Analisis Performansi Algoritma Knapsack

Untuk Optimalisasi Pemilihan Proyek di PT. GITS Indonesia



Bagja Gumelar
10108070

Informatika Teoritik

Latar Belakang

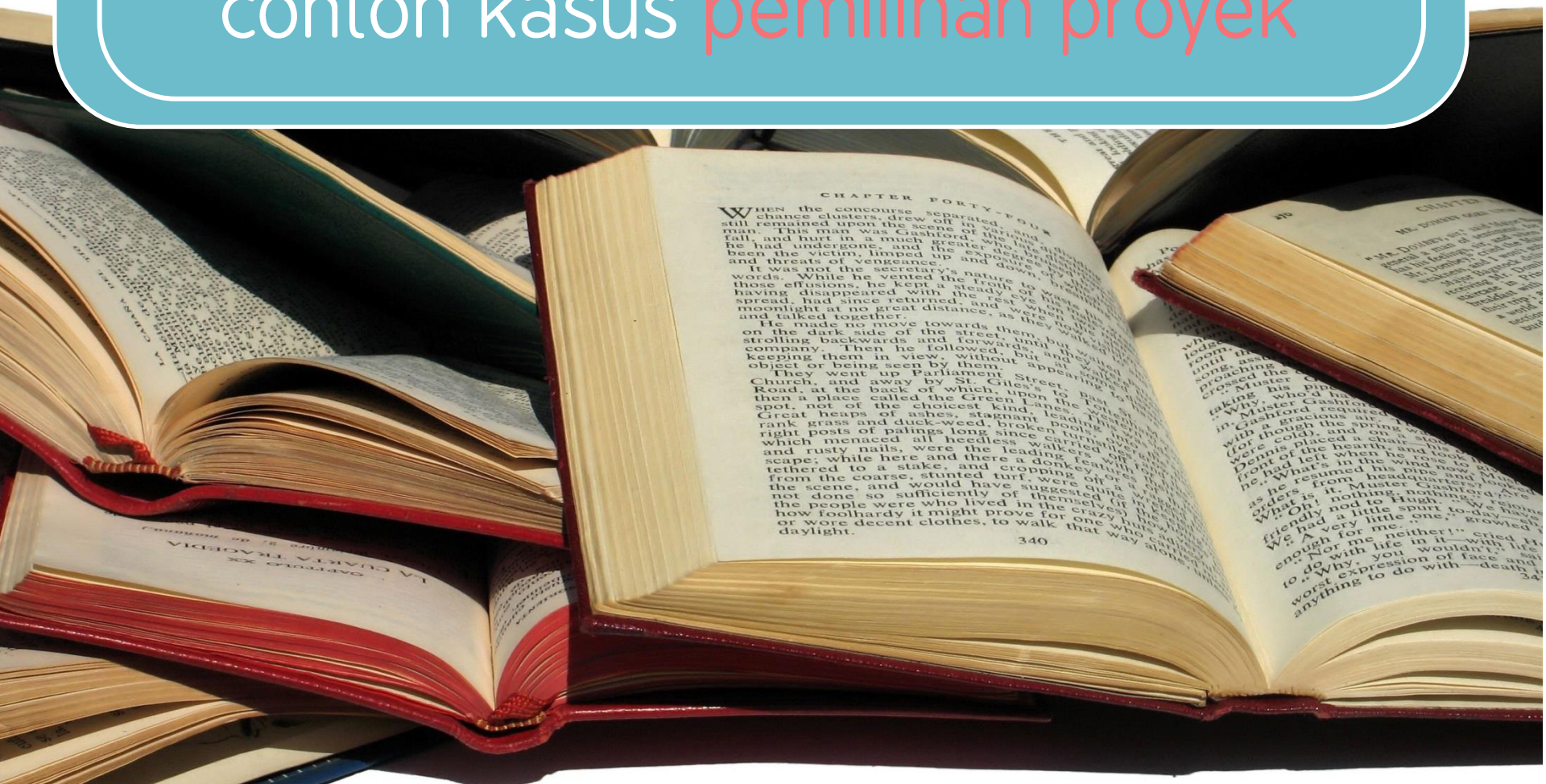
Knapsack diteliti sejak akhir tahun 50-an, untuk diterapkan dalam sektor industri dan manajemen keuangan, tetapi lebih sering dilakukan untuk penelitian yang bersifat teoritis

David Pisinger,1995: 15

Contoh umum ilustrasi algoritma knapsack antara lain : optimalisasi perencanaan investasi, muat barang kargo, kontrol budget keuangan dan manajemen keuangan

David Pisinger,1995: 15

Dari sembilan literatur tentang algoritma knapsack belum ditemukan literatur yang meneliti contoh kasus pemilihan proyek



Kesamaan karateristik dengan pemilihan proyek di PT. GITS Indonesia melibatkan pertimbangan jumlah Sumber Daya Manusia dan Nilai Keuntungan proyek

Perumusan Masalah



Bagaimana cara menganalisis performansi algoritma *Knapsack*

Tujuan Penelitian



Menganalisis performansi algoritma *knapsack* pada kasus pemilihan proyek di PT. GITS Indonesia

Mendapatkan hasil yang optimal dari kasus pemilihan proyek menggunakan strategi *greedy* dengan algoritma *knapsack*

Batasan Masalah

Algoritma yang digunakan adalah Algoritma *Fractional Knapsack dengan pendekatan Greedy By Density*

Model analisis yang digunakan adalah model kompleksitas waktu (Tn)



Notasi asimtotik yang digunakan adalah O (big oh)

Pendukung simulasi algoritma knapsack menggunakan perangkat lunak berbasis desktop


Parameter
Knapsack

Knapsack	Parameter Pemilihan Proyek	Parameter
Weight	Banyak Sumber Daya Manusia yang dibutuhkan untuk mengerjakan sebuah proyek	wi
Profit	Keuntungan tiap proyek	pi
Constraint	Ketersediaan SDM (Sumber Daya Manusia) di perusahaan	W

ANALISIS STRATEGI




Greedy = Rakus

“take

what you

can get

now!”



