

Pemilihan Tanaman pada Rotasi Tanam dengan Memanfaatkan Algoritma Program Dinamis

Dinda Yora Islami - 13516067

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

¹13516067@std.stei.itb.ac.id

Abstract—Melakukan rotasi dalam penanaman akan berdampak baik pada lahan yang ditanami. Penggunaan terus menerus pupuk kimia dan pestisida tanpa melakukan rotasi tanaman akan memperburuk kualitas tanah yang akan berdampak kekeringan tanah, erosi, resistensi hama dan masih banyak lagi. Namun melakukan rotasi tanaman tidak hanya sekedar rotasi tanaman. Perlu diperhatikan tanaman-tanaman yang akan dirotasi, salah melakukan rotasi tanaman malah akan menurunkan hasil panen. Oleh karena itu dibutuhkan strategi untuk melakukan rotasi tanaman. Salah satu penyelesaian masalah ini adalah dengan memodelkan permasalahan ini seperti mencari lintasan terpendek memanfaatkan pemrograman dinamis.

Keywords—rotasi tanaman, lintasan terpendek, optimasi, pemrograman dinamis.

I. PENDAHULUAN

Pangan merupakan kebutuhan mendasar bagi manusia untuk melangsungkan kehidupan. Pangan dapat diperoleh dari hewan dan tumbuhan. Untuk mencukupi kebutuhan pangan yang bersumber dari tumbuhan manusia melakukan budidaya tanaman. Selain bertujuan untuk mencukupi kebutuhan akan pangan budidaya ini juga bertujuan untuk keuntungan ekonomi berupa penjualan hasil budidaya.

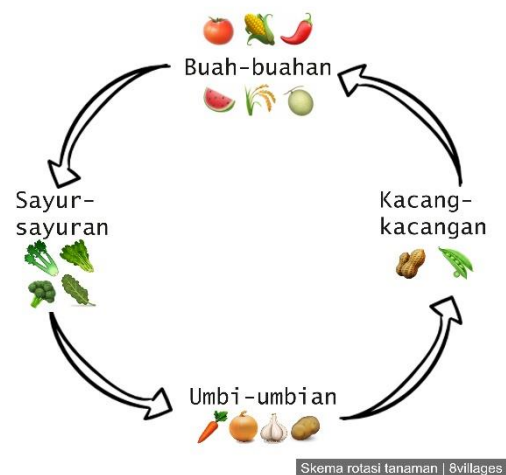
Untuk mendapatkan keuntungan yang besar, para pembudidaya atau yang disebut petani akan memilih tanaman yang memiliki nilai ekonomi tinggi. Namun terkadang petani hanya memprioritaskan nilai ekonomi dan mengabaikan kondisi lahan pertanian. Sehingga hanya melakukan penanaman dengan jenis atau keluarga yang sama secara terus-menerus. Melakukan penanaman seperti itu akan berdampak buruk terhadap lahan pertanian. Hal ini akan mengakibatkan tingkat kesuburan tanah menurun dan serangan organisme pengganggu meningkat. Hal ini dapat menyebabkan penurunan produktifitas lahan yang berakibat penurunan hasil panen.

Untuk menghindari penurunan hasil panen dapat diatasi dengan melakukan rotasi tanaman. Rotasi tanam atau gilir tanam adalah satu sistem budidaya tanaman dengan cara menggilir atau menanam tanaman berbeda dalam satu lahan tertentu. Pelaksanaan rotasi tanaman dapat memberi dampak positif berupa menjaga tekstur tanah sehingga meningkatkan kesuburan tanah, dan mengurangi intensitas serangan hama atau

penyakit.[2]

Pemilihan jenis tanaman dan urutan penanaman untuk dilakukan rotasi tanaman sangat penting. Jika terjadi kesalahan dalam pemilihan dan pengurutan tanam suatu tanaman maka akan berdampak produktivitas hasil tanaman yang mungkin akan lebih sedikit. Seperti tanaman kubis akan berproduksi rendah jika ditanam setelah tanaman kedelai/kacang hijau, namun akan berproduksi tinggi jika ditanam setelah jagung.

Pola tanam rotasi merupakan pola tanam yang ditentukan sesuai alur rotasi penanaman pada jenis tanaman tertentu agar tidak terjadi ketidakberhasilan panen (penurunan hasil panen). Pola tanam rotasi harus disesuaikan dengan pergantian jenis tanaman berikutnya dan harus tepat dalam memasangkan jenis tanaman satu dengan yang lainnya.



