

MODUL PRAKTIKUM

Matematika Informatika 2



Versi	3.1
Tahun Penyusunan	2012
Tim Penyusun	1. Naeli Umniati, ST., MMSI
	2. Lilis Ratnasari, ST., MMSI
	3. Puji Zulaikasari, ST.
	4. Rogayah, ST
	5. Andika Januarioanto, ST

Laboratorium Teknik Informatika

Jurusan Teknik Informatika

Fakultas Teknik Industri

UNIVERSITAS GUNADARMA

Pertemuan 1

Vektor

Objektif:

1. Praktikan memahami teori dasar vektor.
2. Praktikan mampu mencari penjumlahan vektor, perkalian vektor dari 2 buah vektor.
3. Praktikan dapat membuat program tentang penjumlahan vektor, perkalian vektor dari 2 buah vektor.

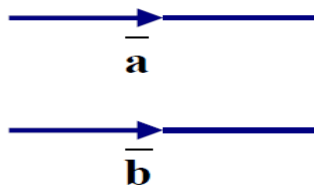
P1.1 Teori

Pengertian Vektor

Vektor adalah suatu potongan (ruang, segmen) garis yang mempunyai arah. Kita dapat menggambarkan suatu vektor dengan memberi tanda panah pada titik ujungnya. Sedangkan untuk menuliskannya, kita dapat memakai salah satu notasi berikut \vec{a} , \vec{a} , \vec{a} , \vec{A} , \vec{AB} ataupun \vec{AB} , (yaitu vektor yang titik awalnya A dan titik ujungnya B) (Pada buku – buku notasi vektor biasanya dicetak tebal). Panjang dari vektor \vec{a} kita tulis $|\vec{a}|$.

Kesamaan Dua Vektor

Dua buah vektor dikatakan sama, jika panjang dan arahnya sama. (Arah sama, artinya mempunyai garis pembawa yang berhimpit atau sejajar, dengan arah panah sama). Jadi, vektor tidak tergantung kepada letaknya, tetapi tergantung pada panjang dan arahnya.



Jika $\vec{a} = \vec{b}$, maka untuk $|\vec{a}| = |\vec{b}|$ (besarnya sama) dan untuk arah $\vec{a} = \vec{b}$, yaitu kedua vektor tersebut sejajar serta searah. Apabila dua vektor $\vec{a} = \vec{b}$ memiliki hubungan $\vec{b} = -\vec{a}$, maka dapat dikatakan bahwa kedua vektor tersebut memiliki besar yang sama dengan posisi yang sejajar tetapi berlawanan arah.

Operasi – operasi pada vektor

1. Penjumlahan vektor

a. Metode jajaran genjang.

vektor hasil (resultan) yaitu $\vec{a} + \vec{b}$ diperoleh dari diagonal jajaran genjang yang dibentuk oleh \vec{a} serta \vec{b} setelah titik awal ditempatkan berhimpit.

b. Metode segitiga.

Resultan kita peroleh dengan menempatkan titik awal salah satu vektor (misalnya \vec{b}) pada titik ujung vektor yang lainnya, maka resultan adalah vektor bertitik awal dititik awal \vec{a} , dan bertitik ujung dititik ujung \vec{b} .

Catatan:

- ✓ Penjumlahan vektor bersifat komutatif, artinya untuk setiap vektor a dan b berlaku $a+b = b+a$; = maka pemilihan vektor mana yang didahulukan tidaklah menjadi persoalan.

2. Perkalian skalar

Kalau k suatu skalar bilangan riil, a suatu vektor, maka perkalian skalar ka menghasilkan suatu vektor yang panjangnya $|k|$ kali panjang a , dan arahnya sama dengan arah a bila k positif atau berlawanan dengan a bila k negatif. Bila $k = 0$ maka $ka = 0$; disebut vektor nol yaitu vektor yang titik awal dan titik ujungnya berhimpit.

Catatan:

- ✓ Sebagai gabungan dari operasi penjumlahan serta perkalian skalar, kita dapat mengurangi vektor-vektor. Misalnya $a - b = a + (-b)$ yaitu menjumlahkan a dan $-b$. Tentu saja pengurangan vektor tidak komutatif, $a-b \neq b-a$.