Greedy membentuk solusi langkah per langkah

Jenis-Jenis Pendekatan Strategi Greedy



Greedy By Weight (Wi)



Greedy By Profit (Pi)





Greedy By Density (Pi/Wi)

Percobaan Perhitungan Strategi *Greedy*

Jumlah SDM yang tersedia *W* = 10

Properti Proyek				Greedy by		
Proyek ke-	*SDM*	*keuntungan*	*density*	*Weight*	*profit*	*density*
1	2	50000	25000	0	0	0
2	3	51000	17000	0	0	0
3	2	45000	22500	0	0	0
4	4	123000	30750	0	0	0
5	2	40000	20000	0	0	0
6	2	56000	28000	0	0	0
Total SDM Terpakai				0	0	0
Total keuntungan				0	0	0

BY WEIGHT

Jumlah SDM yang tersedia $W = 10$

Properti Proyek				Greedy by		
Proyek ke-	SDM	keuntungan	density	Weight	profit	density
1	2	50000	25000	1	0	0
2	3	51000	17000	0	0	0
3	2	45000	22500	0	0	0
4	4	123000	30750	0	0	0
5	2	40000	20000	0	0	0
6	2	56000	28000	0	0	0
Total SDM Terpakai				2	0	0
Total keuntungan				50000	0	0

BY WEIGHT

Jumlah SDM yang tersedia $W = 10$

Properti Proyek				Greedy by		
Proyek ke-	*SDM*	*keuntungan*	*density*	*Weight*	*profit*	*density*
1	2	50000	25000	1	0	0
2	3	51000	17000	0	0	0
3	2	45000	22500	1	0	0
4	4	123000	30750	0	0	0
5	2	40000	20000	0	0	0
6	2	56000	28000	0	0	0
Total SDM Terpakai				4	0	0
Total keuntungan				95000	0	0

BY WEIGHT

Jumlah SDM yang tersedia $W = 10$

Properti Proyek				Greedy by		
Proyek ke-	*SDM*	*keuntungan*	*density*	*Weight*	*profit*	*density*
1	2	50000	25000	1	0	0
2	3	51000	17000	0	0	0
3	2	45000	22500	1	0	0
4	4	123000	30750	0	0	0
5	2	40000	20000	1	0	0
6	2	56000	28000	0	0	0
Total SDM Terpakai				6	0	0
Total keuntungan				135000	0	0

BY WEIGHT

Jumlah SDM yang tersedia $W = 10$

Properti Proyek				Greedy by		
Proyek ke-	SDM	keuntungan	density	Weight	profit	density
1	2	50000	25000	1	0	0
2	3	51000	17000	0	0	0
3	2	45000	22500	1	0	0
4	4	123000	30750	0	0	0
5	2	40000	20000	1	0	0
6	2	56000	28000	1	0	0
Total SDM Terpakai				8	0	0
Total keuntungan				191000	0	0

BY WEIGHT

Jumlah SDM yang tersedia $W = 10$

Properti Proyek				Greedy by		
Proyek ke-	*SDM*	*keuntungan*	*density*	*Weight*	*profit*	*density*
1	2	50000	25000	1	0	0
2	3	51000	17000	2/3 or 66%	0	0
3	2	45000	22500	1	0	0
4	4	123000	30750	0	0	0
5	2	40000	20000	1	0	0
6	2	56000	28000	1	0	0
Total SDM Terpakai				10	0	0
Total keuntungan				225000	0	0

Jumlah SDM yang tersedia *W* = 10

Properti Proyek				*Greedy by*		
Proyek ke-	*SDM*	*keuntungan*	*density*	*Weight*	*profit*	*density*
1	2	50000	25000	1	0	0
2	3	51000	17000	2/3 or 66%	0	0
3	2	45000	22500	1	0	0
4	4	123000	30750	0	0	0
5	2	40000	20000	1	0	0
6	2	56000	28000	1	0	0
Total SDM Terpakai				10	0	0
Total keuntungan				225000	0	0

BY PROFIT

Jumlah SDM yang tersedia $W = 10$

Properti Proyek				Greedy by		
Proyek ke-	*SDM*	*keuntungan*	*density*	*Weight*	*profit*	*density*
1	2	50000	25000	1	0	0
2	3	51000	17000	2/3 or 66%	0	0
3	2	45000	22500	1	0	0
4	4	123000	30750	0	1	0
5	2	40000	20000	1	0	0
6	2	56000	28000	1	0	0
Total SDM Terpakai				10	4	0
Total keuntungan				225000	123000	0

BY PROFIT

Jumlah SDM yang tersedia $W = 10$

Properti Proyek				Greedy by		
Proyek ke-	*SDM*	*keuntungan*	*density*	*Weight*	*profit*	*density*
1	2	50000	25000	1	0	0
2	3	51000	17000	2/3 or 66%	0	0
3	2	45000	22500	1	0	0
4	4	123000	30750	0	1	0
5	2	40000	20000	1	0	0
6	2	56000	28000	1	1	0
Total SDM Terpakai				10	6	0
Total keuntungan				225000	179000	0