Gambar 1 skema rotasi tanaman [3]

Berdasarkan permasalahan-permasalahan dalam pemilihan tanaman pada rotasi tanam, untuk mendapatkan hasil panen yang optimal dibutuhkan pengaturan pola rotasi yang tepat. Salah satu cara untuk mengatur pola rotasi tanam adalah dengan menentukan jenis tanaman yang harus ditanam sebelum tanaman lain ditanam. Selain itu untuk mendapatkan hasil panen yang maksimal dilakukan pembatasan minimal hasil panen yang diinginkan untuk memilih tanaman dari jenis yang sama. Seperti pemilihan tanaman kacang-kacangan setelah tanaman umbi-umbian.

II. DASAR TEORI

A. Program Dinamis (*Dynamic Programming*)

Pemrograman dinamis (*Dynamic Programming*) adalah salah satu metode pemecahan masalah dengan cara menguraikan solusi menjadi sekumpulan tahapan (*stage*) sedemikian sehingga solusi dari persoalan dapat dipandang dari serangkaian keputusan yang saling berkaitan. Istilah program dinamis muncul karena perhitungan solusi dilakukan dengan menggunakan table-tabel. Pemrograman dinamis umumnya digunakan untuk menyelesaikan persoalan optimasi.

Pemecahan masalah dengan menggunakan metode pemrograman dinamis memiliki kemiripan dengan metode *greedy* yang juga membentuk solusi secara bertahap. Perbedaan kedua metode ini terletak pada rangkaian keputusan yang dipertimbangkan dalam penyelesaian masalah. Pada metode *greedy*, hanya satu rangkaian keputusan saja yang dipertimbangkan, yaitu rangkaian keputusan yang terdiri keputusan-keputusan yang memberikan hasil paling baik untuk setiap tahapnya (*optimal local*), yang harapannya dapat mengarah ke *optimal global*. Pada metode *greedy* keputusan yang dipilih pada tahap tertentu tidak mempertimbangkan rangkaian keputusan untuk tahapan berikutnya maupun keputusan yang telah dipilih pada tahap sebelumnya. Salah satu contoh permasalahan yang tidak dapat diselesaikan dengan metode *greedy* (tidak selalu benar) adalah *Integer (I/O) Knapsack Problem*.

Berbeda dengan metode *greedy*, metode pemrograman dinamis mempertimbangkan lebih dari satu rangkaian keputusan. Rangkaian-rangkaian keputusan tersebut dibuat dengan menggunakan prinsip optimalitas yang menyatakan bahwa jika solusi total optimal, maka bagian solusi sampai tahap ke- k juga optimal. Prinsip optimalitas berarti bahwa jika kita bekerja dari tahap k ke tahap $k + 1$, kita dapat menggunakan hasil optimal dari tahap k tanpa harus kembali ke tahap awal. Berdasarkan prinsip optimalitas ini dapat dirumuskan bahwa ongkos pada tahap $k + 1$ adalah (ongkos yang dihasilkan pada tahap k) + (ongkos dari tahap k ke tahap $k + 1$). Pemilihan keputusan pada suatu tahap pada metode pemrograman dinamis dilakukan dengan menggunakan persyaratan optimasi dan kendala untuk membatasi pilihan yang harus dipertimbangkan.

Pemrograman dinamis dapat diterapkan pada persoalan yang memiliki karakteristik sebagai berikut [1].

1. Persoalan dapat dibagi menjadi beberapa tahap (*stage*), yang pada setiap tahap hanya diambil satu keputusan.
2. Masing-masing tahap terdiri dari sejumlah status (*state*) yang berhubungan dengan tahap tersebut. Status merupakan kemungkinan masukan yang ada pada tahap tersebut.
3. Hasil dari keputusan yang diambil pada setiap tahap ditransformasikan dari status yang bersangkutan ke status berikutnya pada tahap berikutnya.
4. Ongkos (*cost*) pada suatu tahap meningkat secara teratur

(steadily) dengan bertambahnya jumlah tahapan.

5. Ongkos pada suatu tahap bergantung pada ongkos tahap-tahap yang sudah berjalan dan ongkos pada tahap tersebut.
6. Keputusan terbaik pada suatu tahap bersifat independen terhadap keputusan yang dilakukan pada tahap sebelumnya.
7. Adanya hubungan rekursif yang mengidentifikasi keputusan terbaik untuk setiap status pada tahap k memberikan keputusan terbaik untuk setiap status pada tahap $k + 1$.
8. Prinsip optimalitas berlaku pada persoalan tersebut.

