesuai dengan gambar dibawah ini



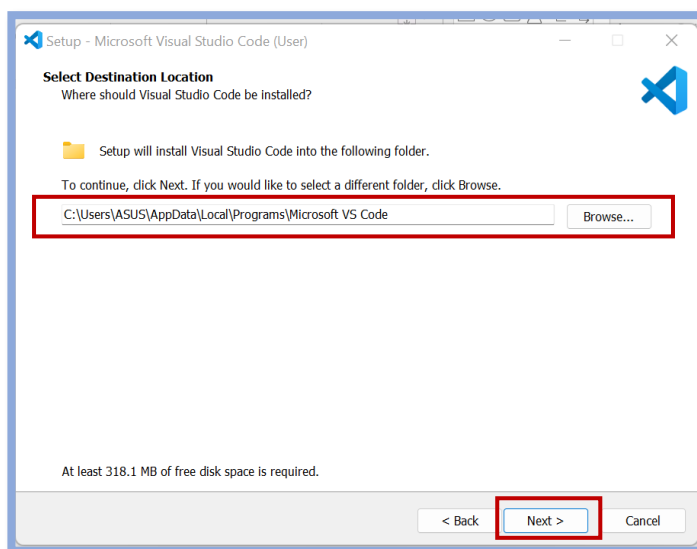
1.4.2 Instalasi Visual Studio Code

Visual Code Studio adalah sebuah code editor gratis yang bisa dijalankan di perangkat desktop berbasis Windows, Linux, dan MacOS. Code editor ini dikembangkan oleh salah satu raksasa teknologi dunia, Microsoft.

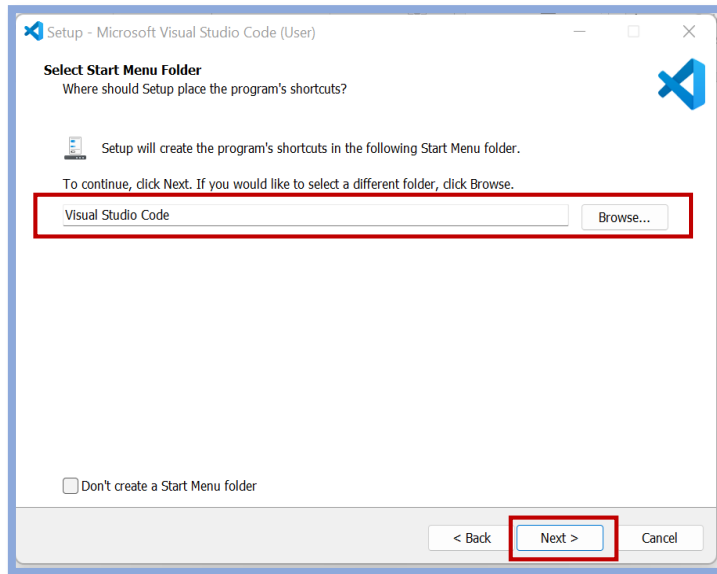
1. Unduh driver dari aplikasi visual studio code, buka hasil unduhan visual studio code, kemudian pilih tombol "Next"



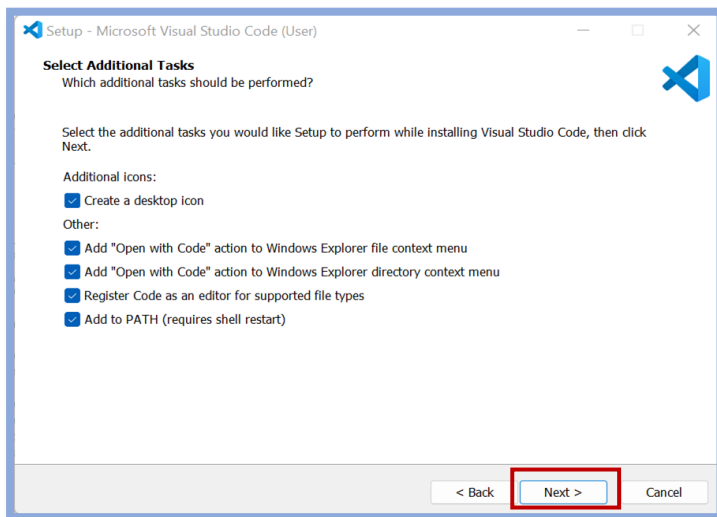
2. Tentukan tempat folder instalasi Visual studio code, kemudian pilih tombol "Next"