BY PROFIT

Jumlah SDM yang tersedia *W* = 10

Properti Proyek				Greedy by		
Proyek ke-	*SDM*	*keuntungan*	*density*	*Weight*	*profit*	*density*
1	2	50000	25000	1	0	0
2	3	51000	17000	2/3 or 66%	1	0
3	2	45000	22500	1	0	0
4	4	123000	30750	0	1	0
5	2	40000	20000	1	0	0
6	2	56000	28000	1	1	0
Total SDM Terpakai				10	9	0
Total keuntungan				225000	230000	0

BY PROFIT

Jumlah SDM yang tersedia *W* = 10

Properti Proyek				Greedy by		
Proyek ke-	*SDM*	*keuntungan*	*density*	*Weight*	*profit*	*density*
1	2	50000	25000	1	½ or 50%	0
2	3	51000	17000	2/3 or 66%	1	0
3	2	45000	22500	1	0	0
4	4	123000	30750	0	1	0
5	2	40000	20000	1	0	0
6	2	56000	28000	1	1	0
Total SDM Terpakai				10	10	0
Total keuntungan				225000	255000	0

Ulangi hal yang sama untuk Greedy By Density!

CONCLUSION

Jumlah SDM yang tersedia $W = 10$

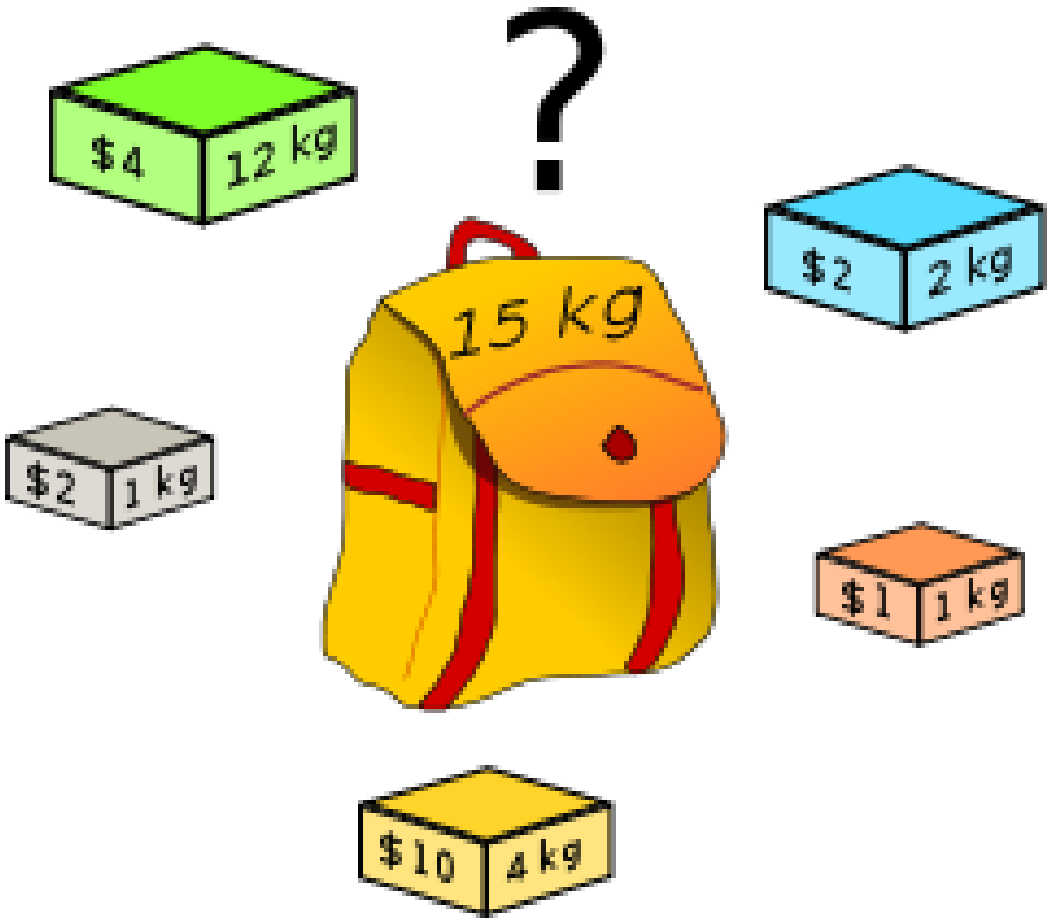
Properti Proyek				Greedy by		
Proyek ke-	*SDM*	*keuntungan*	*density*	*Weight*	*profit*	*density*
1	2	50000	25000	1	1/2 or 50%	1
2	3	51000	17000	2/3 or 66%	1	0
3	2	45000	22500	1	0	1
4	4	123000	30750	0	1	1
5	2	40000	20000	1	0	0
6	2	56000	28000	1	1	1
Total SDM Terpakai				10	10	10
Total keuntungan				225000	255000	274000

CONCLUSION

Jumlah SDM yang tersedia $W = 10$

Properti Proyek				Greedy by		
Proyek ke-	SDM	keuntungan	density	Weight	profit	density
1	2	50000	25000	1	1/2 or 50%	1
2	3	51000	17000	2/3 or 66%	1	0
3	2	45000	22500	1	0	1
4	4	123000	30750	0	1	1
5	2	40000	20000	1	0	0
6	2	56000	28000	1	1	1
Total SDM Terpakai				10	10	10
Total keuntungan				225000	255000	274000