Terdapat dua pendekatan yang dapat digunakan untuk menyelesaikan suatu persoalan dengan pemrograman dinamis yaitu sebagai berikut [1].

1. Pemrograman dinamis maju (*forward* atau *updown*). Pemrograman dinamis bergerak mulai dari tahap 1, lalu ke tahap 2, 3, dan seterusnya sampai tahap ke- n . Rangkaian keputusannya adalah $x_1, x_2, \dots, x_n$.
2. Pemrograman dinamis mundur (*backward* atau *bottom-up*). Pemrograman dinamis bergerak mulai tahap n , lalu mundur ke tahap $n - 1, n - 2$, dan seterusnya sampai tahap ke 1. Rangkaian keputusannya adalah $x_n, x_{n-1}, \dots, x_1$.

Kedua pendekatan ini ekuivalen dan sama-sama menghasilkan solusi yang optimum.

Pada umumnya untuk mengembangkan algoritma pemrograman dinamis terdapat empat langkah yaitu sebagai berikut [1].

1. Karakteristikan struktur solusi optimum.
2. Definisikan secara rekursif nilai solusi optimal.
3. Hitung nilai solusi optimal secara maju atau mundur.
4. Konstruksi solusi optimal.

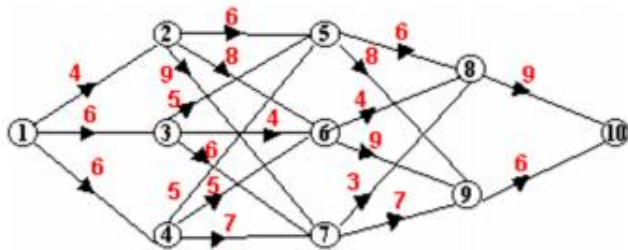
Pada implementasinya, pemrograman dinamis dapat diimplementasikan secara rekursif maupun secara iteratif dengan memanfaatkan larik (*array*) multi dimensi (tergantung banyaknya parameter yang merepresentasikan status untuk suatu tahap) untuk melakukan memorisasi. Implementasi secara iteratif akan membangun solusi dari bawah ke atas (*bottom-up*) dan menelusuri semua status yang mungkin untuk setiap tahap, sedangkan implemetasi secara rekursi akan membangun solusi dari atas ke bawah (*top-down*) membentuk sebuah pohon dan hanya akan mengunjungi status yang dibutuhkan pada setiap tahapnya untuk mencapai solusi optimal (tidak menelusuri semua status yang mungkin untuk setiap tahapnya).

III. PROGRAM DINAMIS DALAM MENENTUKAN LINTASAN TERPENDEK

Dalam menentukan jenis tanaman yang sesuai dengan rotasi tanam kita juga memperhitungkan keuntungan yang akan didapat. Keuntungan ini dapat kita lihat dari 2 faktor yaitu waktu penanaman dan harga tiap komoditas tanaman. Pada makalah ini

kita akan focus pada pemilihan jenis tanaman berdasarkan waktu yang dibutuhkan tanaman untuk tumbuh sampai dapat dipanen.

Untuk mendapatkan keuntungan yang cepat maka kita membutuhkan waktu yang cepat, sehingga pada kasus ini kita mencari waktu yang paling minimal. Kasus ini dapat kita modelkan seperti kasus pencarian lintasan terpendek. Contoh kasus lintasan terpendek dapat dilihat pada gambar 2. Kita mencari lintasan terpendek dari simpul 1 ke simpul 10.



Gambar 2 kasus lintasan terpendek[4]

Pada kasus pada gambar 2 kita dapat menyelesaikannya dengan memanfaatkan program dinamis maju. Kita misalkan x_1, x_2, x_3, x_4 adalah simpul-simpul yang dikunjungi pada tahap k ($k=1,2,3,4$). Maka rute yang dilalui adalah $x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4 \rightarrow 10$, dimana $x_1 = 1$. Pada persoalan ini tahap atau kita lambangkan dengan k adalah proses memilih simpul tujuan berikutnya (ada 4 tahap). Status atau kita lambangkan dengan s merupakan simpul-simpul dalam graf yang berhubungan dengan masing-masing tahap.