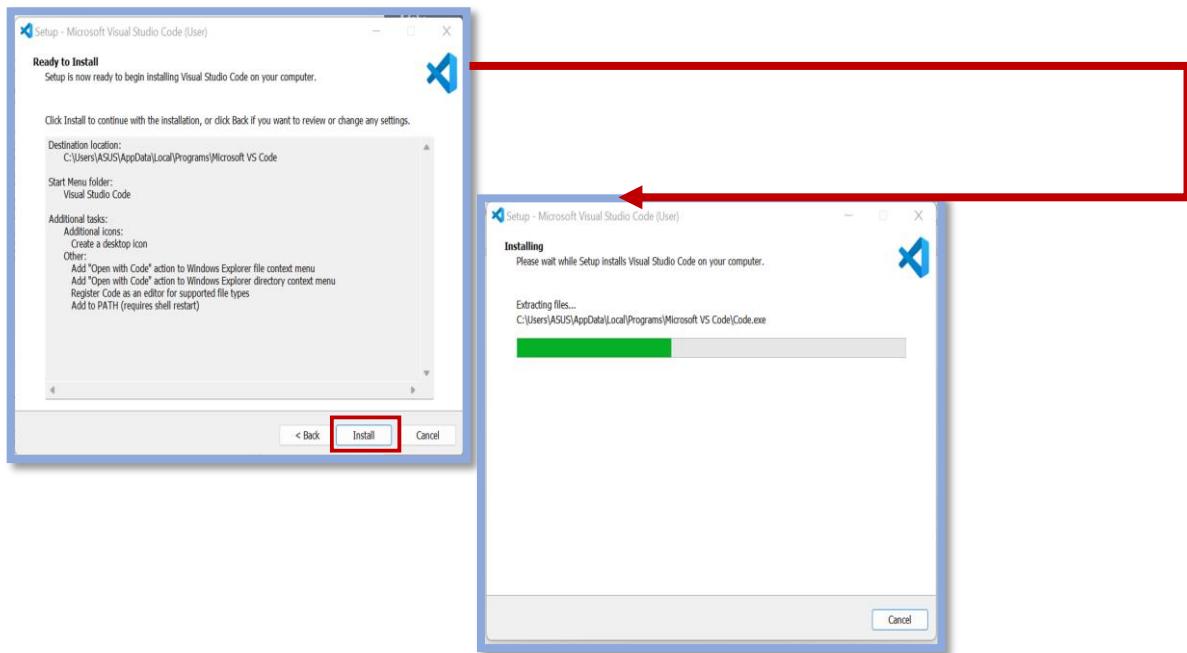
3. Setelah itu select start menu folder, lalu visual studio code dan selanjutnya klik "Next"



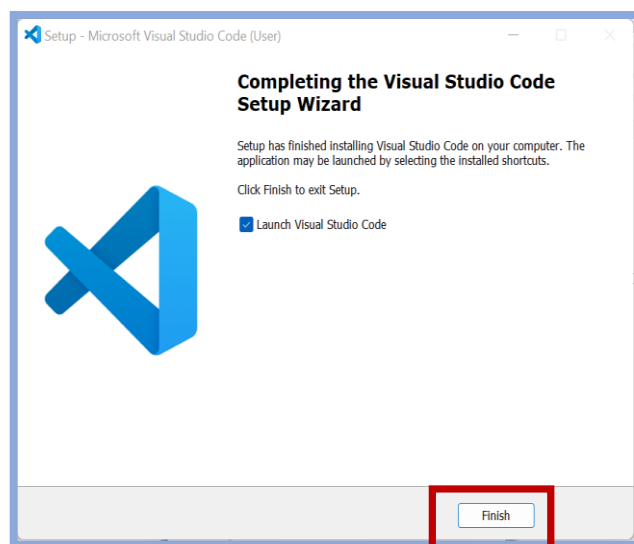
4. Pada additional tasks centang semua pilihan, selanjutnya klik "Next"



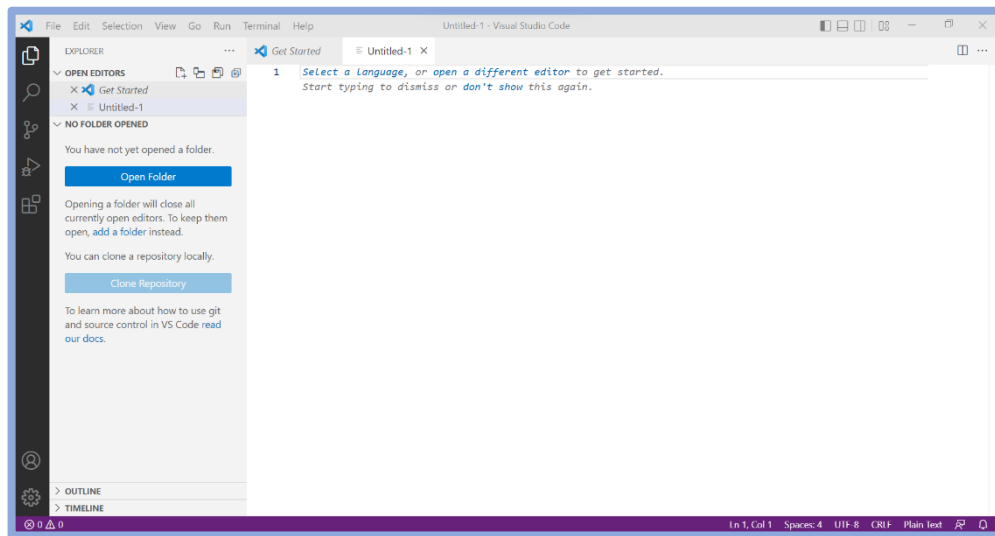
5. Ketika muncul halaman *Ready to Install*, maka bisa diklik pada tombol “Install” dan menunggu sampai instalasi selesai



6. Ketika proses instalasi selesai maka akan muncul halaman seperti dibawah ini, selanjutnya klik “Finish”



7. Visual studio code berhasil terinstal dan akan menampilkan halaman editor yang siap dipakai



POST TEST

Soal tentang Materi Pengantar Algoritma dan Pemrograman

1. Berikan contoh algoritma di kehidupan sehari hari dengan studi kasus dibawah ini:
 - a. Mengambil uang di ATM
 - b. Membuat kopi yang rasa manis dan pahitnya pas
 - c. Menghitung nilai y dari persamaan $y = 3x + 6$
2. Apa yang dimaksud dengan algoritma dan program?
3. Suatu algoritma terdiri dari tiga struktur dasar, yaitu sekuensial, pemilihan, dan perulangan. Jelaskan masing-masing!
4. Uraikan fungsi dari Bahasa pemrograman Java!
5. Uraikan mengenai komponen dari aplikasi Visual Studio Code!