ANALISIS ALGORITMA KNAPSACK



Knapsack = Kantong

Pemilihan barang-barang yang akan dibawa kedalam kantong

Jenis-Jenis Algoritma Knapsack

0/1 Knapsack

Fractional Knapsack

Kelebihan Algoritma Fractional Knapsack

Selalu memberikan solusi yang optimal

Rinaldi Munir,2004: 23

Dalam kasus pemilihan proyek, Fractional knapsack dapat memberikan pertimbangan untuk sisa constraint yang belum terisi dengan memberikan pertimbangan berupa pecahan atau persen (%) sehingga profit dapat lebih dioptimalkan

BIG-O Algoritma Fractional Knapsack

```
read (SDM[i], profit[i], k)
density[i]←profit[i]/SDM[i]
PengurutanNilaiBerdasarkanDensity(density[i])
Opt_sol←0
i←1
for i←1 to n do
   optimum[i] ← 0
endfor
for i ← to n do
  if SDM[i] > k then
    eof
  else
    optimum[i] ← 1
     k←k-SDM[i]
  endif
  if i<= n then optimum[i] ← k/SDM[i]
    else opt_sol←0
    for i←0 to n do
       opt_sol←opt_sol+(profit[i]*optimum[i])
    endfor
  endif
endfor
write (SDM[i], profit[i], density[i], optimum[i], opt_sol)
```

```
read (SDM[i], profit[i], k)
density[i]←profit[i]/SDM[i]
PengurutanNilaiBerdasarkanDensity(density[i])
Opt_sol←0
i←1
for i←1 to n do
   optimum[i] ← 0
endfor
for i ← to n do
  if SDM[i] > k then
    eof
  else
    optimum[i] ← 1
     k←k-SDM[i]
  endif
  if i<= n then optimum[i] ← k/SDM[i]
    else opt_sol←0
    for i←0 to n do
       opt_sol←opt_sol+(profit[i]*optimum[i])
    endfor
  endif
endfor
write (SDM[i], profit[i], density[i], optimum[i], opt_sol)
```

```
read (SDM[i], profit[i], k)
density[i]←profit[i]/SDM[i]
PengurutanNilaiBerdasarkanDensity(density[i])
Opt_sol←0
i←1
for i←1 to n do
   optimum[i] ← 0
endfor
for i ← to n do
  if SDM[i] > k then
    eof
  else
    optimum[i] ← 1
     k←k-SDM[i]
  endif
  if i<= n then optimum[i] ← k/SDM[i]
    else opt_sol←0
    for i←0 to n do
       opt_sol←opt_sol+(profit[i]*optimum[i])
    endfor
  endif
endfor
write (SDM[i], profit[i], density[i], optimum[i], opt_sol)
```



$O(n)$

Kompleksitas waktu asimptotik untuk algoritma fractional knapsack adalah $O(n)$

Contoh Kasus Algoritma Fractional Knapsack

No	Nama Project	SDM yang dibutuhkan	Profit (By Ratio)
1	Mindtalk for Android	2	15300
2	Mindtalk for Samsung	2	23000
3	H!Bali	2	10500
4	Mashup Intelligent Car	4	26950
5	M2M Public Utility	2	24650
6	RCS-e Client Android TV Box	2	12425

FRACTIONAL KNAPSACK

Jumlah SDM yang tersedia $W = 9$

Properti Proyek				Greedy by density
Proyek Ke-	w_i	p_i	p_i / w_i	
1	2	15300	7650	0
2	2	23000	11500	0
3	2	10500	5250	0
4	4	26950	6737,5	0
5	2	24650	12325	0
6	2	12425	6212,5	0
Total keuntungan				0
Total SDM yang terpakai				0
Jumlah SDM yang tersedia				9

FRACTIONAL KNAPSACK

Jumlah SDM yang tersedia $W = 9$

Properti Proyek				Greedy by density
Proyek Ke-	w_i	p_i	p_i / w_i	
1	2	15300	7650	0
2	2	23000	11500	0
3	2	10500	5250	0
4	4	26950	6737,5	0
5	2	24650	12325	0
6	2	12425	6212,5	0
Total keuntungan				0
Total SDM yang terpakai				0
Jumlah SDM yang tersedia				9

FRACTIONAL KNAPSACK

Jumlah SDM yang tersedia $W = 9$

Properti Proyek				Greedy by density
Proyek Ke-	w_i	p_i	p_i / w_i	
1	2	15300	7650	0
2	2	23000	11500	0
3	2	10500	5250	0
4	4	26950	6737,5	0
5	2	24650	12325	1
6	2	12425	6212,5	0
Total keuntungan				24650
Total SDM yang terpakai				2
Jumlah SDM yang tersedia				9

FRACTIONAL KNAPSACK

Jumlah SDM yang tersedia $W = 9$

Properti Proyek				Greedy by density
Proyek Ke-	w_i	p_i	p_i / w_i	
1	2	15300	7650	0
2	2	23000	11500	0
3	2	10500	5250	0
4	4	26950	6737,5	0
5	2	24650	12325	1
6	2	12425	6212,5	0
Total keuntungan				24650
Total SDM yang terpakai				2
Jumlah SDM yang tersedia				9