Relasi rekurs berikut menyatakan lintasan terpendek dari status s ke x_4 pada tahap k :

$$f_1(s) = c_{x_1s} \text{ (basis)}$$

$$f_k(s) = \min \{c_{x_ks} + f_{k-1}(x_k)\}, k = 2,3,4 \text{ (rekurens)}$$

Tujuan dari program dinamis maju yaitu mendapatkan $f_4(10)$ dengan cara mencari $f_1(s), f_2(s), f_3(s)$ terlebih dahulu.

Tahap 1:

$$f_1(s) = c_{x_1s}$$

Tabel I tahap 1 penyelesaian lintasan terpendek

S\X ₁	f ₁ (X ₁ S)			Solusi Optimum	
	1	f ₁ (s)	X ₁		
2	0+4=4	4	1		
3	0+6=6	6	1		
4	0+6=6	6	1		

Tahap 2 :

$$f_2(s) = \min \{c_{x_2s} + f_1(x_2)\}$$

Tabel II. Tahap 2 penyelesaian lintasan terpendek

S\X ₂	f ₂ (X ₂ S)			Solusi Optimum	
	2	3	4	f ₂ (s)	X ₂
5	4+6=10	6+5=11	6+5=11	10	2
6	4+8=12	6+4=10	6+5=11	10	3
7	4+9=13	6+6=12	6+7=13	12	4

Tahap 3:

$$f_3(s) = \min \{c_{x_3s} + f_2(x_3)\}$$

Tabel III. Tahap 3 penyelesaian lintasan terpendek

S\X ₃	f ₃ (X ₃ S)			Solusi Optimum	
	5	6	7	f ₃ (s)	X ₃
8	10+6=16	10+4=14	12+3=15	14	6
9	10+8=18	10+9=19	12+7=19	18	5

Tahap 4:

$$f_4(s) = \min \{c_{x_4s} + f_3(x_4)\}$$

Tabel IV tahap 4 penyelesaian lintasan terpendek

S\X ₄	f ₄ (X ₄ S)		Solusi Optimum	
	8	9	f ₄ (s)	X ₄
10	14+9=23	18+6=24	23	8

Tabel V. Hasil rute terpendek

	X ₄	X ₃	X ₂	X ₁	Panjang lintasan terpendek
10	← 8	← 6	← 3	← 1	23

Jadi lintasan terpendek dari 1 ke 10 yaitu $1 \rightarrow 3 \rightarrow 6 \rightarrow 8 \rightarrow 10$ yang mana panjang lintasan tersebut adalah 23.

Dari penanganan kasus lintasan terpendek pada gambar 2 kita mendapatkan lintasan terpendek atau lintasan yang optimal. Selanjutnya contoh permasalahan ini dapat kita jadikan panduan dalam memecahkan masalah dalam penentuan jenis tanaman untuk rotasi tanam sehingga menghasilkan hasil yang optimal dari segi waktu.

IV. PENERAPAN PROGRAM DINAMIS DALAM MENENTUKAN JENIS TANAMAN

Misalkan seorang petani akan melakukan penanaman tanaman untuk 2-5 tahun kedepan dengan list tanamannya seperti pada tabel VI beserta rincian jenis tanaman, waktu penanaman dan profit yang didapatkan per kilogramnya.

Tabel VI. Daftar Tanaman

No	Nama Tanaman	Jenis Tanaman	Waktu Tumbuh (hari) *	Profit (per kg)**
1.	Kacang Kedelai	Kacang-kacangan	90	13.000
2.	Kacang Hijau	Kacang-kacangan	70	19.000
3.	Buncis	Kacang-kacangan	60	4.500
4.	Sawi hijau	Sayuran	45	2.500
5.	Sawi Putih	Sayuran	40	5.000
6.	Kubis	Sayuran	85	4.300
7.	Brokoli	Sayuran	75	12.000
8.	Jagung	Biji-bijian	125	9.000
9.	Padi	Biji-bijian	110	12.000
10.	Kentang	Umbi	120	9.000
11.	Bawang	Umbi	90	15.000
12.	Wortel	Umbi	120	4.500
13.	Semangka	Buah	80	6.000
14.	Labu	Buah	60	3.500
15.	Tomat	Buah	70	5.000

*Penentuan waktu panen tidak mencerminkan waktu panen sebenarnya karena waktu panen dipengaruhi banyak factor

**Penentuan harga tidak mencerminkan harga sebenarnya karena adanya kenaikan atau penurunan harga

Dari daftar tersebut petani akan memilih serangkaian tanaman yang akan digilirkan di lahannya. Petani menginginkan dapat memperoleh hasil tanamannya dalam satu siklus rotasi dengan waktu yang seminimalnya. Persoalan ini dapat kita gambarkan seperti mencari lintasan terpendek dengan waktu tanam sebagai acuannya. Pada Persoalan ini diasumsikan petani sebelumnya sudah pernah melakukan bercocok tanam sehingga tanaman terakhir yang ditanam dijadikan sebagai patokan.