P1.2 Contoh Kasus

Untuk mencari perhitungan vektor menggunakan operasi-operasi pada vektor itu sendiri (operasi penjumlahan, pengurangan, dan perkalian) maka kita perlu menggunakan bahasa pemrograman Java dan juga bantuan text editor (Notepad / Notepad++ atau EditPlus). Kemudian ketikkan sintak program di bawah ini:

```
import java.io.*;

class vektor
{
    public static void main (String[]args) throws IOException
    {
        BufferedReader input = new BufferedReader ( new
        InputStreamReader (System.in));

        int [] jum = new int [10];
        int [] v1 = new int [10];
        int [] v2 = new int [10];

        System.out.println ("masukkan jumlah elemen vektor : ");

        int n = Integer.parseInt (input.readLine());
        System.out.println ("Vektor 1");
```

```

for (int i=0; i<n; i++)
{
System.out.print("elemen [1,"+i+"]=");
v1[i]=Integer.parseInt(input.readLine());
}
System.out.println ("Vektor 2");
for(int i=0;i<n;i++)
{
System.out.print("elemen [1,"+(i+1)+"]=");
v2[i]=Integer.parseInt(input.readLine());
}
System.out.print("\npenjumlahan vektor:["");
for(int i=0;i<n;i++)
{
jum[i] = v1[i]+v2[i];
System.out.print (jum[i]);
if (i!=n-1)
{
System.out.print (",");
}
}
System.out.print ("]");
System.out.print ("\npengurangan vektor : [");
for (int i=0; i<n; i++)
{
jum[i] = v1[i] - v2[i];
System.out.print (jum[i]);
if (i!=n-1)
{
if (jum [i]<0)
{
System.out.print (" (vektor berlawanan arah)");
}
System.out.print (",");
}
}
System.out.print ("]");
System.out.print ("\nperkalian vektor :[");
for (int i=0; i<n; i++)
{
jum[i]=v1[i]*v2[i];
System.out.print (jum[i]);
if (i!=n-1)
{
System.out.print (",");
}
}
System.out.println ("]");
}
}

```

Sintak program di atas adalah untuk operasi-operasi pada vektor, kita dapat membuat kondisi untuk operasi penjumlahan, pengurangan dan penjumlahan. Program di mulai dari penulisan package dari java.io yg berbasis Objek kemudian pembentukan class. Selanjutnya penggunaan BufferedReader untuk input :

```
BufferedReader input = new BufferedReader ( new  
InputStreamReader (System.in));
```

Kemudian kita menentukan dimensi atau panjang range yang akan di input, contoh :

```
int [] jum = new int [10];
```

sintak di atas berarti bahwa type data integer dan memiliki dimensi atau range sampai indeks ke 10. Setelah itu kita masuk pada tahap deklarasi variabel yg akan di jadikan jumlah vektor :

```
int n = Integer.parseInt (input.readLine());
```

lalu kondisi penjumlahan sbb :

```
jum[i] = v1[i]+v2[i];  
System.out.print (jum[i]);  
if (i!=n-1)
```

pada operasi pengurangan agak berbeda, pada pengurangan harus diberikan kondisi tambahan untuk menyatakan nilai positif, karena nilai positif pada vektor berarti arah vektor tersebut berlawanan, contoh :

```
for (int i=0; i<n; i++){  
jum[i] = v1[i] - v2[i];  
System.out.print (jum[i]);  
if (i!=n-1){  
if (jum [i]<0){  
System.out.print (" (vektor berlawanan arah)");  
}  
System.out.print (",");
```

Untuk operasi perkalian sama seperti penjumlahan, hanya diganti tanda '+' menjadi '*'

Setelah kita selesai menuliskan sintak program tersebut, maka save file tersebut pada direktori E:\java (buat folder dengan nama folder : java) kemudian save dengan nama file: vektor. Kompilasi program tersebut dengan cara menjalankan program javac (java compiler). Jika anda bekerja di lingkungan windows buka command prompt, kemudian ketik program berikut ini :

```
E:\java>javac vektor.java
```

Proses kompilasi akan menghasilkan file yang berekstensi .class, file inilah yang akan kita eksekusi. Jika anda bekerja di lingkungan windows lakukan perintah berikut ini:

```
E:\java>java vektor
```

Program yang kita ketikkan bila dijalankan pada command prompt tampilannya adalah sebagai berikut:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

G:\Users\Ainan>E:
E:>cd java
E:\java>java vektor
masukkan jumlah elemen vektor :
3
Vektor 1
elemen [1,0]=1
elemen [1,1]=2
elemen [1,2]=3
Vektor 2
elemen [1,1]=3
elemen [1,2]=2
elemen [1,3]=1