FRACTIONAL KNAPSACK

Jumlah SDM yang tersedia $W = 9$

Properti Proyek				Greedy by density
Proyek Ke-	w_i	p_i	p_i / w_i	
1	2	15300	7650	0
2	2	23000	11500	1
3	2	10500	5250	0
4	4	26950	6737,5	0
5	2	24650	12325	1
6	2	12425	6212,5	0
Total keuntungan				47650
Total SDM yang terpakai				4
Jumlah SDM yang tersedia				9

FRACTIONAL KNAPSACK

Jumlah SDM yang tersedia $W = 9$

Properti Proyek				Greedy by density
Proyek Ke-	w_i	p_i	p_i / w_i	
1	2	15300	7650	0
2	2	23000	11500	1
3	2	10500	5250	0
4	4	26950	6737,5	0
5	2	24650	12325	1
6	2	12425	6212,5	0
Total keuntungan				47650
Total SDM yang terpakai				4
Jumlah SDM yang tersedia				9

FRACTIONAL KNAPSACK

Jumlah SDM yang tersedia $W = 9$

Properti Proyek				Greedy by density
Proyek Ke-	w_i	p_i	p_i / w_i	
1	2	15300	7650	1
2	2	23000	11500	1
3	2	10500	5250	0
4	4	26950	6737,5	0
5	2	24650	12325	1
6	2	12425	6212,5	0
Total keuntungan				62950
Total SDM yang terpakai				6
Jumlah SDM yang tersedia				9

FRACTIONAL KNAPSACK

Jumlah SDM yang tersedia $W = 9$

Properti Proyek				Greedy by density
Proyek Ke-	w_i	p_i	p_i / w_i	
1	2	15300	7650	1
2	2	23000	11500	1
3	2	10500	5250	0
4	4	26950	6737,5	0
5	2	24650	12325	1
6	2	12425	6212,5	0
Total keuntungan				62950
Total SDM yang terpakai				6
Jumlah SDM yang tersedia				9