Persoalan ini akan diselesaikan dengan menggunakan program dengan metode pemrograman dinamis yang telah dijelaskan pada bagian II. Program dibuat dengan menggunakan algoritma yang telah dibahas pada bagian III dengan

menggunakan program dinamis maju (*forward*).

Pengguna akan memasukkan nama tanaman terakhir yang ditanam. Tanaman ini akan menjadi panduan dalam pemilihan tanaman berikutnya. Penggiliran tanaman dilihat dari jenis tanamannya. Siklus rotasi tanaman yang baik dapat kita lihat pada gambar 1. Kemudian pengguna memasukkan tanaman akhir dari siklus itu berupa tanaman yang sejenis dengan input pertama.

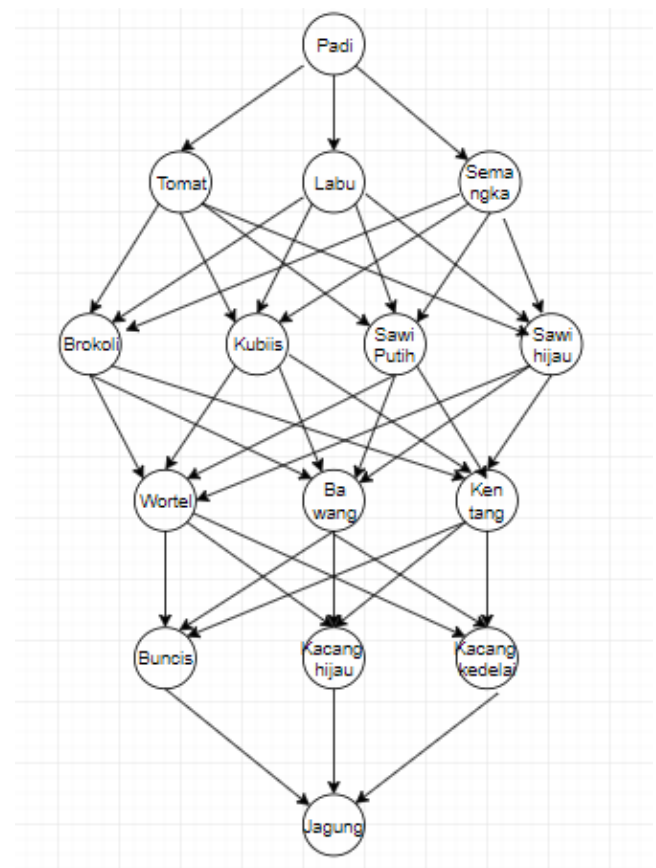
Berikut adalah eksperimen yang dilakukan oleh penulis dengan 2 kasus

1. Kasus 1

Input tanaman awal : Padi

Input tanaman akhir : Jagung

Dari inputan pengguna diperoleh graf berikut



Gambar 3 pemodelan graf kasus 1

Pada kasus ini terdapat 5 tahap yaitu $x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4 \rightarrow x_5 \rightarrow \text{Jagung}$, dimana $x_1 = \text{Padi}$

Tahap 1 :

S\X ₁	f ₁ (X ₁ S)	Solusi Optimum	
	Padi	f ₁ (s)	X ₁
Semangka	0+80=80	80	Padi

Labu	0+60=60	60	Padi
Tomat	0+70=70	70	Padi

Tahap 2:

S\X ₂	f ₂ (X ₂ S)			Solusi Optimum	
	Semangka	Labu	Tomat	f ₂ (s)	X ₂
Brokoli	80+75=155	60+75=135	70+75=145	135	Labu
Kubis	80+85=165	60+85=145	70+85=155	145	Labu
Sawi Putih	80+40=120	60+40=100	70+40=110	100	Labu
Sawi Hijau	80+45=125	60+45=105	70+45=115	105	Labu