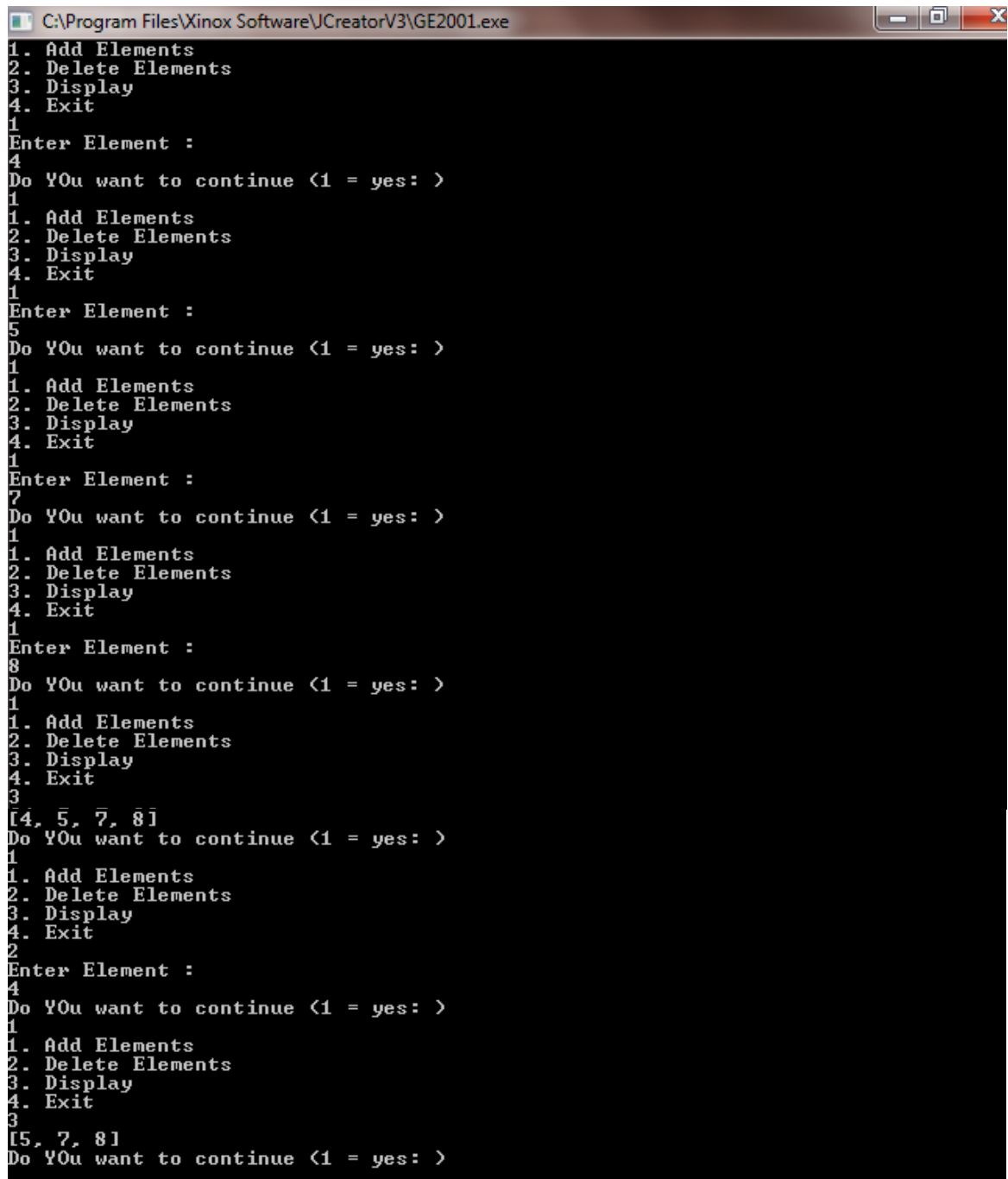
penjumlahan vektor:[4,4,4]
pengurangan vektor : [-2 <vektor berlawanan arah>,0,2]
perkalian vektor : [3,4,3]

E:\java>
```

Gambar 1.1 Output hasil perhitungan elemen vektor menggunakan operasi pada vektor

P1.3 Latihan

Buat program untuk menghasilkan perhitungan untuk beberapa operasi-operasi pada vektor dengan output seperti di bawah ini:



```
C:\Program Files\Xinox Software\JCreatorV3\GE2001.exe
1. Add Elements
2. Delete Elements
3. Display
4. Exit
1
Enter Element :
4
Do YOU want to continue <1 = yes: >
1
1. Add Elements
2. Delete Elements
3. Display
4. Exit
1
Enter Element :
5
Do YOU want to continue <1 = yes: >
1
1. Add Elements
2. Delete Elements
3. Display
4. Exit
1
Enter Element :
7
Do YOU want to continue <1 = yes: >
1
1. Add Elements
2. Delete Elements
3. Display
4. Exit
1
Enter Element :
8
Do YOU want to continue <1 = yes: >
1
1. Add Elements
2. Delete Elements
3. Display
4. Exit
3
[4, 5, 7, 8]
Do YOU want to continue <1 = yes: >
1
1. Add Elements
2. Delete Elements
3. Display
4. Exit
2
Enter Element :
4
Do YOU want to continue <1 = yes: >
1
1. Add Elements
2. Delete Elements
3. Display
4. Exit
3
[5, 7, 8]
Do YOU want to continue <1 = yes: >
1
```


P1.4 Daftar Pustaka

<http://ainanug.blogspot.com/2011/03/program-vektor-pada-java.html>

<http://kumpulanprogram.nikisunday.com/2011/01/java-implementasi-vektor.html>

<http://vimyza.blogspot.com/2010/05/vektor-pada-java.html>

Yahya, Yusuf. *Matematika Dasar*. Penerbit Ghalia Indonesia. 2001