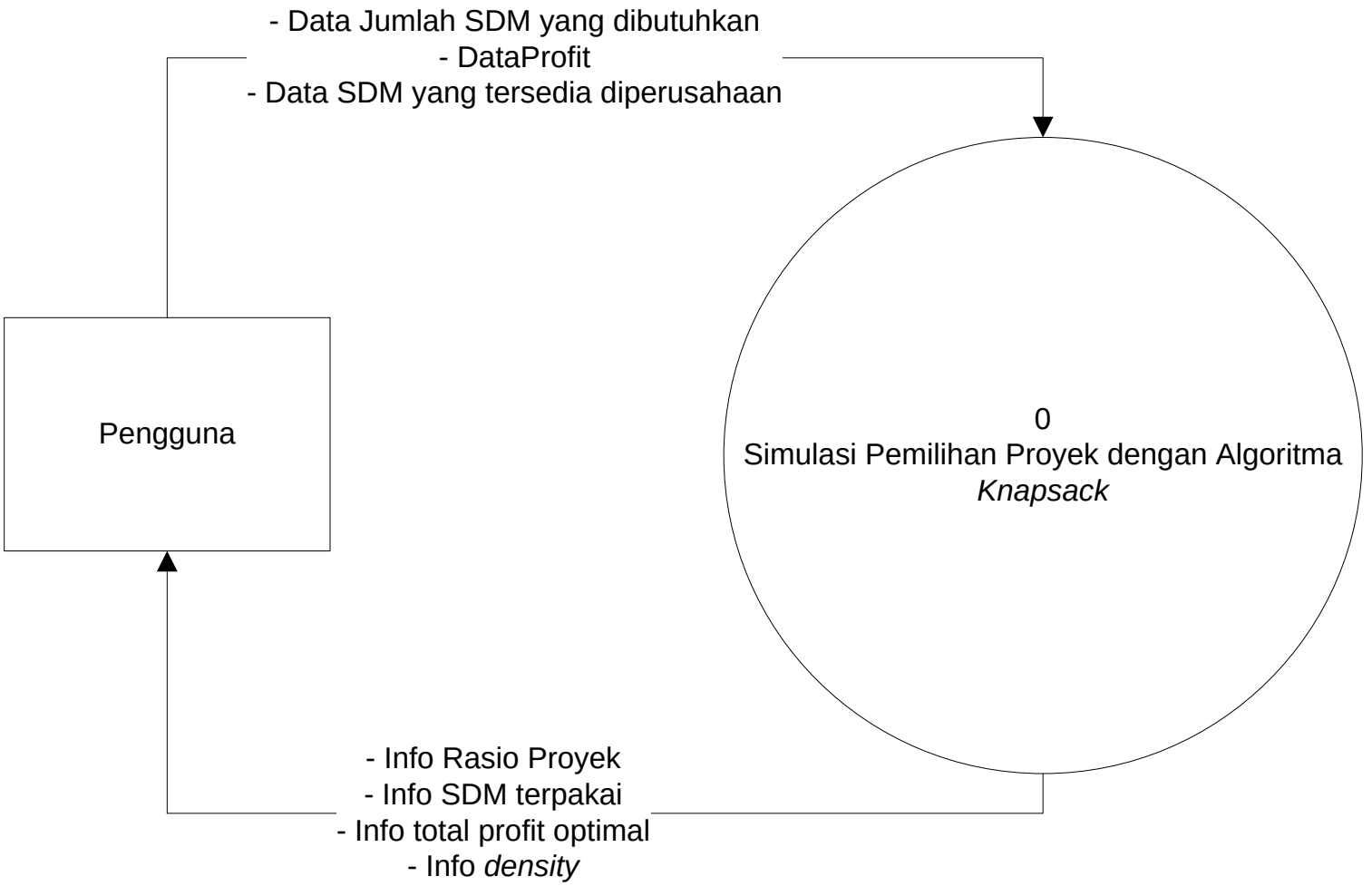
FRACTIONAL KNAPSACK

Jumlah SDM yang tersedia $W = 9$

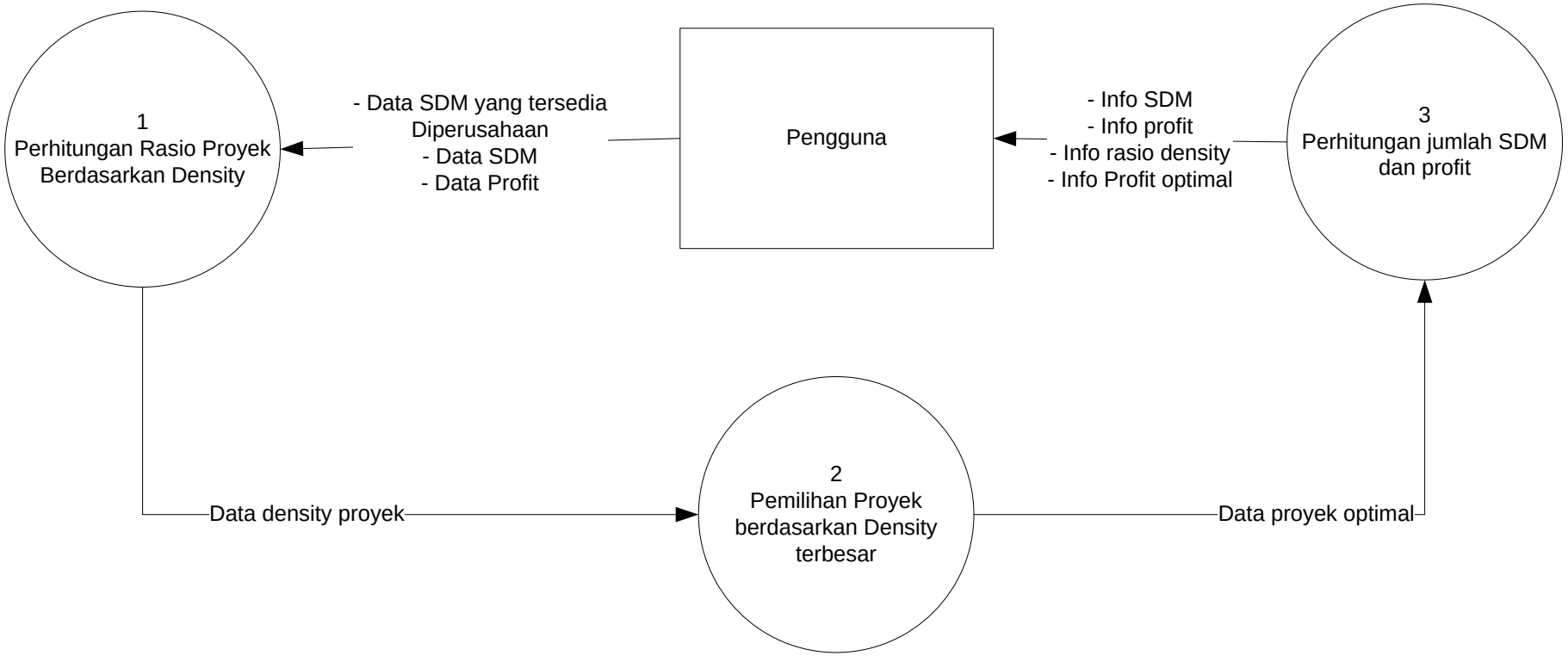
Properti Proyek				Greedy by density
Proyek Ke-	w_i	p_i	p_i / w_i	
1	2	15300	7650	1
2	2	23000	11500	1
3	2	10500	5250	0
4	4	26950	6737,5	0,75 or 75%
5	2	24650	12325	1
6	2	12425	6212,5	0
Total keuntungan				83163
Total SDM yang terpakai				9
Jumlah SDM yang tersedia				9


Perancangan

Diagram Konteks



DFD Level 1


1
Perhitungan Rasio Proyek Berdasarkan Density
- Data SDM yang tersedia Diperusahaan
- Data SDM
- Data Profit
Pengguna
- Info SDM
- Info profit
- Info rasio density
- Info Profit optimal
3
Perhitungan jumlah SDM dan profit
Data density proyek
2
Pemilihan Proyek berdasarkan Density terbesar
Data proyek optimal

Implementasi



Spesifikasi Implementasi Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	Processor	Pentium Dual Core T4200 2.00 GHz
2	RAM	2 Gb
3	Harddisk	250 Gb
4	VGA	256 Mb
5	Monitor	1024x768