Tahap 3

S\X ₃	f ₃ (X ₃ S)				Solusi Optimum	
	Brokoli	Kubis	Sawi P	Sawi H	f ₃ (s)	X ₃
Wortel	135 + 120 = 255	145 + 120 = 265	100 + 120 = 220	105 + 120 = 225	220	Sawi P
Bawang	135 + 90 = 225	145 + 90 = 235	100 + 90 = 190	105 + 90 = 195	190	Sawi P
Kentang	135 + 120 = 255	145 + 120 = 265	100 + 120 = 220	105 + 120 = 225	220	Sawi P

Tahap 4 :

S\X ₄	f ₄ (X ₄ S)			Solusi Optimum	
	Wortel	Bawang	Kentang	f ₄ (s)	X ₄
Buncis	220+60=280	190+60=250	220+60=280	250	Bawang
Kacang Hijau	220+70=290	190+70=260	220+70=290	260	Bawang
Kacang Kedelai	220+90=310	190+90=280	220+90=310	280	Bawang

Tahap 5

S\X ₅	f ₅ (X ₅ S)			Solusi Optimum	
	Buncis	Kacang Hijau	Kacang Kedelai	f ₅ (s)	X ₅
Jagung	250+125=375	260+125=385	280+125=405	375	Buncis

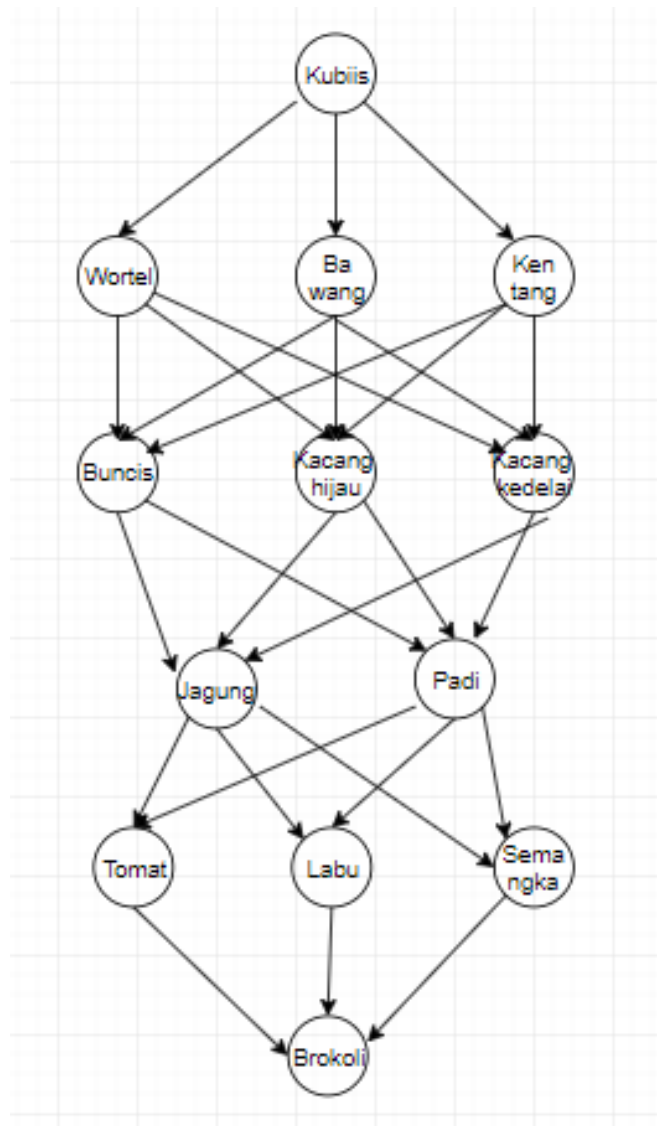
Berdasarkan penyelesaian kasus 1 dari tahap 1 sampai 5 disimpulkan bahwa jenis tanaman yang di lanjutkan adalah

Padi→Labu→Sawi Putih→Bawang→Buncis→Jagung
 Dengan waktu satu siklus selama 375 hari.
 2. Kasus 2

Input tanaman awal : Kubis

Input tanaman akhir : Brokoli

Dari inputan pengguna diperoleh graf berikut



Gambar 4 pemodelan graf kasus 2

Tahap 1 :