Spesifikasi Implementasi Perangkat Lunak

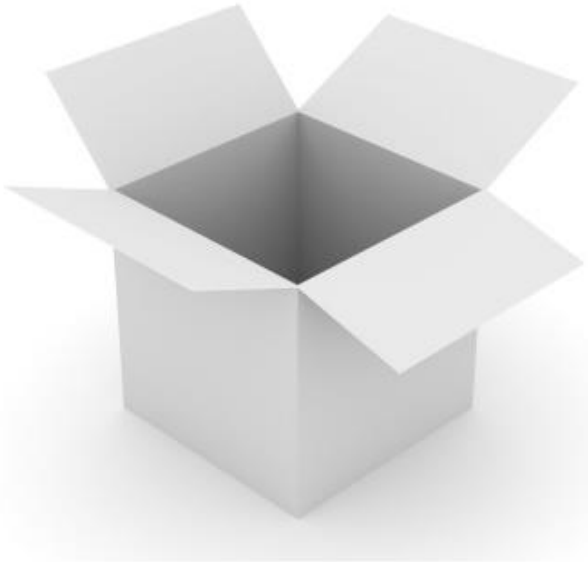
No	Perangkat Lunak	Keterangan
1	Windows 8	Sistem Operasi
2	Delphi	Tool Pembangun Perangkat Lunak
3	Visio	Pemodelan Sistem

Implementasi Antar Muka

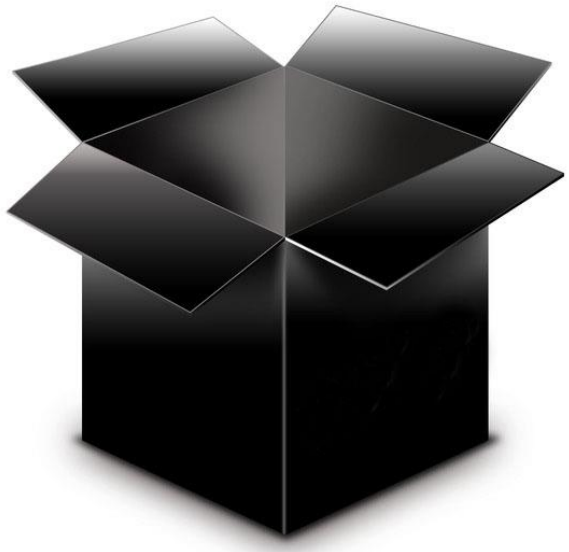
No	Form Antarmuka	Proses	Nama File	Keterangan
1	Jumlah SDM di Perusahaan	Inputan jumlah SDM yang tersedia di perusahaan	Unit_SDM.dfm	Tampilan untuk menginputkan data jumlah SDM yang tersedia di perusahaan
2	Form Input Data Proyek	Proses inputan data proyek	UnitPemilihanKnapsack.dfm	Tampilan untuk menginputkan data proyek
3	Form Tampil Data Perhitungan	Proses tampil data perhitungan knapsack	Unit_Hasil.dfm	Tampilan untuk menampilkan hasil perhitungan knapsack



Pengujian



&

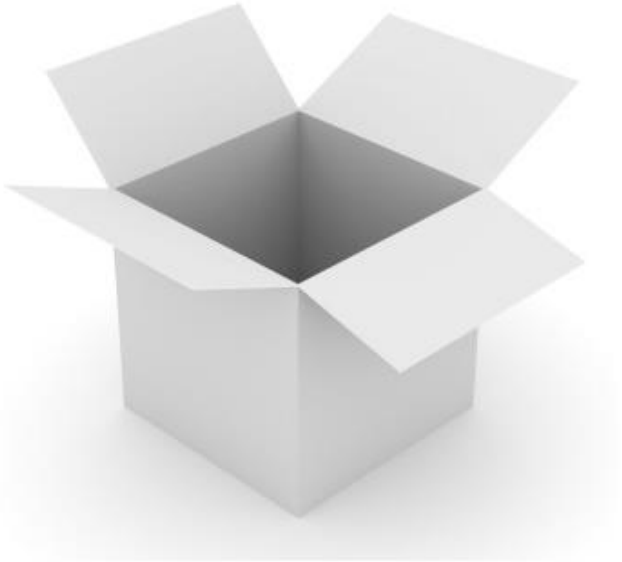


White Box & Black Box

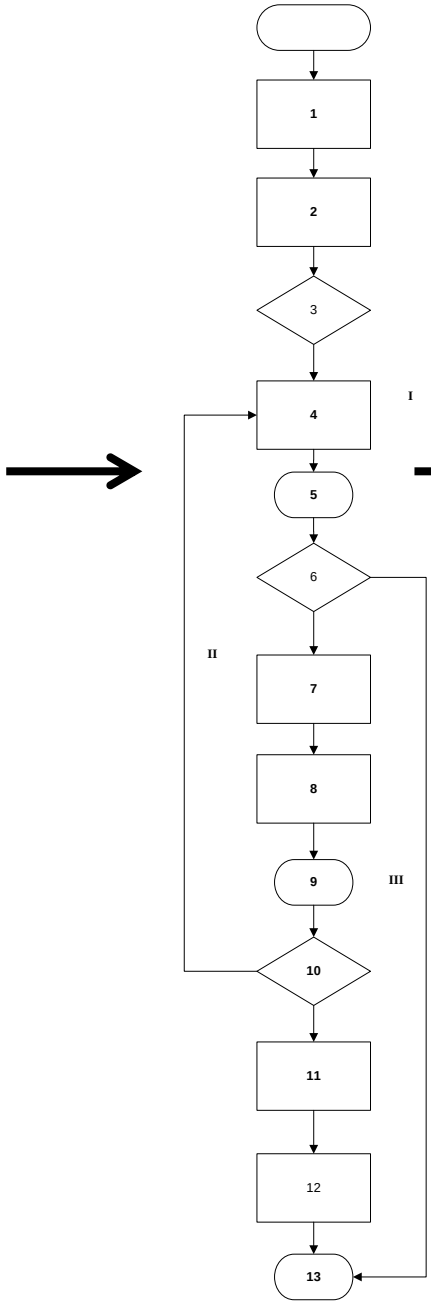
Rencana Pengujian

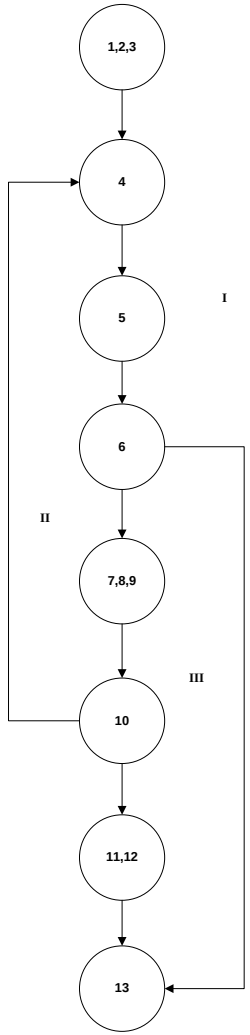
No	Item Uji	Detail Pengujian	Jenis Uji
1	Uji Algoritma knapsack	Cyclometic Complexity, independent path, region, predicate node, graph matrix	White box
2	Validasi data masukkan profit dan SDM	Masukkan data SDM yang tersedia di perusahaan	Black Box
		Masukkan data Jumlah SDM yang dibutuhkan	
		Masukkan data Profit	
3	Perhitungan knapsack / pemilihan solusi optimum proyek	Keluaran Status solusi optimal proyek	Black Box
		Keluaran Total profit optimum	Black Box

White Box



```
1     Opt_sol←0
2     i←1
3       for i←1 to n do
4         optimum[i] ← 0
5       Endfor
6       if SDM[i] > k
          then  Eof
7     Else optimum[i] ← 1
8       k←k-SDM[i]
9     Endif
10    if i<= n
11      then optimum[i] ← k/SDM[i]
12/13   opt_sol←0 (asumsikan ada endif)
```





Cyclometic Complexity

V(G) = 3

Independent Path

=

3 Path

Hitung Jumlah Region

=

3 Region

Predicate Node

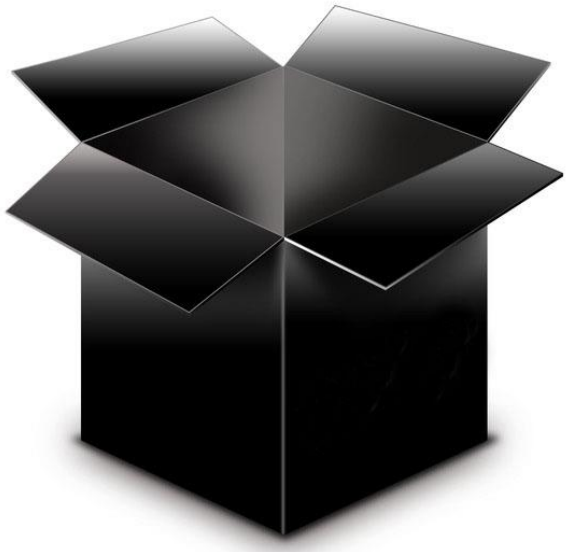
=

3 Predicate Node

Graph Matriks

$V(G) = 3$

Black Box



No	Item Uji	Detail Pengamatan	Hasil Pengamatan
1	Masukkan Jumlah SDM yang tersedia di perusahaan	Data Benar, Data Salah dan Data Kosong	[✓] Diterima [ ] Ditolak
	Masukkan data Profit	Data Benar, Data Salah dan Data Kosong	[✓] Diterima [ ] Ditolak
	Masukkan data Jumlah SDM yang dibutuhkan	Data Benar, Data Salah dan Data Kosong	[✓] Diterima [ ] Ditolak
2	Keluaran Status solusi optimal proyek	Data Benar, Data Salah dan Data Kosong	[✓] Diterima [ ] Ditolak
	Keluaran Total profit optimum	Data Benar, Data Salah dan Data Kosong	[✓] Diterima [ ] Ditolak

Properti Proyek				Fractional Knapsack With Greedy by density		Kesimpulan
i	w_i	p_i	p_i / w_i	Solusi Perhitungan Matematis	Solusi dengan Simulasi	
1	2	50000	25000	1	1	Diterima
2	3	51000	17000	0	0	Diterima
3	2	45000	22500	1	1	Diterima
4	4	123000	30750	1	1	Diterima
5	2	40000	20000	0	0	Diterima
6	2	56000	28000	1	1	Diterima
Total SDM Terpakai				10	10	Diterima
Total keuntungan				274000	274000	Diterima
Waktu Eksekusi Perangkat Lunak				1197 milisecond		
Kebutuhan Memori Saat Eksekusi				7.468 KB		

Kesimpulan Pengujian

Perangkat lunak simulasi dapat berfungsi sesuai dengan yang diharapkan serta perangkat lunak simulasi yang dibangun telah memenuhi tujuan awal penelitian


Kesimpulan

Kesimpulan (1)

Algoritma *fractional knapsack* dengan strategi *greedy by density* memiliki performansi yang baik dalam kasus pemilihan proyek dengan studi kasus PT. GITS Indonesia

Kesimpulan (2)

Algoritma *fractional knapsack* dengan menggunakan strategi *greedy by density* menghasilkan pemilihan proyek yang optimal dalam kasus pemilihan proyek


Saran

Pengembangan penelitian dapat dilakukan dengan mengukur dan memperhitungkan kemampuan sumber daya manusia

TERIMAKASIH!