S\X ₁	f ₁ (X ₁ S)	Solusi Optimum	
	Kubis	f ₁ (s)	X ₁
Wortel	0+120=120	120	Kubis
Bawang	0+90=90	90	Kubis
Kentang	0+120=120	120	Kubis

Tahap 2:

$S \setminus X_2$	$f_2(X_2S)$			Solusi Optimum	
	Wortel	Bawang	Kentan	$f_2(s)$	X_2
Buncis	120+60 =180	90+60 =150	120+60 =180	150	Bawang
Kacang Hijau	120+70 =190	90+70 =160	120+70 =190	160	Bawang
Kacang Kedelai	120+90 =210	90+90 =180	120+90 =210	180	Bawang

Tahap 3

$S \setminus X_3$	$f_3(X_3S)$			Solusi Optimum	
	Buncis	Kacang Hijau	Kacang Kedelai	$f_3(s)$	X_3
Jagung	150+125 = 275	160+125 =285	180+125 =305	275	Buncis
Padi	150+110 =260	160+110 =270	180+110 =290	260	Buncis

Tahap 4 :

$S \setminus X_4$	$f_4(X_4S)$		Solusi Optimum	
	Jagung	Padi	$f_4(s)$	X_4
Semangka	275+80 =355	260+80 = 340	340	Padi
Labu	275+60 =335	260+60 = 320	320	Padi
Tomat	275+70 =345	260+70 = 330	330	Padi

Tahap 5

$S \setminus X_5$	$f_5(X_5S)$			Solusi Optimum	
	Semangka	Labu	Tomat	$f_5(s)$	X_5
Brokoli	340+75 = 415	320+75 = 395	330+75 =405	395	Labu

Berdasarkan penyelesaian kasus 2 dari tahap 1 sampai 5 disimpulkan bahwa jenis tanaman yang dianjurkan adalah Kubis→Bawang→Buncis→Padi→Labu→Brokoli

Dengan waktu satu siklus selama 395 hari.

V. KESIMPULAN

Algoritma pemrograman dinamis dapat digunakan untuk menyelesaikan permasalahan optimasi dikarenakan terdapat prinsip optimalitas. Salah satu aplikasi dari program dinamis adalah mencari lintasan terpendek dari suatu lintasan.

Persoalan lintasan terpendek yang diselesaikan dengan pemrograman dinamis dapat digunakan untuk memodelkan berbagai persoalan yang berhubungan dengan pemilihan objek-objek tertentu dengan tujuan optimalisasi. Salah satunya adalah untuk memilih tanaman yang sesuai rotasi tanamnya namun memberikan waktu yang optimal dalam proses penanamannya. Dari hasil percobaan dapat dilihat bahwa hasil yang didapatkan merupakan hasil yang optimal, namun hasil tersebut belum bisa digunakan langsung di dunia nyata. Hal ini disebabkan banyaknya faktor-faktor lain yang berpengaruh seperti ketersediaan air, iklim atau cuaca, dan jenis tanaman itu sendiri.

V. UCAPAN TERIMA KASIH

Pada bagian ini, penulis ingin mengucapkan terima kasih kepada Allah SWT atas segala rahmat dan berkahnya sehingga penulis dapat menyelesaikan makalah ini. Ucapan terima kasih juga penulis sampaikan kepada orangtua penulis, yang selalu mendukung dan mendoakan penulis tanpa henti. Kemudian, terima kasih juga penulis ucapkan kepada ibu Masayu Leylia Khodra, S.T., M.T. selaku dosen mata kuliah IF2211 Strategi Algoritma, atas bimbingan beliau selama ini dalam mengajar dan memberikan ilmu sehingga penulis mampu membuat makalah ini. Terakhir, terima kasih kepada teman-teman dan sahabat atas dukungannya selama ini.

REFERENSI

- [1] Munir, Rinaldi. Diktat kuliah IF2211 Strategi Algoritma. Program Studi Teknik Informatika. Institut Teknologi Bandung. 2009
- [2] <https://lmgagro.wordpress.com/2015/06/16/pentingnya-rotasi-tanaman-untuk-kesuburan-tanah/> [diakses pada 13 Mei 2018 pukul 16.43]
- [3] <http://8villages.com/full/petani/article/id/59db479a536469d27e7b881b> [diakses pada 13 Mei 2018 di akses pukul 16.30]
- [4] <http://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2017-2018/PR-DP.pdf> [diakses pada 13 Mei 2018 pukul 23.00]

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 13 Mei 2018



Dinda Yora Islami - 13516067